Rough Set Approach to Learning in MAS

COMPIÈGNE, FRANCE • SEPTEMBER 19, 2005

Nguyen Hung Son

Warsaw University, Poland

Sponsors

In Conjunction with the
2005 IEEE/WIC/ACM International Conference on
Web Intelligence and Intelligent Agent Technology

ISBN 2-913923-22-4
ROUGH SET APPROACH TO LEARNING IN MULTI AGENT SYSTEMS

Nguyen Hung Son

Institute of Mathematics, Warsaw University, Poland
email: son@mimuw.edu.pl

2005 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology Compiègne, France — September 19, 2005

OUTLINE

1. MOTIVATION
   - Learning in Multi Agent Systems
   - Rough Sets and Approximate Reasoning Problem

2. Rough Set theory
   - Introduction
   - Rough approximation of concepts

3. Rough set approach to learning
   - RS methods in learning
   - Rough sets vs. other learning methods

4. Applications of RS in MAS
   - Rough sets in Conflict Resolution
   - Rough set approach to approximate reasoning in MAS
   - Case Studies
Characteristics of MAS

- each agent has incomplete information or capabilities for solving the problem and, thus, has a limited viewpoint;
- there is no system global control;
- data are decentralized;
- computation is asynchronous.

K. Sycara,
Multiagent Systems,

Learning in MAS

- Learning from and adapting to novel environments are the most important skills of autonomous intelligent agents;
- The necessary of learning occurs in many problems in MAS, e.g.,
  - Cooperation;
  - Communication, negotiation;
  - Distributed planning;
  - Conflict recognizing and resolving;
  - Modeling other agents;
  - Coalition formation;
  - etc.
The environment effectively changes as other agents learn. Learning a model of other agents does not necessarily improve the behavior of an agent:
- Action taken by the learning agent can strongly bias which range of behaviors is encountered.
- Other agents' actions are often not directly observable,
- Example: Robocup;

**OUTLINE**

1. **Motivation**
   - Learning in Multi Agent Systems
   - Rough Sets and Approximate Reasoning Problem

2. Rough Set theory
   - Introduction
   - Rough approximation of concepts

3. Rough set approach to learning
   - RS methods in learning
   - Rough sets vs. other learning methods

4. Applications of RS in MAS
   - Rough sets in Conflict Resolution
   - Rough set approach to approximate reasoning in MAS
   - Case Studies
ROUGH SETS

- Approximating concepts.
- Identifying attribute dependencies.
- Reducing the problem size.
- Modification of attributes.
- Construction of decision rules.
- Knowledge discovery and representation.
- Model fusion.
- Approximate reasoning under uncertainty.

THE NEED FOR APPROXIMATE REASONING IN AGENT SYSTEMS

Assume that there are
- Two agents $A_1$ and $A_2$;
- They are talking about objects from a common universe $U$;
- They use different languages $\mathcal{L}_1$ and $\mathcal{L}_2$;

*any formula $\psi$ in $\mathcal{L}_1$ (and in $\mathcal{L}_2$) can be interpreted as a subset $C_\psi$ of objects from $U$.*

Each agent, who wants to understand the other, should perform
- an approximation of concepts used by the other;
- an approximation of reasoning scheme, e.g., derivation laws;
An universe of keys

**Teacher**

$L_1 = \{\text{keyboard, \ldots}\}$

**Learner**

$L_2 = \{\text{black, brown, white, metal, plastic, \ldots}\}$

---

**Complicated Concept Approximation Problems**

**UAV: Unmanned Aerial Vehicle**

- **Universe**: all possible manoeuvres of cars on the road;
- **Agent 1**: Road traffic expert: “dangerous situation”, “traffic jam” ... 
- **Agent 2**: UAV: pictures and movies from camera, meteorological information, sensor data like: brightness, humidity, ...

**Search Engines**

- **Universe**: internet resources;
- **Agent 1**: The searcher: interesting documents that satisfy query $q$
- **Agent 2**: Search engine: approximation of the concept by: ranking, clustering, dialog, ...
**Hardness of Approximation**

**Why the Concept Approximation Problem is Hard?**

- **Learnability of the target concept:** some concepts are too complex and cannot be approximated directly from feature value vectors.
  - PAC algorithms;
  - Effective learnability of some concept spaces;
  - VC dimension, ...
- **Time and space complexity:** Many problems related to optimal approximation are NP-hard.

**Existing Approximate Reasoning Methods**

- Statistical approaches: multiple regression, Principal Components Analysis and Factor Analysis, Discriminant Analysis, Cluster Analysis, ...
- Fuzzy set and fuzzy logic.
- Rough set theory.
- And many, many others.
ROUGH SET THEORY

- Introduction
- Rough approximation of concepts

Rough set approach to learning
- RS methods in learning
- Rough sets vs. other learning methods

Applications of RS in MAS
- Rough sets in Conflict Resolution
- Rough set approach to approximate reasoning in MAS
- Case Studies

ROUGH SETS’ HISTORY IN A NUTSHELL

- Around 1982 - the beginning - publication of several papers by Zdzisław Pawlak including:
- 1991 - Publication of the book:
- 1994 - International Rough Set Society (IRSS) www.roughsets.org
- 2004 - Transactions on Rough Sets - a journal subline in Lecture Notes in Computer Science, Springer-Verlag
ROUGH SETS – MOTIVATION

We are given a set of data representing the task at hand. This data may be (and usually is) incomplete, incoherent, vague, discrepant, partial, ... Can we:

- extract a suitable model,
- identify the most important elements of data,
- find relationships,
- select best attributes,
- incorporate experts’ knowledge?

ROUGH SETS – BASIC STEPS

- Approximating concepts.
- Identifying attribute dependencies.
- Reducing the problem size.
- Modification of attributes.
- Construction of decision rules.
- Knowledge discovery and representation.
- Model fusion.
- Approximate reasoning under uncertainty.
A Simple Example

<table>
<thead>
<tr>
<th>No.</th>
<th>Age</th>
<th>LEM</th>
<th>Walk</th>
</tr>
</thead>
<tbody>
<tr>
<td>u₁</td>
<td>16-30</td>
<td>50+</td>
<td>yes</td>
</tr>
<tr>
<td>u₂</td>
<td>16-30</td>
<td>0</td>
<td>no</td>
</tr>
<tr>
<td>u₃</td>
<td>31-45</td>
<td>1-25</td>
<td>no</td>
</tr>
<tr>
<td>u₄</td>
<td>31-45</td>
<td>1-25</td>
<td>yes</td>
</tr>
<tr>
<td>u₅</td>
<td>46-60</td>
<td>26-49</td>
<td>no</td>
</tr>
<tr>
<td>u₆</td>
<td>16-30</td>
<td>26-49</td>
<td>yes</td>
</tr>
<tr>
<td>u₇</td>
<td>46-60</td>
<td>26-49</td>
<td>no</td>
</tr>
</tbody>
</table>

In rough sets we call such a rectangular data table an information system. If the decision attribute is distinguished, we name it decision table.
**Example Continued**

In our example the information is not precise (complete) enough. We cannot discern precisely walking from non-walking patients. So, the concepts of walking and non-walking patient are partly *indiscernible*.

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1-25</th>
<th>26-49</th>
<th>50+</th>
</tr>
</thead>
<tbody>
<tr>
<td>16-30</td>
<td>u2</td>
<td>u6</td>
<td>u1</td>
<td></td>
</tr>
<tr>
<td>31-45</td>
<td>u3u4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>46-60</td>
<td></td>
<td>u5,u7</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Indiscernibility Classes**

A set of objects that are seemingly identical if we consider only features that are in our data we call an *indiscernibility class*. The decomposition into indiscernibility classes represents our knowledge about the structure of considered data in view of information carried by attributes.

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1-25</th>
<th>26-49</th>
<th>50+</th>
</tr>
</thead>
<tbody>
<tr>
<td>16-30</td>
<td>u2</td>
<td>u6</td>
<td>u1</td>
<td></td>
</tr>
<tr>
<td>31-45</td>
<td>u3u4</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>46-60</td>
<td></td>
<td>u5,u7</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
**Approximations**

- Lower approximation – we are sure that these objects are in the set (concept of walking patient).
- Upper approximation - it is possible (likely, feasible) that these objects belong to our set (concept). They *roughly* belong to the set.

![Diagram showing approximations with regions for different age groups and objects u1, u2, u3, u4, u5, u6, u7]

**Rough Sets**

- Definable or crisp sets (concepts) are sets for which each element (object) either belongs to the set or not.
- Rough sets are sets that contain elements which are imprecisely defined. We cannot be sure whether some objects belong to a given concept or to its complement. Lower approximation of a rough set is a crisp set.
- The difference between the upper and the lower approximation is usually called *boundary region*. It is the relative size of this region that decides how vague, imprecise, approximate, uncertain our considered concept (set) is.
**Information Systems and Decision Tables**

An information system $A$ is what we call in Machine Learning a (training) sample with no labels. Namely:

$$A = (U, A)$$

where:

- $U$ is a finite set (domain, universe) from which we draw examples.
- Elements of $U$ are called objects.
- $A$ is a finite set of attributes i.e. functions of the form $a : U \rightarrow V_a$.

**Decision table** is a special case of information system with distinguished decision attribute.

$$A = (U, A \cup \{d\})$$

where $d \notin A$

Elements of $A$ are called conditional attributes, $d$ is called decision.

---

**Rough Approximation Defined by Attributes**

- Given an information system $S = (U, A)$ and a concept $X \subset U$.
- Assume that only some attributes $B \subset A$ are available.
- We define $B$-indiscernibility relation by
  $$IND(B) = \{(x, y) \in U \times U : \inf_B(x) = \inf_B(y)\}.$$  
  This is an equivalence relation, let
  $$[x]_{IND(B)} = \{u \in U : (x, u) \in IND(B)\}.$$  
- The lower and upper approximation of $X$ by attributes from $B$ are defined by
  $$L_B(X) = \{x \in U : [x]_{IND(B)} \subseteq X\}$$
  $$U_B(X) = \{x \in U : [x]_{IND(B)} \cap X \neq \emptyset\}$$
- The rough membership function is defined by
  $$\mu^B_X(x) = \frac{\text{card}(X \cap [x]_{IND(B)})}{\text{card}([x]_{IND(B)})}.$$
Rough sets: Extensions

Rough approximation of the concept $C'$ (induced by a sample $X$):

any pair $P = (L, U)$ satisfying the following conditions:

- $L \subseteq U \subseteq U$;
- $L$ and $U$ are subsets of $U$ expressible in the language $L_2$;
- $L \cap X \subseteq C \cap X \subseteq U \cap X$;
- (*) the set $L$ is maximal (and $U$ is minimal) in the family of sets definable in $L$ satisfying (3).

Rough membership function of concept $C$:

any function $f : U \rightarrow [0, 1]$ such that the pair $(L_f, U_f)$, where

- $L_f = \{x \in U : f(x) = 1\}$ and
- $U_f = \{x \in U : f(x) > 0\}$.

is a rough approximation of $C$ (induced from sample $U$)
Rough sets: Extensions

- VPRS: Variable Precision Rough Sets (Ziarko);
- Generalized approximation space (Skowron);
- Rough Approximation Defined by Classifiers;

Rough Approximation Defined by Rules

$$w_{yes} = \sum_{r \in R_{yes}} strength(r)$$
$$w_{no} = \sum_{r \in R_{no}} strength(r)$$

$$\mu_C(x) = \begin{cases} 
\text{undetermined} & \text{if } \max(w_{yes}, w_{no}) < \omega \\
0 & \text{if } w_{no} \geq \max(\omega, w_{yes} + \theta) \\
1 & \text{if } w_{yes} \geq \max(\omega, w_{no} + \theta) \\
\frac{\theta + (w_{yes} - w_{no})}{2\theta} & \text{in other cases}
\end{cases}$$
ROUGH SET METHODOLOGY

- Rough set methodology is based on indiscernibility between objects.
- Rough set methods utilize the comparison between elements, e.g.,
  - discernibility,
  - indiscernibility,
  - similarity, ...
- Many issues in rough set theory may be interpreted in terms of Boolean reasoning.
- Rough set methodology can be applied to:
  - implement efficient methods for mining interesting templates from data: data reduction, minimal decision rules, decomposition, hierarchical learning ...
  - cooperate and improve existing methods like decision trees, association rules, clustering, kNN, neural networks, Bayesian networks...
ROUGH SET TECHNIQUES IN LEARNING

Over the years work of several “rough setters” resulted in creation of algorithmic methods for analysis of various kinds of data. These methods utilize fundamental rough set notions such as discernibility, approximation, information function and reduct. These methods are dealing with:

- Reduction of data size and complexity.
- Discovery of frequent patterns and decision rule discovery.
- Continuous attributes’ discretization.
- Data decomposition.
- Others - including new feature construction, instance-based learning, ...

REDUCTION

- Do we need all attributes?
- Do we need to store the entire data?
- Is it possible to avoid a costly test?

Reducts are subsets of attributes that preserve the same amount of information. They are, however, (NP-)hard to find.

- Efficient and robust heuristics exist for reduct construction task.
- Searching for reducts may be done efficiently with the use of evolutionary computation.
- Overfitting can be avoided by considering several reducts, pruning rules and lessening discernibility constraints.

For more refer to: [Pawlak Book], [RS tutorial], [RS algorithms]
**Data Reduction in Rough Sets**

**What is a reduct?**

Reducts are minimal subsets of attributes which contain a necessary portion of information of the set of all attributes.

- Given an information system $S = (U, A)$ and a monotone evaluation function
  \[ \mu_S : \mathcal{P}(A) \rightarrow \mathbb{R}^+ \]

- The set $B \subset A$ is called \( \mu \)-reduct, if
  - $\mu(B) = \mu(A)$,
  - for any proper subset $B' \subset B$ we have $\mu(B') < \mu(B)$;

- The set $B \subset A$ is called approximated reduct, if
  - $\mu(B) \geq \mu(A) - \varepsilon$,
  - for any proper subset ...

**Some Types of Reducts**

- Information reduct:
  \[ \mu_1(B) = \text{number of pairs of objects discerned by } B \]

- Decision-oriented reduct:
  \[ \mu_2(B) = \text{number of pairs of conflict objects discerned by } B \]

- Object-oriented reduct:
  \[ \mu_x(B) = \text{number of objects discerned with } x \text{ by } B \]

- Frequent reducts;
- $\alpha$-reducts;
- ...
PATTERN AND RULE DISCOVERY

By examining the structure of indiscernibility classes and reducts one can summarize information carried by objects using patterns of the form:

\[(a_{i1} = v_1) \land \ldots \land (a_{ik} = v_k)\]

These patterns may be further converted into associations. In classification problems, we can produce decision rules of the form:

\[(a_{i1} = v_1) \land \ldots \land (a_{ik} = v_k) \Rightarrow d = v_d\]

Patterns and rules can be filtered, pruned, generalized, and composed. That permits management of discovered knowledge. For more refer to: [RS algorithms], [Nguyen S.H. Thesis], [LEM algorithm].

DISCRETIZATION

Attributes that have many different values, e.g., real-valued, may pose a technical problem for some algorithmic DM methods. Discretization (quantization) of attribute values can be done using rough set framework. We consider all pairs of objects. Then we consider all possible cuts on attributes in discourse. We choose the cut that induces best split w.r.t the number of objects from different decision classes that are discerned by this split. This is called Maximal Discernibility (MD) heuristic. Various modifications, especially concerning the choice of best cut, exist. For more refer to: [Nguyen H.S. Thesis], [DM discretization], [RS algorithms], [RS tutorial].
DECOMPOSITION

Large data sets may not be possible to process as-is. The ability to decompose data set into smaller chunks is a requirement. These fragments, after decomposition represented as leaves in decomposition tree, are supposed to be more uniform and easier to cope with decision-wise. For more refer to: [Nguyen S.H. Thesis],

ROUGH SET SOFTWARE

Several rough set algorithms are implemented in freely available Rough Set Exploration Sytem - RSES
RSES can be downloaded from:

http://logic.mimuw.edu.pl/~rses
### Rough Sets in Decision Tree Construction

**Main problem:**
Search for minimal decision tree compatible with a given decision table. This problem is NP-hard

**Heuristics:**
- Decision tree are reconstructed from a given set of candidate partitions;
- Best-first searching strategy, a quality measure must be defined, e.g.,
  - Entropy gain;
  - Gini’s index;
- Rough set based measure:
  - discernibility measure = number of resolved conflicts
- **In our recent research:** decision trees constructed by discernibility measure have many interesting properties.
**Rough Sets vs. k-NN**

- How to define the measure function or neighborhood?
- Similarity relation can be learned from data using rough set methods, see [Hoa Nguyen], [A. Wojna Thesis].
- A simple idea: the more reducts an attribute appears in the more important this attribute is.
- Filtering methods in rule based classifiers can simulate both decision tree and k-NN classifiers.

**Multivariate Analysis**

- Multiple Regression: analyzes the relationship between several attributes and a decision;
- Principal Components Analysis and Factor Analysis – A linear dimensionality reduction technique, which:
  - identifies orthogonal directions of maximum variance in the original data,
  - projects the data into a lower-dimensionality space formed of a sub-set of the highest-variance components.
- Discriminant Analysis: searching for the best set of attributes that discriminates objects from two or more decision classes.
- Cluster Analysis: grouping similar objects.
**MULTIVARIATE VS. ROUGH SETS**

- PCA and clustering methods can be applied as preprocessing step to rough set methods.
- e.g to extract new features from the original set of attributes.
- Experimental results: they can improve the quality of rough set classifiers;

---

**ROUGH SETS IN NETWORK CONSTRUCTION**

- A Bayesian network is an acyclic directed graph that models probabilistic dependencies between the domain variables:
  - Q: How to construct Bayesian networks from data?
    - In many cases, the problem is NP-hard.
  - A: Searching for structure + probability distribution;
  - RS: Structure can be reconstructed by calculating frequent reducts!
- Rough sets vs. neural networks:
- Rough sets vs. Petri net
- Rough sets vs. Belief (cause-and-effect) networks.
Pawlak's Conflict Model

**Example**

<table>
<thead>
<tr>
<th></th>
<th>a</th>
<th>b</th>
<th>c</th>
<th>d</th>
<th>e</th>
</tr>
</thead>
<tbody>
<tr>
<td>emp.</td>
<td>-1</td>
<td>0</td>
<td>-1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>TU1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>-1</td>
</tr>
<tr>
<td>TU2</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>-1</td>
<td>-1</td>
</tr>
</tbody>
</table>

\[ Con(a) = \frac{|X_a^+| \cdot |X_a^-|}{\left\lfloor \frac{n}{2} \right\rfloor \cdot (n - \left\lfloor \frac{n}{2} \right\rfloor)} \]

\[ Con(A) = \sum_{a \in A} \frac{Con(a)}{|A|} \]

- a — increasing the employees’ incomes,
- b — improving the work conditions,
- c — increasing the social care, warranty of the current level of employment,
- d — increasing the factory profit by reducing the costs of work (reductions in employment),
- e — increasing the level of investment to grow up the factory profit.
**Pawlak’s Conflict Model (cont.)**

- Distance between agents

\[
d(x, y) = \frac{\sum_{a \in A} \phi_a(x, y)}{|A|}
\]

- Most conflicting and least conflicting attributes
- Coalition formulation (confederation)

**Pawlak, Z. (1998).**

Reasoning

- Modal logics based methods (Vakarelov, Bannerjee, Düntsch, see [Demri and Orlowska])
- Approximation Tranducers [Doherty et. al.].
- Rough mereology (Polkowski, Skowron) [Mereology].
- and various other techniques.

Reasoning via Layered Learning

Given:
- $U$: the set of examples;
- $A$: the set of attributes;
- $H$: concept decomposition diagram;
- $D = dec_{C_1}, dec_{C_2}, ... dec_{C}$

Goal: For each concept $C$ in the hierarchy:
- construct a decision system $S_C$;
- induce a rough approximation of $C$, i.e., a rough membership functions for $C$: $[\mu_C^+(x), \mu_C^-(x)]$

System control: The system can be tuned by
- uncertainty parameters: $\theta$;
- learning parameters for each level.
\( S_C = (U, A_C, \text{dec}_C) \), where

\[ A_C = \{a_{C1}, \ldots, a_{Cn}\} \]

is a collection of rough approximations of subconcepts \( C_1, \ldots, C_n \):

- either \( a_{Cj} = [\mu_{j+}, \mu_{j-}] \);
- or \( a_{Cj} = [w_{yes}, w_{no}] \).
La yered learning algorithm

1: for \( l := 0 \) to \( \text{max\_level} \) do
2: for (any concept \( C_k \) at the level \( l \) in \( H \)) do
3: if \( l = 0 \) then
4: \( S_{C_k} := (U, A_{C_k}, \text{dec}C_k) \);
5: else
6: \( A_k := \bigcup O_k \); \( \)
7: \( S_{C_k} := (U, A_k, \text{dec}C_k) \);
8: end if
9: generate the rule set \( RULES(S_{C_k}) \) for decision table \( S_{C_k} \);
10: generate the output vector \( O_k = \{ w^{C_k}_{\text{yes}}, w^{C_k}_{\text{no}} \} \),
11: end for
12: end for

Outline

1 Motivation
   - Learning in Multi Agent Systems
   - Rough Sets and Approximate Reasoning Problem

2 Rough Set theory
   - Introduction
   - Rough approximation of concepts

3 Rough set approach to learning
   - RS methods in learning
   - Rough sets vs. other learning methods

4 Applications of RS in MAS
   - Rough sets in Conflict Resolution
   - Rough set approach to approximate reasoning in MAS
   - Case Studies
**Case Studies**

- Robocup;
- Search engines:

**References - Books**

- Zdzisław Pawlak
  *Rough Sets: theoretical aspects of reasoning about data.*

- Stephane P. Demri and Ewa S. Orłowska
  *Incomplete information: Structure, Inference, Complexity*
  Springer-Verlag, Heidelberg, 2002

- Lech Polkowski and Andrzej Skowron (eds.)
  *Rough Sets in Knowledge Discovery, part 1 and 2*
  Physica-Verlag, Heidelberg, 1998

- Sankar K. Pal, Lech Polkowski and Andrzej Skowron (eds.)
  *Rough-Neural Computing*
  Springer-Verlag, Heidelberg, 2004
REFERENCES - THESSES

Nguyen Hung Son,
Discretization of real value attributes. Boolean reasoning approach.
Ph. D. Thesis, Department of Math., Comp. Sci. and Mechanics,
Warsaw University, Warsaw, 1997

Nguyen Sinh Hoa Thi,
Data regularity analysis and applications in data mining.
Ph. D. Thesis, Department of Mathematics, Informatics. and Mechanics, Warsaw University, Warsaw, 1999

Arkadiusz Wojna.
Analogy-based Reasoning in Classifier Construction.
Ph. D. Thesis, Department of Mathematics, Informatics. and Mechanics, Warsaw University, Warsaw, 2004

REFERENCES - ARTICLES

Komorowski J., Pawłak Z., Polkowski L., Skowron A.
Rough Sets: A tutorial.

Bazan J., Nguyen H. S., Nguyen S. H., Synak P., and Wróblewski J.
Rough set algorithms in classification problem.

Nguyen, H.S., Nguyen, S.H.
Discretization Methods for Data Mining.

Nguyen H.S., Nguyen S.H.
Rough Sets and Association Rule Generation.
REFERENCES - ARTICLES CONT.

Grzymała-Busse, J.
A New Version of the Rule Induction System LERS.
Fundamenta Informaticae, Vol. 31(1), 1997, pp. 27–39

Nguyen, H.S., Nguyen, S.H.
Discretization Methods for Data Mining.

Doherty P., Łukaszewicz W., Skowron A., Szałas A.
Approximate Transducers and Trees: A technique for combining rough and crisp knowledge.

Polkowski P., Skowron A.
Rough mereology: A new paradigm for approximate reasoning.

REFERENCES - WWW RESOURCES

IRSS – International Rough Set Society
http://www.roughsets.org
Contains references, tutorial notes, and other materials

RSES – Rough Set Exploration System
http://logic.mimuw.edu.pl/~rses
Freely available software for rough set data analysis.

Rosetta – a Rough Set Toolkit for Analysis of Data
http://rosetta.lcb.uu.se/
Another publicly available RS software.

Transactions on Rough Sets
http://www.springeronline.com/sgw/cda/frontpage/0,11855,5-164-2-99627-0,00.html
A journal subline of Lecture Notes in Computer Science, Springer-Verlag.