Taming Large Rule Models in Rough Set Approaches

Thomas Ågotnes[1], Jan Komorowski[1], and Terje Løken[2]

[1] Knowledge Systems Group, Department of Information and Computer Science,
Norwegian University of Science and Technology, N-7491 Trondheim, Norway
{agotnes, janko}@idi.ntnu.no
[2] Department of Computer Science,
SINTEF Telecom and Informatics, N-7465 Trondheim, Norway
Terje.Loken@informatics.sintef.no

Abstract. In knowledge discovery from uncertain data we usually wish to obtain models that have good predictive properties when applied to unseen objects. In several applications, it is also desirable to synthesize models that in addition have good descriptive properties. The ultimate goal therefore, is to maximize both properties, i.e. to obtain models that are amenable to human inspection and that have high predictive performance. Models consisting of decision or classification rules, such as those produced with rough sets [19], can exhibit both properties. In practice, however, the induced models are often too large to be inspected. This paper reports on two basic approaches to obtaining manageable rule-based models that do not sacrifice their predictive qualities: *a priori* and *a posteriori* pruning. The methods are discussed in the context of rough sets, but several of the results are applicable to rule-based models in general. Algorithms realizing these approaches have been implemented in the Rosetta system. Predictive performance of the models has been estimated using accuracy and receiver operating characteristics (ROC). The methods has been tested on real-world data sets, with encouraging results.

1 Introduction

Rough set theory, introduced by Pawlak [19], provides a theoretically sound framework for extracting models, in the form of propositional decision rules, from data. There are two main tasks for which a model is useful: *prediction* and *description*. When performing knowledge discovery from databases (KDD), we are interested in finding as good a model as possible from a set of data. However, what constitutes a good model may vary, depending on the goals of the particular KDD process. If the goal is to build a model that is able to classify unseen objects as accurately as possible then the predictive quality is all-important. If the goal of the KDD process is description, we need to formalize what it means that a model displays good descriptive qualities. This is a difficult task, but the *size* of the model is of fundamental importance. A model consisting of thousands of

J.M. Żytkow and J. Rauch (Eds.): PKDD'99, LNAI 1704, pp. 193–203, 1999.

rules, or utilizing several hundred attributes is incomprehensible, while a small model, for example containing in the neighborhood of twenty relatively short rules or less, is easily understood by a human.

There are two fundamentally different approaches to finding models displaying good descriptive characteristics. The first approach starts out by generating as good a predictive model as possible, ignoring the descriptive quality. It is then believed that within this model there are some patterns which are of fundamental importance, and others which are redundant or only apply to a small number of objects in the data. Each model consisting of rules which are contained in the original model is called a *submodel*, and using various *filtering* strategies, one accepts a small drop in predictive performance in exchange for a submodel which is significantly smaller than the original model. The other approach aims at locating the most important patterns directly from the data. This approach is based on a generalization of Kowalczyk's *Rough Data Modelling* [12].

The first approach may be computationally expensive, because the problem of finding all models (i.e. reducts) is NP-hard [23], even if it may, at times, be alleviated through the use of appropriate heuristics.

In the second approach, the cost of computing is much smaller, for the price of, possibly, not finding the best model. However, the models that are found may be sufficiently good for the problem at hand. It may also be that the computational cost of the first approach is indeed unacceptable.

Clearly, there are advantages and disadvantages with both approaches and there is an apparent need to explore both dimensions in depth.

The models considered here are induced from data and are on the form $RUL = \{r_1, \ldots, r_n\}$, $r_i = \alpha_i \to \beta_i$, 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 [18, 11] which is a tool-kit for data mining and knowledge discovery within the framework of rough set theory.

Predictive performance of the models is estimated using 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 [25]. Differences in calculated AUC values for two classifiers may be due to chance. Hanley and McNeil [7] provide a statistical hypothesis test for detecting statistically significant difference in two correlated (calculated from the same data) AUC values. We used this test with a 5% significance level for the two-tailed test using Pearson's correlation measure.

Both approaches are illustrated here with experiments on real-world data sets, and the results are encouraging. The methods and experiments are further discussed in [1] and [13].

In the following it is assumed that the reader has some familiarity with the use of the rough set framework for synthesizing models from data. A tutorial introduction to rough sets may be found in [10]. Kowalczyk's method is, however, briefly explained, as it may be less known.

2 Filtering Strategies

2.1 A Posteriori Filtering

The general problem of post-pruning of models is finding submodels with lower complexity but without significantly lower predictive performance. Thus, two general properties of models are *performance* and *complexity*. 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.

Genetic Filtering A genetic algorithm for rule filtering was implemented in the Rosetta framework, based on an implementation by Vinterbo [26]. Given an initial rule set $RUL = \{r_1, \ldots, r_n\}$ the search space $\{R : R \subseteq RUL\}$ is represented by bit strings of length n, where the semantics of a bit string $b_1 \cdots b_n$ is the rule set $\{r_i : b_i = 1\}$. The fitness function is a weighted sum of the performance and the inverse of rule complexity. Both the weight and the termination criteria for the algorithm is set by the user.

Quality-based Filtering Numerical measures of the quality of individual decision rules are generally derived from the contingency tables. The contingency table for the rule $r = \alpha \to \beta$ tabulates the number of objects from the rule's originating decision system that matches the antecedent and/or the consequent. For each $\phi \in \{\alpha, \neg\alpha\}$ and $\psi \in \{\beta, \neg\beta\}$, $n_{\phi,\psi}$ denotes the number of objects matching both ϕ and ψ. Also, $n_\psi = n_{\psi,\beta} + n_{\psi,\neg\beta}$ and $n_\psi = n_{\alpha,\psi} + n_{\neg\alpha,\psi}$. The total number of objects in the originating decision system is denoted $|U|$. Relative frequencies are often 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 $r = \alpha \to \beta$ are $accuracy(r) = n_{\alpha,\beta}/n_\alpha$ and $coverage(r) = n_{\alpha,\beta}/n_\beta$. Coverage was used as a measure for rule filtering in [16].

An overview of rule quality formulae is given by Bruha in [3] and is summarized in Table 1. We used these quality functions for rule filtering by plotting the performance of the model versus the number of the individually best rules included from the corresponding orderings of the unfiltered rule set, and selecting models according to given problem specific criteria.

2.2 A Priori Filtering

Several approaches to a priori rule filtering exist. In this paper, Rough Data Modelling, and its generalization, Rough Modelling, have been investigated.

Rough Data Modelling is a method introduced in [12] which attempts to address two common problems found in traditional data mining methods: the

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	$2\,(n_{\alpha,\beta} \ln(n_{\alpha,\beta} \cdot \vert U\vert/(n_{\alpha} \cdot n_{\beta})) + n_{\alpha,\neg\beta} \ln(n_{\alpha,\neg\beta} \cdot \vert U\vert/(n_{\alpha} \cdot n_{\neg\beta})))$
J	$quality_{G2}(r)/(2\vert U\vert)$
Cohen	$(\vert U\vert \cdot n_{\alpha,\beta} + \vert U\vert \cdot n_{\neg\alpha,\neg\beta} - n_{\alpha} \cdot n_{\beta})/(\vert U\vert^{2} - n_{\alpha} \cdot n_{\beta} - n_{\neg\alpha} \cdot n_{\neg\beta})$
Coleman	$(\vert U\vert \cdot n_{\alpha,\beta} - n_{\alpha} \cdot n_{\beta})/(n_{\alpha} \cdot n_{\neg\beta})$
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 [3]

computation cost of model generation and the inability to tailor the method to the specific needs of each data mining session. Kowalczyk argues, as do many others, that knowledge discovery should be looked upon as an iterative process where the goals of the process may vary. With the large number of alternatives found at each stage in the knowledge discovery process (feature selection, discretization, data mining, etc.), it is impossible to find general guidelines which will always produce the best model. This means that it is desirable to generate as large a number of models as possible and search among these in order to find a satisfying model. However, the computational cost of many commonly used algorithms prohibits the generation of numerous models, and in addition, most algorithms are specifically designed to maximize (or minimize) a certain measure, or a predefined combination of measures (accuracy, specificity, misclassification cost, etc.).

Rough data modelling simplifies the model generation process and makes it feasible to search among a large number of models. In addition, it allows the user to tailor the data mining process to suit his/her own needs, by allowing the user to specify in detail how to evaluate a model. From a decision system $\mathcal{A} = (U, A \cup \{d\})$, a rough data model is a triple $\mathcal{M} =< B, \bar{d}_B, \preceq >$, where B is an attribute subset of A that induces a set of equivalence classes E^B. $\bar{d}_B$ is a *class decision function*, which maps the equivalence classes E^B to a single decision value for each class, and $\preceq$ a linear ordering of the equivalence classes in E^B. Both the class decision function and the ordering on the classes are decided by the user. The process of generating high-quality models using rough data modelling is a three-step process [12]:

1. Specify the performance measure that is to be optimized, and specify the measure used to rank classes in a model.
2. Determine the search space, i. e. the collection of models that should be searched for a model maximizing the measure found in step 1. This step also involves determining the class decision function $\bar{d}_B$.

3. Determine the search procedure.

And so now, each equivalence class in a particular rough data model is uniquely identified by the values of the attributes in B. It is thus equivalent to a single rule of the form

$$(a_1(x) = v_1) \wedge \ldots \wedge (a_n(x) = v_n) \rightarrow (d(x) = \bar{d}_B([x]_B)) \tag{1}$$

where $B = \{a_1, \ldots a_n\}$, and $\{v_1, \ldots v_n\}$ are the characteristic values of the equivalence class of x, $[x]_B$.

The complexity of rough data modelling is linear with respect to the number of objects in the data set [12]. The size of the search space is decided by the user-specified upper and lower bounds on how many attributes to include in the model, and it is only feasible to search for models which use relatively few (5-10) attributes.

Rough data modelling is closely related to several other strategies for extracting small, yet accurate models from possibly large data sets, for example rough classifiers [20] and feature subset search [9]. It is desirable to develop a unified framework for mining compact, yet accurate models which is able to incorporate new developments in rough set theory, such as replacement of the discernibility relation with a similarity or tolerance relation [22, 24], or the incorporation of ordinal properties of attribute values obtained through the use of a dominance relation [5].

As a step towards such a framework, we introduce the concept of *rough models*, which is a generalization of rough data models. While a rough data model *partitions* the universe U using the indiscernibility relation, a rough model is only required to have classes that *cover* U, imposing no restrictions on how these are found. For a decision system $\mathcal{A} = (U, A \cup \{d\})$, a rough model is a tuple

$$\mathcal{M} =< B, \hat{d}_B, E, \preceq, \mathcal{R} > \tag{2}$$

where $B \subseteq A$ is a set of attributes, $\hat{d}_B : U \rightarrow V_d$ is an *object* decision function, not a class decision function as used in rough data models. This allows us the added flexibility of being able to assign different decision values for objects which belong to the same decision class, if we so desire. E is a set of object classes which cover the universe U, $\preceq$ is a linear ordering on the classes in E, and $\mathcal{R}$ is a set of reducts for the decision system $\mathcal{A}'$, which is obtained by replacing the decision values of each object $x \in U$ with $\hat{d}_B(x)$.

Rough data models are a kind of rough models where $\mathcal{R} = B$, and $E = U/IND(B)$. In addition, one can easily define, for example, *rough similarity models*, where the classes E are induced by a form of similarity relation [24][1], *rough dominance models*, where the dominance relation, explained in [5], is used to partition the universe into classes E, or *rough Holte models*, where each attribute in B is regarded as a reduct, yielding a model containing univariate rules.

[1] In this case, the classes will overlap.

3 Experimental Results

3.1 Rule Filtering

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 [6]. This data set has previously been mined in [4] 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 [14], 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 learning, hold-out and testing sets three times. After preprocessing, rule induction (using dynamic reducts [2]) were performed using the learning sets. Rule filtering was done using the hold-out sets, and the performance assessed and compared to the unfiltered sets using the testing sets.

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 the maximal size of a rule set. The execution of the experiments differed in the selected parameters (e.g. the weight for the fitness function in the genetic algorithm), and in the procedure used to select a particular filtered model from several alternatives.

The estimated performance of the pruned models is shown in Table 2.

	Acute Appendicitis			Cleveland		
Method	Size	AUC (SE)	p-value	Size	AUC (SE)	p-value
Unfiltered	447	0.9043 (0.0363)		6949.67	0.9035 (0.0374)	
Genetic	6.67	0.8930 (0.3318)	0.6209	34	0.8776 (0.0377)	0.4475
Michalski ($\mu = 0$)	9.67	0.8890 (0.0392)	0.6195	65	0.8920 (0.0353)	n/a
Pearson	8.33	0.8661 (0.0426)	0.3253	92.33	0.8846 (0.0366)	n/a
J	7	0.8734 (0.0411)	0.4813	119.33	0.8753 (0.0380)	n/a

Table 2. Performance of filtered versus unfiltered rule sets on previously unseen data. Only the best quality functions are shown. All values are averages over three different splits of the data set. p-values outside the Hanley/McNeil lookup table are specified as n/a.

3.2 Rough Modelling

In order to investigate the performance of the rough modelling approach within the ROSETTA toolkit, several data sets were analyzed using rough modelling, as well as traditional rough set methods for comparison. All results shown below were computed using a genetic algorithm for rough model search, as initial experiments proved this algorithm on average used only 20-30% of the time used by an exhaustive search and still consistently returned the same models. Rough models were generated from the Pima Indian diabetes data set from the UCI repository, and the acute appendicitis data set described earlier.

In order to investigate the effect of excluding the smallest equivalence classes from the rough data models, the size threshold for inclusion into the rough model was varied and a rough model generated from two thirds of the objects in the data set. The remaining objects were used to test the performance of the resulting model, using ROC analysis. The results for both data sets are shown in Table 3, and the numbers are labeled A_O and P_O, for the appendicitis and diabetes data, respectively.

In addition to a decision function $\hat{d}_B$ which sets the decision value for each object to the dominating decision value for the equivalence class of that object, a decision function which copies the original decision value for each object was implemented. This breaks with the principle put forth by Kowalczyk that each equivalence class should have a single decision value associated with it and thereby produces a model which may contain indeterministic rules. Table 3 also shows the results using this decision function, labeled A_D and P_D for the two data sets used. The splits were unaltered to facilitate comparison of the AUC values.

The AUC values were compared using the Hanley-McNeil test. This was only done for the appendicitis data, as no known benchmark existed for the diabetes data. The numerical results indicate that rough data models perform somewhat worse than the best known RS model, but the difference is not significant if the correct threshold for class size is selected. This means that one may accept a slight drop in performance, but the drop is often insignificant. No good measure of descriptiveness exists, but while the RS models for the appendicitis data contained between 850 and 900 rules, the rough data models contained 15 rules or fewer if the smallest classes (less than 5 objects) were filtered out from the model. On all other data sets examined, the size of the models found by a rough model search were comparable to the results reported for the appendicitis set (between 5 and 20 rules, if a small cut-off on class size is used).

The performance of rough Holte models was only briefly investigated on the appendicitis data. The complete set of univariate rules for the data set, denoted `1R`, was used as a benchmark, and a 3-fold split-validation was carried out, with different lower limits on the classes to be included in the model. The results are labeled A_H in Table 3. No significant differences were found between the different rough Holte models and the `1R` rule set, or between the different rough Holte models. This means that it is possible to mimic the performance of the `1R` rule set (which in turn has been found to be comparable to the performance of the best reduct-based models) using only a handful of attributes.

Size	**Model AUC (SE)**				
limit	A_D	A_O	A_H	P_D	P_O
RS	0.907 (0.031)	0.907 (0.031)	0.907 (0.031)	–	–
1R	0.908 (0.032)	0.908 (0.032)	0.908 (0.032)	–	–
0	0.783 (0.049)	0.794 (0.048)	0.925 (0.028)	0.692 (0.036)	0.748 (0.034)
5	0.835 (0.043)	0.897 (0.033)	0.894 (0.034)	0.694 (0.036)	0.730 (0.034)
10	0.816 (0.045)	0.857 (0.039)	0.884 (0.035)	0.732 (0.035)	0.763 (0.033)
15	0.771 (0.050)	0.806 (0.047)	0.903 (0.032)	0.715 (0.035)	0.767 (0.033)

Table 3. Results from analysis on the acute appendicitis data and diabetes data, using a variety of rough models.

4 Discussion

4.1 Rule Filtering

The results from both experiments are rather 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 performance of the quality functions for rule filtering purposes was diverse. It seems that the Michalski formula with $\mu = 0$ (see Table 1) can be recommended. Corresponding to filtering according to the coverage only (similar to [16]), this is a rather surprising result. In addition to the Michalski formula, the Pearson χ^2 statistic and the J-measure seem to perform well. The results back up [3] in that the theoretically based quality formulae generally do not perform better than the empirically based. The genetic 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. For practical purposes, rule filtering with the genetic algorithm is much more time consuming than using quality-based filtering.

4.2 Rough Modelling

As with a posteriori pruning, the results obtained so far using rough modelling are not a solid enough foundation on which to make strong claims about the performance of rough models versus models mined using traditional rough set methods. There is some indication that the performance of rough models falls slightly short of the performance of larger rough set induced models, but the evidence is not conclusive. However, there is strong evidence supporting the conjecture that rough models of comparable performance to traditional RS models are far more descriptive. No universally agreed upon measure of descriptiveness exists, but a decrease in size from thousands of rules to between five and twenty, few attributes used, as well as the option of determinism, add up to a very large improvement in descriptive capability.

The results in the experiments performed support the observations made by Holte [8], that very simple rules often perform just as well as more complex rules. For the appendicitis data, it may seem that the predictive capabilities of univariate rules match and possibly exceed those of rough data models as well as the best known rough set-based models, which both contain rules with several attributes in the antecedent. However, univariate rules are less interesting from a knowledge discovery standpoint, as they only represent simple correlations between individual attribute values and a decision value. For a related discussion, see [17] in this volume.

5 A Comparison and Future Research

The results on rule filtering are rather encouraging. It is possible to obtain small models that preserve predictive quality. While the computational costs may be high, the method guarantees that no significant *combination* of features will be excluded from the original set. Applications of rule filtering vary widely in their goals, and it may be impossible to construct an automatic method serving all needs.

The simplicity of rough model generation means that rough modelling is applicable in particular to large databases; it is also well-suited as an initial approach to mining data. By searching for various forms of rough models, models which are of a high descriptive and predictive quality may be generated quickly. The insight gained from inspecting the rough models may then be used to process the data before using reduct-based model inducers to mine the best possible predictive model, if this is of interest.

The experiments presented here are still preliminary and not sufficient to draw general and definite conclusions regarding the applicability of the proposed methods. A further investigation should include a wider range of data sets, induction algorithms, and, in the case of genetic algorithms, various parameter settings. Also, more sophisticated experiment designs — such as cross-validation — should be considered.

We also plan to investigate the relationship of our approaches to the so-called templates developed now at Logic Section of Warsaw University [15].

Not surprisingly, we can conclude that a knowledge discoverer will be best served if both approaches are present in his/her arsenal. The choice of tool will depend on the task at hand. We hope that this work will help him/her decide which method to choose.

References

1. 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.
2. Jan G. Bazan, Andrzej Skowron, and Piotr Synak. Dynamic reducts as a tool for extracting laws from decision tables. In *Proc. International Symposium on Methodologies for Intelligent Systems*, number 869 in Lecture Notes in Artificial Intelligence, pages 346–355. Springer-Verlag, 1994.
3. 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.
4. 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.
5. S. Greco, B. Matarazzo, and R. Słowiński. New developments in the rough set approach to multi-attribute decision analysis. *Bulletin of International Rough Set Society*, 2(2/3):57–87, 1998.
6. 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.
7. 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.
8. R.C. Holte. Very simple classification rules perform well on most commonly used datasets. *Machine Learning*, 11:63–91, 1993.
9. R. Kohavi and B. Frasca. Useful feature subsets and rough set reducts. In T.Y. Lin and A.M. Wildberger, editors, *3rd International Workshop on Rough Sets and Soft Computing (RSSC '94), San Jose, USA*, 1994.
10. J. Komorowski, Z. Pawlak, L. Polkowski, and A. Skowron. Rough sets: A tutorial. In S.K. Pal and A. Skowron, editors, *Rough Fuzzy Hybridization – A New Trend in Decision-Making*, pages 3 – 98. Springer, 1999.
11. Jan Komorowski, Aleksander Øhrn, and Andrzej Skowron. ROSETTA and other software systems for rough sets. In Willy Klösgen and Jan Zytkow, editors, *Handbook of Data Mining and Knowledge Discovery*. Oxford University Press, 2000.
12. W. Kowalczyk. Rough data modelling : a new technique for analyzing data. In *Rough Sets and Knowledge Discovery 1: Methodology and Applications [21]*, chapter 20, pages 400–421. Physica-Verlag, 1998.
13. 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.
14. 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`].
15. Hoa S. Nguyen. *Data regularity analysis and applications in data mining*. PhD thesis, Warsaw University, 1999.
16. 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, 1998.

17. Aleksander Øhrn and Jan Komorowski. Diagnosing acute appendicitis with very simple classification rules. In Jan Rauch and Jan Zytkow, editors, *Proceedings of the Third European Symposium on Principles and Practice of Knowledge Discovery in Database (PKDD'99)*, Prague, Czech Republic, September 1999. Springer-Verlag.
18. 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, 1998.
19. Z. Pawlak. Rough sets. *International Journal of Information and Computer Science*, 11(5):341–356, 1982.
20. Z. Piasta and A. Lenarcik. Rule induction with probabilistic rough classifiers. Technical report, Warszaw University of Technology, 1996. ICS Research Report 24/96.
21. L. Polkowski and A. Skowron, editors. *Rough Sets in Knowledge Discovery 1: Methodology and Applications*, volume 1 of *Studies in Fuzziness and Soft Computing*. Physica-Verlag, 1998.
22. A. Skowron, L. Polkowski, and J. Komorowski. Learning tolerance relations by boolean descriptors, automatic feature extraction from data tables. In S. Tsumoto, S. Kobayashi, T. Yokomori, H. Tanaka, and A. Nakamura, editors, *Proceedings of the Fourth International Workshop on Rough Sets, Fuzzy Sets, and Machine Discovery, RSFD'96, Tokyo, Japan*, pages 11–17, 1996.
23. A. Skowron and C. Rauszer. The discernibility matrices and functions in information systems. In R. Słowiński, editor, *Intelligent Decision Support Systems — Handbook of Applications and advances in Rough Set Theory*, pages 331–362. Kluwer Academic Publishers, 1991.
24. R. Słowiński and D. Vanderpooten. Similarity relation as a basis for rough approximations. In P.P. Wang, editor, *Advances in Machine Intelligence & Soft Computing*, volume 4, pages 17–33. Duke University Press, 1997.
25. J. A. Swets. Measuring the accuracy of diagnostic systems. *Science*, 240:1285–1293, 1988.
26. 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.