Rough Set Based Data Exploration Using ROSE System

Bartłomiej Prędki, Szymon Wilk

Institute of Computing Science
Poznan University of Technology
Piotrowo 3A, 60-965 Poznań, Poland

Abstract. This article briefly describes the process of data exploration based on rough set theory and also proposes ROSE system as a useful toolkit for doing such data analysis on PC computers.

1 Introduction

Today, when the cost of acquiring and storing data is so low, many people keep electronic track of their activity. Doctors write down information about their patients in databases, sellers record transactions, etc. After accumulating all this information, the owners of data become interested in exploring it . Usually, they aim for the following goals:

- checking the consistency of data,
- reducing superfluous information,
- transforming data to knowledge, i.e. discovering interesting and useful information patterns hidden in data.

Currently, there are many techniques that can be used to achieve the above goals, including statistics, data mining and machine learning. If the objects in a database can be described using attributes and it is possible to discern condition and decision attributes then the data can be structured in, so called, information table, whose rows are objects and columns are condition and decision attributes. Each entry object-attribute is a value called descriptor. One of the methodologies gaining popularity during the last decade that can analyze data stored in information table, is the rough set theory, proposed by Pawlak [6]. It is especially useful with inconsistent data. This methodology is implemented in our software system called ROSE (Rough Set Data Explorer). In this article we present a data exploration process using the ROSE system.

2 ROSE system

ROSE is a software package developed in the Laboratory of Intelligent Decision Support Systems, Institute of Computing Science, Poznan Technical University. It implements the rough set based data exploration methodology with variable precision model and similarity relation extensions. It works on PCs running 32-bit operating systems (Windows 95/98/NT). It is a successor of the RoughDAS system - one of the first successful implementations of rough set theory [10].

ROSE is designed to be easy in use, point and click, menu-driven, user friendly tool for exploration and data analysis. It is meant as well for experts as for occasional users who want to perform the data exploration. System communicates with users using dialog windows and all the results are represented in the environment. Data can be edited using spreadsheet like interface.

ROSE is built using modular architecture. It means that every task is performed by standalone program module. For ease of use all modules are integrated in single environment - user interface.

ROSE was written and designed in C++ language. We have tried to obtain maximum transferability between operating systems. All computational modules can be compiled on platforms containing ANSI C++ compiler. So the engine of the system can be moved to more powerful UNIX machines. Only Graphical User Interface (GUI) and visualization modules are bound with Microsoft Windows systems.

To simplify management of the data exploration process ROSE uses special structures called projects. Project contains information concerning not only data file, but also current computational engine options, for example like α and β parameters of variable precision model.

We have decided to store all data and results in plain text files. It guarantees better reusability and allows easier import from databases or spreadsheets. We've introduced new file format called ISF. It supports better syntax checking and allows extended definition of attributes which can have real, coded or even lexical values. The import/export mechanism to other formats is also included (RoughDAS, LERS, etc.) as well as possibility to obtain data from commonly used database formats (dBase and Paradox).

3 Exploration process using ROSE

Data exploration process using ROSE system can be broken up into several phases, represented in Figure 1.

At first, a user of the system has to convert data to the information table and save it in ISF file. For ease of use import mechanism from several sources is provided.

If there are missing values the user should perform preprocessing phase. This phase is also necessary when there are continuous attributes and indiscernibility relation is to be used later in the exploration process.

Next, the user is obliged to select the model of data exploration. Currently ROSE supports three independent models described in detail in Section 5. Model selection may imply inavailability of some phases in data analysis process.

Depending on the earlier choice, the user can select following stages of data exploration: rough approximations, reduction of attributes, rule induction and classification.

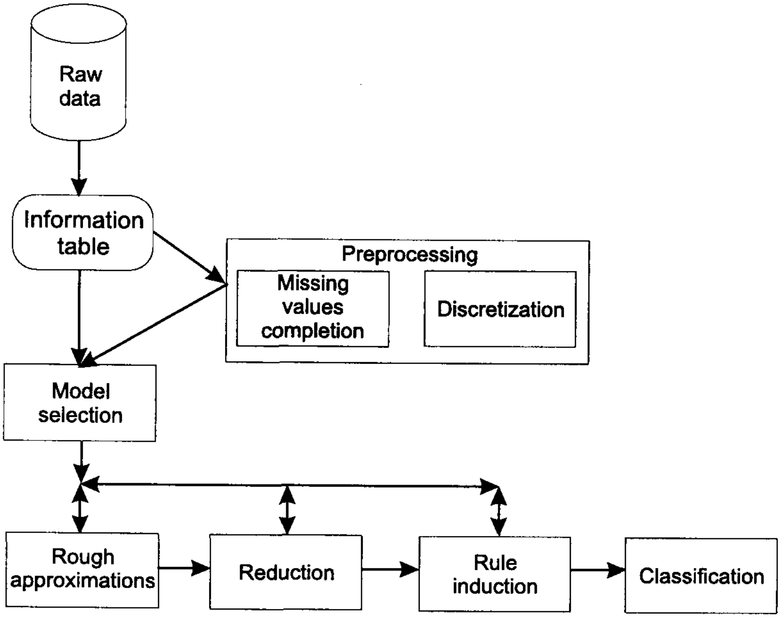

Fig. 1. Scheme of data exploration in ROSE

More information about each phase can be found in the following sections.

4 Preprocessing

The goal of preprocessing phase is to prepare data for further analysis to make it appropriate for used methodology.

In real life applications missing values are usually found in collected data. They pose a real problem for data exploration. The method implemented in module *completer* is based on statistical analysis of most frequent values (usually used in machine learning). In the future we plan to add more advanced methods, dealing with incomplete information [12].

When user selects exploration model based on indiscernibility relation it is suggested to replace continuous domain attributes with discrete ones. Such process is called discretization and there are two available approaches: discretization based on expert's knowledge and automatic discretization based on information theory. We have implemented both methods in ROSE system. To be more specific, automatic discretization uses algorithm based on entropy measure, introduced by Fayyad & Irani [1]. User driven discretization includes possibility to create discretization table or to visually divide attribute domain into intervals.

5 Available models

Currently, there are three data exploration engine models available in ROSE system: classical rough set model, variable precision rough set model and similarity-based rough set model.

Rough set theory was developed by Pawlak in 1982 [6] and since then it was successfully used in many applications. However, there are data sets that are more demanding, for example containing continuous values. That is the reason for research to expand the capabilities of this methodology. One of such extensions is variable precision model introduced by Ziarko [16]. It changes the definition of approximations (see Section 6). A different approach is used in similarity relation model introduced by Slowinski & Vanderpooten [13][14]. Similarity substitutes indiscernibility relation and keeps reflexivity as the only property from among reflexivity, symmetry and transitivity characterizing the indiscernibility. It allows analysis of numerical attributes without earlier discretization. It is also possible to induce rules with an extended syntax employing similarity.

In the near future we plan to include very new exploration models adapted to an understanding of some or all attributes as criteria with preference-ordered scales. The key element of these models is the dominance relation used instead of indiscerniblity or similarity [2]. These new exploration models proposed by Greco, Matarazzo and Slowinski can be combined with fuzzy logic in order to handle uncertainty and imprecision.

6 Approximations

The most important part of the rough set theory is approximation. The first step to find approximations is creating elementary sets (also called atoms). An elementary set contains objects indiscernible on all condition attributes, it means these objects have identical values on all condition attributes. When using similarity relation model, elementary sets are substituted by similarity classes.

When the cardinality of an elementary set is larger then one it is probable, that its objects will belong to different decision classes, so we have an ambiguity. Because of that we define two approximations:

1. Lower approximation with respect to the exploration model being used:
   - in the classical model - it contains all the elementary sets included in the decision class,
   - in the variable precision model - it contains all the elementary sets that have at least $\beta * 100\%$ objects belonging to the decision class,
   - in the similarity relation model - it contains all objects whose inverse similarity classes are included in the decision class,
2. Upper approximation with respect to the exploration model being used:
   - in the classical model - it contains all elementary sets that have a non-empty intersection with the decision class,

- in the variable precision model - it contains all elementary sets that have at least $\alpha * 100\%$ objects belonging to the decision class,
- in the similarity relation model - it contains all objects whose inverse similarity classes have a non-empty intersection with the decision class.

ROSE implements all of the above approximations.

Based on the approximations some measures are calculated, including accuracy of approximation, accuracy of classification, and most important in rough set theory, quality of classification.

7 Reduction

One of important properties of rough set theory is reduction of attributes. We want to check if some of the attributes are not redundant in the information table. There are usually many possibilities of selection of such attributes so the main goal of the reduction phase is looking for all the subsets of attribute set, which guarantee the same value of quality of classification as the complete set (it means they approximate the data in the same way). Those subsets are called reducts and the set of most significant attributes is called a core (it is also the common part of all reducts).

Beacuse the problem of finding all reducts for given information table is NP-hard, it is important to develop methods that, if possible, find all reducts in a reasonable time or introduce heuristic approaches generating some reducts. ROSE is currently equipped with four reduct generation methods.

Historically the first is an algorithm based on lattice search introduced by Romanski [7] which tries to reduce the search space by cutting-off some part which has no potential of including a reduct. It is useful, when the number of reducts is rather small (less then 1000), because of memory requirements.

The nowadays most efficient algorithm for reduct generation was developed by Skowron [8], based on discernibility matrix. It is very fast, although the initial cost of building the matrix can be a disadvantage for datasets having only a couple of reducts. For example, it is possible to generate 809 reducts in ESWL information table, containing 500 objects described by 26 attributes in 2 seconds on PC with Pentium 166MMX processor). The main limit of the algorithm is the memory requirement depending on the size of the information table.

For even larger datasets it is possible to search for some of the reducts using the heuristic approach. It implements strategy based on adding the attributes to the core. It is useful only when other methods fail.

The last option of ROSE concerning reducts is manual generation of reducts. The set of attributes is presented to the user together with possible increase or decrease of quality of classification and he/she can decide which attributes to add or to remove from a set. This approach is meant especially for experts who have also background knowledge about the meaning and possible coalitions of attributes.

Of course, there is an option to generate the core of attributes. Due to its properties it can be found in linear time.

8 Rule induction

The knowledge contained in the analyzed data set can be expressed in form of decision rules, i.e. *if* ... *then* ... statements. A decision rule consists of a condition part – conjunction of elementary tests on attribute values, and a decision part – an assignment to one or more decision classes.

One should note, that the induction of decision rules is a stand-alone problem, which can be considered independently of the Rough Sets theory. Rule induction algorithms generate rules for a given set of objects. In the simplest case such set consists of all objects from a given decision class. The Rough Sets methodology is useful if the data set is inconsistent and objects described by the same values of condition attributes belong to different decision classes. In such situation decision rules can be generated from approximations or from boundaries of decision classes.

The user can choose one of three schemes of rule induction [15]:

1. Minimal description.
   The resulting description is a minimal set of rules (i.e. the smallest set of rules) that cover all objects from the given set (a rule covers an object when all conditions in the rule's condition part are true for object's attribute values).
2. Satisfactory description.
   The resulting description contains only rules that satisfy requirements specified by the user (e.g. rules that are strong enough or that have good discriminating capabilities).
3. Exhaustive description.
   The generated description contains all possible rules that can be induced from the given set of objects.

8.1 Minimal description

The minimal description is generated by the LEM2 algorithm [3].

Depending on the definition of the set of objects, for which rules are generated, LEM2 induces two types of rules:

- exact rules are generated for the set of objects defined as the lower approximation of a given decision class,
- approximate rules are generated for the set of objects defined as a boundary of a given decision class (a difference between a lower and an upper approximation of the class).

Beside the original LEM2 algorithm, ROSE contains two modified versions:

- the LEM2 algorithm with interval extension,
- the LEM2 with similarity extension [4].

8.2 Satisfactory description

Within this scheme ROSE contains the implementation of Explore algorithm [5]. This algorithm is based on the breadth-first search strategy. It starts the generation with the shortest rules (containing one condition in their conditional part), and then gradually increases the length of generated rules. The search space is limited by thresholds defined by the user:

1. Maximal rule length – i.e. the maximal number of conditions in a condition part of the rule.
2. Minimal rule strength – i.e. the minimal number of objects covered by the rule, that belong to the decision class pointed by the rule.
3. Minimal discrimination level – i.e. the ratio of the number of objects covered by the rule, that belong to the class pointed by the rule to the number of all objects covered by the rule.

8.3 Exhaustive description

The Explore algorithm is also able to generate the exhaustive description consisting of all possible rules for the given set of objects. To achieve this, the maximal length should be equal to the number of attributes and the minimal strength should be set to 1. One should stress however, that generation of all rules can be extremely time and memory consuming, even for data sets of medium size.

9 Classification

In this phase, the decision rules generated in the previous phase are used for classifying objects (i.e. assigning them to decision classes). The assignment process is performed in the following steps:

1. if an object is covered by exactly one rule, then the object is assigned to the decision class pointed by this rule,
2. if there are several rules covering an object, then the conflict is resolved by assigning the object to the class with the highest number of votes or to the class pointed by this of the considered rules, that has the highest value of Laplace correction; the proper conflict resolution strategy is chosen by the user,
3. if no rule covers an object, then the object is assigned to the decision class pointed by the nearest rule, i.e. nearest according to the selected distance metric (the Lp-metric or valued closeness relation [11]). If there are several rules with the minimal distance to the object, the the same conflict resolution strategy as described in step 2 is applied.

The classifier implemented in ROSE can be used for two different tasks:

1. Classification of a new object, which membership to decision classes is unknown.

2. Classification of an already classified object (so called reclassification).

The first of the mentioned tasks is performed in the interactive way for single objects. After the classification, the user is presented the assigned decision class and the detailed information considered during classification process.

The second task is a single step of a reclassification test. The results are presented to the user after performing the whole test ROSE offers two scenarios of such tests:

- Living-one-out – suggested to be used in case of small data sets (smaller than 100 objects).
- K-fold cross-validation – intended to be applied in case of larger data sets. The user can choose random or stratified division into folds (in the latter division the distribution of number of objects from decision classes in each fold is the same as in the whole data set).

After the test a detailed statistical information about the classification accuracy (i.e. average values, standard deviations, distribution in decision classes) is presented. It is supplemented by a confusion matrix.

10 Availability

Demonstration version of ROSE is available on our WWW server at address: http://www-idss.cs.put.poznan.pl/rose. It is limited to information systems, that contain less then 200 objects and 5 attributes. Otherwise it is fully functional, so users can try it on several popular data sets available in the scientific community.

11 Summary

We have presented the process of data exploration based on rough set theory using ROSE software system. ROSE is 32-bit application implementing classical rough set theory as well as new extensions based on variable precision model and similarity relation. Further development of ROSE is in progress.

12 Acknowledgments

Authors of this paper would like to acknowledge the financial support from State Committee for Scientific Research, KBN research grant no. 8T11C 013 13 and from CRIT 2 - Esprit Project no. 20288. We would also like to thank Prof. Słowiński for his support and remarks to the earlier version of this paper and J. Stefanowski, R. Susmaga and R. Mieńko for their cooperation in development of ROSE system.

References

1. U.M. Fayyad, K.B. Irani. On the Handling of Continuous-Valued Attributes in Decision Tree Generation, Machine Learning, Vol 8, 1992, 87-102.
2. S. Greco, B. Matarazzo, R. Slowinski: A new rough set approach to multicriteria and multiattribute classification. [In] L. Polkowski, A. Skowron. (eds.), Proc. of the First Internat. Conference on Rough Setc and Current Trends In Computing - RSCTS'98, Warsaw, Springer-Verlag, 1998, 60-67.
3. J.W. Grzymala-Busse. LERS - a system for learning from examples based on rough sets. In R. Slowinski, (ed.) Intelligent Decision Support, Kluwer Academic Publishers, 1992, 3-18.
4. K. Krawiec, R. Slowinski, D. Vanderpooten. Learning of decision rules from similarity based rough approximations, [In] A. Skowron, L. Polkowski (eds.), Rough Sets in Knowledge Discovery vol. 2, Physica Verlag, Heidelberg, 1998, 37-54.
5. R. Mienko, J. Stefanowski, K. Tuomi, D. Vanderpooten. Discovery-Oriented Induction of Decision Rules. Cahier du Lamsade no. 141, Paris, Univeriste de Paris Dauphine, spetembre 1996.
6. Z. Pawlak - Rough sets. Int. J. Computer and Information Sci., 11, 1982, 341-356.
7. S. Romanski. Operation on families of sets for exhaustive search, given a monotonic function. In W. Beeri, C. Schmidt, N. Doyle (eds.), Proceedings of the 3rd Int. Conference on Data and Knowledge Bases, Jerusalem 1988, 310-322.
8. A. Skowron, Rauszer C. The discernibility matrices and functions in information systems in: Slowinski R. (ed.) Intelligent Decision Support. Handbook of Applications and Advances of the Rough Sets Theory. Kluwer Academic Publishers, 1992, 331-362.
9. R. Slowinski. Rough sets learning of preferential attitude in multi-criteria decision making. In Komorowski J., Ras Z.W. (eds.), Proc. of Int. Symp. on Methodologies for Intelligent Systems, Springer Verlag LNAI 689, 1993, 642-651.
10. R. Slowinski, J. Stefanowski. 'RoughDAS' and 'RoughClass' software implementations of the rough set approach. In R. Slowinski (ed.) Intelligent Decision Support. Handbook of Applications and Advances of the Rough Sets Theory. Kluwer Academic Publishers, 1992, 445-456.
11. R. Slowinski, J. Stefanowski. Rough classification with valued closeness relation. [In] E. Diday, Y. Lechavalier, M. Schrader, P. Bertrand, B. Burtschy (eds.), New Approaches in Classification and Data Analysis. Springer-Verlag, Berlin, 1994, 482-489.
12. R. Slowinski, J. Stefanowski. Rough set reasoning about uncertain data. Fundamenta Informaticae, 27 (2-3), 1996, 229-244.
13. R. Slowinski, D. Vanderpooten: Similarity relation as a basis for rough approximations [In] P.P. Wang (ed.): Advances in Machine Intelligence & Soft-Computing. Bookwrights, Raleigh, NC, 1997, 17-33.
14. R. Slowinski, D. Vanderpooten: A generalized definition of rough approximations based on similarity. IEEE Transactions on Data and Knowledge Engineering (to appear).
15. J. Stefanowski. On rough set based approaches to induction of decision rules. [In] A. Skowron, L. Polkowski (eds.), Rough Sets in Knowledge Discovery Vol. 1, Physica Verlag, Heidelberg, 1998, 500-529
16. W. Ziarko. Analysis of Uncertain Information in The Framework of Variable Precision Rough Sets. Foundations of Computing And Decision Sciences Vol 18 (1993) No. 3-4, 381-396.