Rough Set Analysis Tool

Pratanu Mandal
Student, B. Tech
Dept. of Computer Science & Engg.
JIS College of Engineering
Kalyani, India

Madhav Kumar Jha
Student, B. Tech
Dept. of Computer Science & Engg.
JIS College of Engineering
Kalyani, India

Apurba Paul
Asst. Professor
Dept. of Computer Science & Engg.
JIS College of Engineering
Kalyani, India

***Abstract* — In this paper the rudiments of rough set theory as described by Zdzisław I. Pawlak, will be discussed in brief. We shall also discuss how reducts can be found using rough sets. We shall then illustrate the need and features of the Rough Set Analysis Tool that we have developed using Python, the underlying technologies, and algorithms used. Finally, we shall provide a demonstration of the Rough Set Analysis Tool, and discuss our future plans regarding our tool.**

***Index Terms* — Rough set, analysis, tool, python, cross-platform, set theory, set approximations, discernibility matrix.**

I. Introduction

A rough set is a formal approximation of a crisp set. It was first described by Zdzisław I. Pawlak in 1991. While regular crisp sets use a deterministic approach, rough sets use a probabilistic approach. It provides a systematic framework for the study of the problems arising from imprecise and insufficient knowledge. It is typically used for finding the reduct and core and the attribute dependencies in a decision system and apply this knowledge for feature reduction and rule extraction. The set of rules extracted can be used for classifying new data in the future.

The main objective of this paper is to explore the tool that we have developed using Python for analyzing Rough Sets and compare it with pre-existing solutions. We shall also take a look at the basic concepts of Rough Set Theory which has been a subject of research for many years (Orlowska & Pawlak, 1984; Pawlak, 1984).

II. Rough Sets

A rough set is a formal approximation of a crisp set [1]. It is defined by a pair of sets which give the lower and upper approximation of the crisp set. Unlike crisp sets where the decision can only be in terms of discrete values, rough sets are probabilistic in nature.

III. Information and Decision Systems

An information system is a collection of attributes for more than one objects. It may be defined as a pair of (U, A) where U is the non-empty finite set of objects, A is non-empty finite set of attributes such that a: U → V_a for every a ∈ A, V_a is the value set of a [5].

An Information system augmented with a decision attribute is known as a decision system. Mathematically, it can be defined as DS: T = (U, A{d}) where d ∉ A is the decision attribute [6].

IV. Set Approximations

Formally, rough sets are defined in terms of set approximations.

A. Lower Approximation

Lower Approximation is a description of the domain objects that are known with certainty to belong to the subset of interest. The Lower Approximation Set of a set X, with regard to R is the set of all of objects, which certainly can be classified with X regarding R, that is, set B” [2].

B. Upper Approximation

Upper Approximation is a description of the objects that possibly belong to the subset of interest. The Upper Approximation Set of a set X regarding R is the set of all of objects which can be possibly classified with X regarding R, that is, set B* [2].

C. Boundary Region

Boundary Region is description of the objects that of a set X regarding R is the set of all the objects, which cannot be classified neither as X nor -X regarding R. If the boundary region is a set X = ∅ (Empty), then the set is considered "Crisp", that is, exact in relation to R; otherwise, if the boundary region is a set X ≠ ∅ (Empty) the set X "Rough" is considered. In that the boundary region is BR = B* - B” [2].

V. Rough Membership

Rough sets are also described with help of rough membership of individual elements. The membership of an object x to a rough set X with respect to knowledge in B is expressed as μ_X^B (x) [3]. Rough membership is similar but not identical, to fuzzy membership. It is defined as:

$$\mu_X^B(x) = \frac{|[x]_B \cap X|}{|[x]_B|}$$

Rough membership values lie within the range of 0 to 1.

VI. Reducts

Reducts are the sets of attributes that preserve the indiscernibility relation and, consequently, the set approximation [7]. There are usually several such subsets of attributes and those which are minimal are called reducts.

An information system I = (U, A), a reduct is a minimal set of attributes $B \subseteq A$ such that IND (B) = IND1(A).

A reduct with minimal cardinality is called a minimal reduct.

VII. Discernibility Matrix and Function

A. Discernibility Matrix

For an information system I = (U, A) with n objects, the discernibility matrix D is a symmetric n x n matrix where the (i, j)th element d_{ij} is given by $d_{ij} = \{a \in A \mid a(x_i) \neq a(x_j)\}$. Each entry of a discernibility matrix is one or more attributes for which the objects x_i and x_j differ.

B. Discernibility Function

With every discernibility matrix one can uniquely associate a discernibility function.

A discernibility function f_I for an information system I = (U, A) is a boolean function of n boolean variables $a_1, a_2, a_3, ..., a_n$ corresponding to the n number of attributes $A_1, A_2, A_3, ..., A_n$ such that

$$f_I(a_1, a_2, a_3, ..., a_n) = \{\vee d_{ij} \mid 1 \leq i \leq n,\ d_{ij}\}$$

Where, d_{ij} is the (i, j)th entry of the discernibility matrix.

The set of all prime implicants corresponds to the set of all reducts of I. Hence, our aim is to find the prime implicants of f_I.

A prime implicant is a minimal implicant (with respect to the number of its literals).

VIII. Algorithm for finding Minimal Reducts

We have used the exhaustive algorithm for finding minimal reducts in our application. We however have plans to implement more algorithms and also support plugins.

Procedure Find-Min-Reduct (U, A)

This is the exhaustive algorithm for finding minimal reducts [4].

```
/* Given an information system I = (U, A) where U is a non-empty set of objects and A is a non-empty set of attributes, to find the set of reducts of I, and thereby the minimal reducts of I. */

1.  Begin

2.  Construct the discernibility matrix D of I. Let $c_1, c_2, …, c_r$ be the sets of attributes corresponding to the non-empty cells of D.

3.  Arrange $c_1, c_2, …, c_r$ in non-decreasing order of their sizes. Let $C_1, C_2, …, C_r$ be the rearranged sets of attributes such that
        $|C_1| \leqslant |C_2| \leqslant \cdots \leqslant |C_r|$

4.  Let T = {} and S = $\{C_1, C_2, …, C_r\}$

5.  Repeat Steps 5, 6, 7, and 8 While $S \neq \emptyset$

6.  Let c be a minimal member of S, i.e., $|c| \leqslant |C_i|$ for all $C_i \in S$

7.  $T = T \cup \{c\}$

8.  For all $C_i \in S$, If $c \subseteq C_i$, Then remove $C_i$ from S, $S = S - \{C_i\}$

9.  Let $t_1, t_2, …, t_k$ be the members of T constructed through steps 4-7 above. For each $t_i \in T$ form a Boolean clause $T_i$ as the disjunction of the attributes in $t_i$. Construct the discernibility function $f_D$ as the conjunction of all $T_i$'s.

10. Simplify $f_D$ to sum-of-products form. Each product term corresponds to a reduct of the information system I. Any one of the product terms with minimal cardinality is a minimal reduct of I.

11. END-Find-Min-Reduct
```

IX. Pre-existing tools for Rough Set Analysis

A. RSES

RSES is a toolset for analyzing data with the use of methods coming from Rough Set Theory. It is a graphical, user-friendly front-end running under Windows NT/98/95/2000/XP and providing access to methods from RSESlib library. RSESlib is a core of RSES' computational kernel. The RSES GUI allows point-and-click operation for making Rough Set computations. Both library and GUI are designed and implemented at the Group of Logic, Institute of Mathematics, Warsaw University and the Group of Computer Science, Institute of Mathematics, University of Rzeszów, Poland.

B. ROSETTA

ROSETTA is a toolkit for analyzing tabular data within the framework of rough set theory. ROSETTA is designed to support the overall data mining and knowledge discovery process: From initial browsing and preprocessing of the data, 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.

C. Rough Sets [R Package]

It is a free package for R language that facilitates data analysis using techniques put forth by Rough Set and Fuzzy Rough Set Theories. It does not only provide implementations for basic concepts of RST and FRST but also popular algorithms that derive from those theories. The functionalities provided by the package include Discretization, Feature selection, Instance selection, Rule induction, and Classification based on nearest neighbors.

X. Need for a New Rough Set Analysis Tool

All these tools, including several less popular tools, are quite outdated. It has been quite some time since any of them had a new release. RSES is practically not supported anymore as stated on its website, and will not be receiving further updates.

Furthermore, ROSETTA is only available on the Windows Operating System. RSES supports both Windows and Linux. Neither of the applications supports Mac.

Rough Sets [R Package], although a powerful tool, with support for Windows, Linux and Mac, is only available as a command line tool. It has no support for a GUI.

Thus, we decided to build a cross-platform rough set analysis tool based on modern technologies such as Python. Although the tool we have developed is in its infancy, due to the dynamic nature of the technologies used, it will be much easier to extend the tool to include more powerful features. For instance, it is possible to incorporate a plugin support in the future, which will allow researchers to plug in their own algorithms for reduct finding.

Moreover, the extremely powerful "numpy" package in Python hardly has an analogue in other languages, and will certainly help boost productivity. Moreover, we have witnessed a global trend of a shift towards scripting languages like Python for data analysis. Thus, we are open to the huge number of packages for Machine Learning and Data Analysis already available in Python, and more are surely to come up. These can be used to further extend the capabilities of the tool with time.

XI. Rough Set Analyzer

It is a cross-platform rough set analysis tool that we have developed using modern technologies such as Python.

It is able to load datasets from MS Excel (.xls, .xlsx) files, and classify the datasets as rough sets, obtain accuracy of the classification, calculate set approximations, and find minimal reducts (along with discernibility matrix).

It is built using python 2.7, and we have used packages wxpython, xlrd, and numpy to build the various components of the tool.

XII. Demonstration

Test Case: Pima Indians Diabetes Data Set
No. of attributes: 9
No. of entries: 768

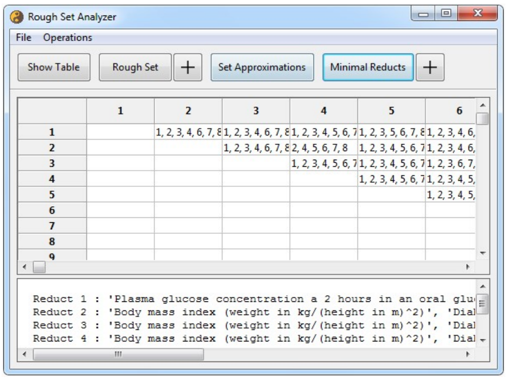
Training

No. of attributes: 9
No. of entries: 613

Testing

No. of attributes: 9
No. of entries: 155

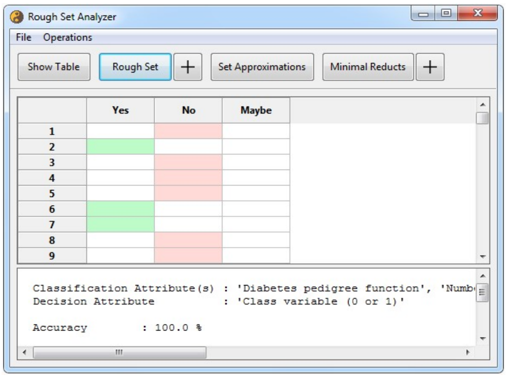
Accuracy: 100%



	1	2	3	4	5	6
1		1, 2, 3, 4, 6, 7, 8	1, 2, 3, 4, 6, 7, 8	1, 2, 3, 4, 5, 6, 7	1, 2, 3, 5, 6, 7, 8	1, 2, 3, 4, 6,
2			1, 2, 3, 4, 6, 7, 8	2, 4, 5, 6, 7, 8	1, 2, 3, 4, 5, 6, 7	1, 2, 3, 4, 6,
3				1, 2, 3, 4, 5, 6, 7	1, 2, 3, 4, 5, 6, 7	1, 2, 3, 6, 7,
4					1, 2, 3, 4, 5, 6, 7	1, 2, 3, 4, 5,
5						1, 2, 3, 4, 5,
6						
7						
8						
9						

```
Reduct 1 : 'Plasma glucose concentration a 2 hours in an oral glu
Reduct 2 : 'Body mass index (weight in kg/(height in m)^2)', 'Dia
Reduct 3 : 'Body mass index (weight in kg/(height in m)^2)', 'Dia
Reduct 4 : 'Body mass index (weight in kg/(height in m)^2)', 'Dia
```

Fig. 1. Demonstration training screenshot



	Yes	No	Maybe
1		🟥	
2	🟩		
3		🟥	
4		🟥	
5		🟥	
6	🟩		
7	🟩		
8		🟥	
9		🟥	

```
Classification Attribute(s) : 'Diabetes pedigree function', 'Numb
Decision Attribute          : 'Class variable (0 or 1)'

Accuracy        : 100.0 %
```

Fig. 2. Demonstration testing screenshot

XIII. Future Plans

Although the tool we have developed is in its infancy, due to the dynamic nature of the technologies used, it will be much easier to extend the tool to include more powerful features. For instance, it is possible to incorporate a plugin support in the future, which will allow researchers to plug in their own algorithms for reduct finding.

We also have plans to improve the underlying frameworks and algorithms to make the application more efficient and robust.

We intend to distribute the application with proper support for at least the three major operating systems – Windows, OS X, and Linux.

ACKNOWLEDGMENT

We would like to express our gratitude to Mr. Apurba Paul for providing effective guidance in development of this project work. His conscription of the topic and all the helpful hints, he provided, contributed greatly to successful development of this work, without being pedagogic and overbearing influence.

We also express our sincere gratitude to Mr. Amrut Ranjan Jena, HOD of the Department of Computer Science and Engineering of JIS College of Engineering and all the respected faculty members of Department of CSE for giving the scope of successfully carrying out the project work.

Finally, we take this opportunity to thank Dr. Malay R Dave, Principal of JIS College of Engineering and Dr. Somsubhra Gupta, Dean Academic Affairs, JISCE for giving us the scope of carrying out the project work.

REFERENCES

[1] Zdzisław Pawlak, "Rough set theory and its applications", Institute of Theoretical and Applied Informatics, Polish Academy of Sciences, Bałtycka st 5, 44-000 Gliwice, Poland.

[2] Zdzisław Pawlak, S. K. M Wong, and Wojciech Ziarko, "Rough sets: probabilistic versus deterministic approach", Department of Complex Control Systems, Polish Academy of Sciences, Baltycka 5, 44-000 Gliwice, Poland, and Department of Computer Science, University of Regina, Regina, Saskatchewan, Canada.

[3] Silvia Rissino and Germano Lambert-Torres, "Rough Set Theory – Fundamental Concepts, Principals, Data Extraction, and Applications", Federal University of Rondonia, and Itajuba Federal University, Brazil.

[4] Samir Roy and Udit Chakraborty, "Introduction to Soft Computing", Pearson.

[5] Mehdi Khosrow-Pour, "Encyclopedia of Information Science and Technology, 2nd Edition", Information Resources Management Association, USA.

[6] Jorge Marx Gómez, Michael Sonnenschein, Martin Müller, Heinz Welsch, and Claus Rautenstrauch, "Information Technologies in Environmental Engineering: ITEE 2007 – Third International ICSC Symposium", Springer.

[7] Andrzej Skowron, and Zbigniew Suraj, "Rough Sets and Intelligent Systems - Professor Zdzisław Pawlak in Memoriam, Volume 2", Springer.