More Rough Sets

Various Reducts and Rough Sets Applications

Rough Sets - Reducts

Algorithms for computing reducts or reduct approximations are discussed following. Note that any attribute subset is in this context considered to be an approximation to a reduct.

Input to a Reducer algorithm is a decision table, and a set of reducts is returned. The returned reduct set may possibly have a set of rules attached to it. A reduct is a collection of attribute indices into the table to which the reduct belongs.

Two main types of discernibility are considered and both these types can be computed modulo the decision attribute or not:

- Full: Computes reducts relative to the system as a whole, i.e., minimal attribute subsets that preserve our ability to discern all relevant objects from each other.

Rough Sets - Reducts

Object: Computes reducts relative to a fixed object, i.e., minimal attribute subsets that preserve our ability to discern that object from the other relevant objects. Generally, instead of fixing a single object x, we select a subset X of U, and process each x 2 X sequentially. That is, we first compute the minimal attribute subsets that discern the first object in X from all other relevant objects in U, before proceeding to compute the minimal attribute subsets that discern the second object in X from all other relevant objects in U, etc.

TIP If the reducts are relative to an object, rules or patterns are computed on the fly as well for reasons of efficiency.

Rough Sets - Reducts

Option	Subset
All	X = U
Index	X = {x}
Value	X = {x ∈ U | a(x) = v}
File	X = {x ∈ U | x is listed in a file}

Table 1: Options for selecting subsets of U.

For reducts relative to an object, the set X can be selected in different ways, as shown in Table 1.

Rough Sets - Reducts

A table can either be interpreted as a decision system or as a general Pawlak information system. If the option to compute reducts modulo the decision attribute is desired, the table is interpreted as a decision system. If the decision system contains inconsistencies, boundary region thinning [*] should be considered.

Since a reduct is a prime implicant of a discernibility function, algorithms for computing reducts can be used for more general Boolean reasoning, too. See Øhrn [**]

*W. Ziarko. Variable precision rough set model. Journal of Computer and System Sciences, 46:39–59, 1993.
**A. Øhrn. Cracking a logical puzzle with ROSETTA. Technical report, Knowledge Systems Group, Department
•of Computer and Information Science, NTNU, Trondheim, Norway, Dec. 1999.

Rough Sets - Reducts - QuickReduct(C,D).

C: the set of all conditional attributes; D: the set of decision attributes.

1) $R \leftarrow \{\}$
2) **do**
3) $T \leftarrow R$
4) $\forall x \in (\mathrm{C} - R)$
5) **if** $\gamma_{R \cup \{x\}}(\mathrm{D}) > \gamma_T(\mathrm{D})$
6) $\quad T \leftarrow R \cup \{x\}$
7) $R \leftarrow T$
8) **until** $\gamma_R(\mathrm{D}) == \gamma_{\mathrm{C}}(\mathrm{D})$
9) **return** R

The QuickReduct algorithm attempts to calculate reducts for a decision problem without exhaustively generating all possible subsets.

It starts off with an empty set and adds in turn, one at a time, those attributes that result in the greatest increase in the rough set dependency metric, until this produces its maximum possible value for the dataset.

Rough Sets - Reducts - Genetic Algorithm

The genetic algorithm for computing minimal hitting sets, is described by Vinterbo and Øhrn. The algorithm has support for both cost information and approximate solutions.

The algorithm's fitness function *f* is defined below, where S is the set of sets corresponding to the discernibility function*. The parameter α defines a weighting between subset cost and hitting fraction, while ε is relevant in the case of approximate solutions.

$$f(B) = (1-\alpha) \times [\mathrm{cost}(A)-\mathrm{cost}(B)]/\mathrm{cost}(A) +$$
$$\alpha \times \min\{\varepsilon_f\ (|[S \text{ in } S \mid S \cap B \neq \Theta]| \ / \ |S|)\}$$

*See Øhrn {A. Øhrn. Discernibility and Rough Sets in Medicine: Tools and Applications. PhD thesis, Norwegian University of Science and Technology, Department of Computer and Information Science, Dec. 1999. NTNU

•report 1999:133. [http://www.idi.ntnu.no/~aleks/thesis/]} [26, pages 52–55] for details. The expression for the hitting fraction in the definition of f is here somewhat simplified. In reality, we associate a weight w(S) with each S is S.

Rough Sets - Reducts - Genetic Algorithm (hitting sets)

A hitting set of a given bag or multiset* S of elements from 2^A is a set $B \subseteq A$ such that the intersection between B and every set in S is non-empty. The set $B \in HS(S)$ is a *minimal hitting set* of S if B ceases to be a hitting set if any of its elements are removed. Let HS(S) and MHS(S) denote the sets of hitting sets and minimal hitting sets, respectively.

$$HS(S) = \{B \subseteq A \mid B \cap S_i \neq \Theta \text{ for all } S_i \text{ in } S\}$$

*A bag or a multiset is conceptually an unordered collection of elements where the same element may occur more than once. Mathematically, therefore, it is common to define a multiset through a mapping from the element domain into the set of natural numbers, with the mapping defining the occurrence count. Here notation will be abused slightly and set-like syntax will in places be employed for convenience, even though duplicates are allowed. The text should make it clear whether we are dealing with sets or multisets. For additional clarity, a list-like notation with square brackets will be adopted for multisets in lieu of curly braces.

Rough Sets - Reducts - Genetic Algorithm (hitting sets cont)

The problem of computing prime implicants is easily transformed into the problem of computing minimal hitting sets.

A hitting set of S(*h*) defines an implicant of *h*, and subsequently, a minimal hitting set corresponds to a prime implicant. Relating this connection to reducts, we thus have the following relationships:

$B \in RED(A) \Leftrightarrow B \in MHS(S(g_A(U)))$

$B \in RED(A,x) \Leftrightarrow B \in MHS(S(f_A(x)))$

Rough Sets - Reducts - Genetic Algorithm

The subsets B of A that are found through the evolutionary search driven by the fitness function and that are "good enough" hitting sets, i.e., have a hitting fraction of at least ε, are collected in a "keep list".

The function cost specifies the cost of an attribute subset. If no cost information is used, a default unit cost defining cost(B) = |B| is used.

Approximate solutions are controlled through two parameters, ε and *k*. ε signifies a minimal value for the hitting fraction, while *k* denotes the number of extra keep lists in use by the algorithm. If $k = 0$, then only minimal hitting sets with a hitting fraction of approximately ε are returned. If $k > 0$, then $k+1$ groups of minimal hitting sets are returned, each group having an approximate (but not smaller) hitting fraction evenly spaced between ε and 1. $\varepsilon = 1$ implies minimal hitting sets.

Rough Sets - Reducts - Genetic Algorithm

Each reduct in the returned reduct set has a support count associated with it. The support count is a measure of the “strength” of the reduct, and may interpreted differently according to which algorithm that produced the reduct. For reducts computed with this genetic algorithm, the support count equals the reduct’s hitting fraction.

Rough Sets - Reducts - Johnson's Algorithm

Invokes a variation of a simple greedy algorithm to compute a single reduct only, as described by Johnson [D. S. Johnson. Approximation algorithms for combinatorial problems. Journal of Computer and System Sciences, 9:256–278, 1974.]. The algorithm has a natural bias towards finding a single prime implicant of minimal length.

The reduct B is found by executing the algorithm outlined below, where S denotes the set of sets corresponding to the discernibility function, and w(S) denotes a weight for set S in S that automagically gets computed from the data.

A greedy algorithm is a "single-minded" algorithm that gobbles up all of its favorites first. The greedy algorithm performs a single procedure over and over until it can't be done any more. It may not completely solve the problem, or, if it produces a solution, it may not be the very best one, but it *is* one way of approaching the problem and sometimes yields very good (or even the best) results.

Rough Sets - Reducts - Johnson's Algorithm

1. Let B = Θ.
2. Let *a* denote the attribute that maximizes $\sum$ w(S), where the sum is taken over all sets S in S that contain *a*. Ties are resolved arbitrarily.
3. Add *a* to B.
4. Remove all sets S from S that contain *a*.
5. If S = Θ return B. Otherwise, goto step 2.

Support for computing approximate solutions is provided by aborting the loop when "enough" sets have been removed from S, instead of requiring that S has to be fully emptied.

Rough Sets - Reducts - Johnson's Algorithm - Example

Let S = {{cat, dog, fish}, {cat, man}, {dog, man}, {cat, fish}} and let for simplicity w be the constant function that assigns 1 to all sets S in S. Step 2 in the algorithm then amounts to selecting the attribute that occurs in the most sets in S.

Initially, B = Θ. Since cat is the most frequently occurring attribute in S, we update B to include cat. We then remove all sets from S that contain cat, and obtain S = {{dog, man}}. Repeating the process, we arrive at a tie in the occurrence counts of dog and man, and arbitrarily select dog. We add dog to B, and remove all sets from S that contain dog. Now, S = Θ, so we're done. Our computed answer is thus B = {cat, dog}.

Rough Sets - Reducts - Holte's Algorithm

Returns all singleton attribute sets, inspired by the paper of Holte [*]. The set of all rules, i.e., univariate decision rules, are indirectly returned as a child of the returned set of singleton reducts.

*R. C. Holte. Very simple classification rules perform well on most commonly used datasets. Machine Learning, 11(1):63–91, Apr. 1993.

Rough Sets - RSES and Rosetta Implementations

RSES is a free software system for data exploration, classification support and knowledge discovery. The main functionalities of this software system are presented along with a brief explanation of the algorithmic methods used by RSES. Many of the RSES methods have originated from rough set theory introduced by Pawlak during the early 1980s.

ROSETTA is a rough set theory toolkit for analyzing tabular data. It is designed to support data mining and knowledge discovery: from data preprocessing, via computation of minimal attribute sets and generation of if-then rules or descriptive patterns, to validation and analysis of the induced rules or patterns. ROSETTA is intended as a general-purpose tool for discernibility-based modelling, and is not geared specifically towards any particular application domain.

Rough Sets – Applications - One

User-Centric Personalization to Predict User Purchases Based on the Discovery of Important Association Rules Using Rough Set Data Analysis

Abstract. In this paper, we present a model to extract important rules from user browsing history in an online purchasing database that makes use of user-centric data. Users' behaviours across all web sites visited is gathered into a database. This database is then mined for important association rules in order to predict the potential online buyers for certain products. Our research includes a method for constructing features to reflect online purchases based on the user-centric data collected from across multiple websites. It also introduces a new Rule Importance Measure based on the rough sets theory that provides an objective determination of the most appropriate rules to employ for the prediction task. Through experiments using a user-centric clickstream dataset from an online audience measurement company (showing customer online search experiences on search engines and shopping sites), we demonstrate how the Rule Importance Measure can be well adapted to predict online product purchases. In particular, we are able to isolate those user-centric features that are most important for predicting online purchases.

Rough Sets – Applications - Two

Discernibility and Rough Sets in Medicine: Tools and Applications

Abstract. This thesis examines how discernibility-based methods can be equipped to posses several qualities that are needed for analyzing tabular medical data, and how these models can be evaluated according to current standard measures used in the health sciences. To this end, tools have been developed that make this possible, and some novel medical applications have been devised in which the tools are put to use.

Rough set theory provides a framework in which discernibility-based methods can be formulated and interpreted, and also forms an appealing foundation for data mining and knowledge discovery. When the medical domain is targeted, several factors become important. This thesis examines some of these factors, and holds them up to the current state-of-the-art in discernibility-based empirical modeling. Bringing together pertinent techniques, suitable adaptations of relevant theory for model construction and assessment are presented. Rough set classifiers are brought together with ROC analysis, and it is outlined how attribute costs and semantics can enter the modeling process

Rough Sets – Applications - Three

Application of Clustering for Feature Selection Based on Rough Set Theory Approach

Abstract. Unsupervised clustering is an essential technique in Datamining. Since feature selection is a valuable technique in data analysis for information preserving data reduction, researchers have made use of the rough set theory to construct reducts by which the unsupervised clustering is changed into the supervised reduct. Rule identification involves the application of Datamining techniques to derive usage patterns from the information system. Knowledge extraction from data is the key to success in many fields. Knowledge extraction techniques and tools can assist humans in analyzing mountains of data and to turn the information contained in the data into successful decision making. This paper proposes, to consider an information system without any decision attribute. The proposal is useful when we get data, which contains only input information (condition attributes) but without decision (class attribute). K-Means algorithm is applied to cluster the given information system for different values of K. Decision table could be formulated using this clustered data as the decision variable. Then Quickreduct and VPRS algorithms are applied for selecting features. Ultimately, Rule Algorithm is used for obtaining optimum rules. The experiments are carried out on data sets of UCI machine learning repository and the HIV data set to analyze the performance study.

Rough Sets – Applications - Four

A foundation of rough sets theoretical and computational hybrid intelligent system for survival analysis

Abstract. What do we (not) know about the association between diabetes and survival time? Our study offers an alternative mathematical framework based on rough sets to analyze medical data and provide epidemiology survival analysis with risk factor diabetes. We experiment on three data sets: geriatric, melanoma and Primary Biliary Cirrhosis. A case study reports from 8547 geriatric Canadian patients at the Dalhousie Medical School. Notification status (dead or alive) is treated as the censor attribute and the time lived is treated as the survival time.

The analysis result illustrates diabetes is a very significant risk factor to survival time in our geriatric patients data. This paper offers both theoretical and practical guidelines in the construction of a rough sets hybrid intelligent system, for the analysis of real world data. Furthermore, we discuss the potential of rough sets, artificial neural networks (ANNs) and frailty index in predicting survival tendency.

Rough Sets – Applications - Five

A NOTE ON ROUGH SET THEORY APPLICATIONS IN POWER ENGINEERING

Abstract. Rough Set theory, proposed by Pawlak in 1982, has proved to be an adequate technique in imperfect data analysis, which has found interesting extensions and various applications. It can be regarded as complementary to other theories that deal with imperfect knowledge, such as or fuzzy sets or Bayesian inference. The paper presents some Rough Set theory applications in electrical power engineering.

Using the data taken from a power system control center, the authors suggested a systematic transformation of an extensive set of examples into a concise set of rules. RS theory is used in order to classify the current state of the power system in one of the three categories: normal (S), abnormal (U1) and restorative (U2).

Rough Sets – Example (cont)

It is possible for the core to be empty, which means that there is no indispensable attribute. Any single attribute in the information system can be deleted without altering the equivalence class structure. In such cases, there is no *essential* or necessary attribute which is required for the class structure to be represented.

Concluding Remarks

A PSYCHOLOGICAL TIP

Whenever you're called on to make up your mind,
and you're hampered by not having any,
the best way to solve the dilemma, you'll find,
is simply by spinning a penny.
No -- not so that chance shall decide the affair
while you're passively standing there moping;
but the moment the penny is up in the air,
you suddenly know what you're hoping