On Rough Set Modelling for Data Mining

V S Jeyalakshmi[1],
Centre for Information Technology and Engineering,
M. S. University, Abhisekapatti.
Email: vsjeyalakshmi@yahoo.com

G Ariprasad[2],
Fatima Michael College of Engineering and Technology,
Madurai.
Email: zorro_200812@yahoo.com

Abstract- Many problems in real world can be explained in natural languages. Rough Set Theory is defined with many operations, rules extended from classical set theory and is widely used to model systems related to data mining. The paper is a description of Rough Set Theory and an illustration of modeling a real world problem based on Rough set approximations.

Keywords – *Rough Set Theory* **(***RST), Knowledge Discovery in Data mining (KDD).*

I INTRODUCTION

World is made up of a large amount of information. The information may be uncertain, imprecise and vague. The acquired information from a particular domain is measured by human experts in the form of database or tables. The information may be about events, objects or living beings. The attributes of the acquired tables are features of the particular concept. The data in these tables shall be missing and determining those missing values is a challenging job. The other dimension in information management is to keep only the reduced number of attributes that preserve indiscernible relation among the objects captured in database tables. Nowadays, Rough Set Theory (RST) does these tasks and gets successful results with mathematical operations easily. As pointed out by Z. Pawlak, a Polish Scientist Professor in the early 1980's, [1], [2] "Rough Set philosophy is based on the assumption that every object of the universe of discourse is associated with some information (data, Knowledge)". Rough Set Theory is the approximation of two crisp sets and the roughness of the rough set is used to understand the decisions produced by the system. RST has a promising avenue for sound basis knowledge discovery in data mining (KDD).

There are numerous areas of successful applications of rough set as follows [3]: Analysis of data from peritoneal lavage in acute pancreatitis, supporting of therapeutic decisions, knowledge acquisition in nursing, diagnosis of pneumonia patients, Medical databases (e.g. headache, meningitis) analysis, Image analysis for medical applications, Modeling cardiac patient set residuals, Multistage analysis of therapeutic experience with acute pancreatitis, Breast cancer detection using electro-potentials, Analysis of medical data of patients with suspected acute appendicitis, Attribute reduction in a database for hepatic diseases, Prediction of behavior of credit card holders, Drafting and advertising budget of a company, Customer behavior patterns and Response modeling in database marketing.

Data mining is the process of discovering knowledge from large amount of data [4]. The various steps involved in it are data cleaning, data integration, data selection, data transformation, data mining, pattern evaluation and knowledge presentation. Data cleaning is the removal of noise and inconsistent data. Data integration is the process to combine the multiple data sources. Data selection is to analyze and retrieve the relevant data from the database. Data transformation is the place where data are transformed or consolidated into forms appropriate for mining by performing summary or aggregation operation. Data mining is an essential process where intelligent methods are applied in order to extract data patterns. Pattern evaluation is to identify the truly interesting patterns representing knowledge based on some interestingness measures. Knowledge presentation is where visualization and knowledge representation techniques are used to present the mined knowledge to the user.

In this paper, we use the concept of Rough Set as data mining process to extract data patterns. The paper illustrates various concepts of RST in a simple way with an example data set and thus demonstrates Rule mining through Rough Set modeling for Data mining related problems.

II ROUGHNESS

2.1 Information System and Knowledge Base

The Information System for a real world problem can be defined as a system having information about the domain of that problem called the Universe (U) and every features of that domain called attributes (A). Thus Information System is the system, I = (U, A) where U is the Universe and A is the feature set of that particular domain such that for every $a \in A$, V_a is the domain of the attribute a. An Information System is illustrated with a real world problem involving the identification of a right car suitable for a car race as follows:

Consider a simple Information system having eight objects and four attributes. The objects considered here are eight different cars. These objects are characterized by four attributes such as Name, Mileage, Engine and Car condition. The eight objects are denoted as y_1, y_2, y_3, y_4, y_5, y_6, y_7, y_8 and the set of all these objects forms the Universe of the problem. The four attributes are Name, Mileage, Engine and Car condition. The details are shown in Table 1.

Table 1 - Information System

U	Name	Mileage	Engine	Car condition	Decision for eligibility of car in race
y_1	Porsche 917	Above 200 mph	Flat12	Good	Yes
y_2	Maruty	0-100 mph	796CC F8BMP FI13	Good	No
y_3	2012 Nissan 370Z	100-200 mph	V6	Good	Yes
y_4	Tata Sumo	0-100 mph	Direct Injection Common Rail	Bad	No
y_5	Porsche 917	Above 200 mph	Flat12	Bad	No
y_6	Ferrari Testarossa	0-100 mph	Flat12	Good	Yes
y_7	Porsche 908	100-200 mph	Flat8	Good	Yes
y_8	Maruty	0-100 mph	796CC F8BMP FI13	Bad	No

In this table, the Universe is U = $\{y_1, y_2, y_3, y_4, y_5, y_6, y_7, y_8\}$ and the set of attributes of the given problem is A = {Name, Mileage, Engine, Car condition}.

The domain Name of the attribute set A is given by

$$V_{Name} = \{\{y_1, y_5\}, \{y_2, y_8\}, \{y_3\}, \{y_4\}, \{y_6\}, \{y_7\}\}.$$

The domain Mileage of the attribute set A is given by

$$V_{Mileage} = \{\{y_1, y_5\}, \{y_2, y_4, y_6, y_8\}, \{y_3, y_7\}\}.$$

The domain Engine of the attribute set A is given by

$$V_{Engine} = \{\{y_1, y_5, y_6\}, \{y_2, y_8\}, \{y_3\}, \{y_4\}, \{y_7\}\}.$$

The domain Car condition of the attribute set A is given by

$$V_{Car\ condition} = \{\{y_1, y_2, y_3, y_6, y_7\}, \{y_4, y_5, y_8\}\}.$$

Now (U, A) of the given problem is determined. We have to decide whether the car is eligible for a race or not according to their features. Rough Set Theory always deals with inconsistencies. Here we use RST to find the lower and upper approximations and to make conclusions about the decision making system.

Knowledge Base: Knowledge base for a particular real world problems can be defined as a system having information about the domain of that problem called the Universe (U) and similar characters from that domain collected or classified as relations (R)[3].

Knowledge base is the system, K = (U, R) where U is the universe and R is the family of equivalence relations over U.

From the Table 1, the classification of U according to Name are given as $\{y_1, y_5\}$ for the car name Porsche 917, $\{y_2, y_8\}$ for the car name Maruty, $\{y_3\}$ for the car name 2012 Nissan 370Z, $\{y_4\}$ for the car name Tata sumo, $\{y_6\}$ for the car name Ferrari Testarossa and $\{y_7\}$ for the car name Porsche 908. Thus the relation R_1 for the attribute name in the domain V_{Name} is

$R_1 = \{\{y_1, y_5\}, \{y_2, y_8\}, \{y_3\}, \{y_4\}, \{y_6\}, \{y_7\}\}$.

The classification of U according to Mileage are given as $\{y_1, y_5\}$ for mileage above 200 miles per hour, $\{y_2, y_4, y_6, y_8\}$ for mileage 0 to 100 miles per hour and $\{y_3, y_7\}$ for mileage 100 to 200 miles per hour. Thus the relation R_2 for the attribute experience in the domain $V_{Mileage}$ is

$R_2 = \{\{y_1, y_5\}, \{y_2, y_4, y_6, y_8\}, \{y_3, y_7\}\}$.

The classification of U according to Engine are given as $\{y_1, y_5, y_6\}$ for flat 12 and $\{y_2, y_8\}$ for 796CC F8B MPFI 13, $\{y_3\}$ for V6, $\{y_4\}$ for Direct Injection Common Rail and $\{y_7\}$ for flat 8. Thus the relation R_3 for the attribute Engine in the domain V_{Engine} is

$R_3 = \{\{y_1, y_5, y_6\}, \{y_2, y_8\}, \{y_3\}, \{y_4\}, \{y_7\}\}$.

The classification of U according to Car condition are given as $\{y_1, y_2, y_3, y_6, y_7\}$ for car condition good and $\{y_4, y_5, y_8\}$ for car condition bad. Thus the relation R_4 for the attribute car condition in the domain $V_{Car\ condition}$ is

$R_4 = \{\{y_1, y_2, y_3, y_6, y_7\}, \{y_4, y_5, y_8\}\}$.

Thus the relation set for the Information System in Table 1 is $R = \{R_1, R_2, R_3, R_4\}$ and Knowledge Base $K = (U, \{R_1, R_2, R_3, R_4\})$ where R_1, R_2, R_3, R_4 are elementary categories.

2.2 Indiscernibility

For any subset P of the attribute set A, the indiscernibility relation of P (IND(P))[3] can be defined as

$$IND(P) = \{(y_1, y_2) \in U \times U \mid \forall\ a \in P, a(y_1) = a(y_2)\} \qquad (1)$$

Two objects are said to be indiscernible then they have the same values for all the attribute of P. It can also be called as an equivalence relation and the set of all such equivalence relation can also be denoted as $[y]_P$. Indiscernibility relations can be explained with respect to Table 1 as follows:

$[y]_{Name} = IND(Name) = \{\{y_1, y_5\}, \{y_2, y_8\}, \{y_3\}, \{y_4\}, \{y_6\}, \{y_7\}\}$.

$[y]_{Mileage} = IND(Mileage) = \{\{y_1, y_5\}, \{y_2, y_4, y_6, y_8\}, \{y_3, y_7\}\}$.

$[y]_{Engine} = IND(Engine) = \{\{y_1, y_5, y_6\}, \{y_2, y_8\}, \{y_3\}, \{y_4\}, \{y_7\}\}$.

$[y]_{Car\ condition} = IND(Car\ condition) = \{\{y_1, y_2, y_3, y_6, y_7\}, \{y_4, y_5, y_8\}\}$.

$[y]_{Name,\ Engine} = IND(Name, Engine)$

$= \{\{y_1, y_5\}, \{y_2, y_8\}, \{y_3\}, \{y_4\}, \{y_6\}, \{y_7\}\} \cap \{\{y_1, y_5, y_6\}, \{y_2, y_8\}, \{y_3\}, \{y_4\}, \{y_7\}\}$.

$= \{\{y_1, y_5\}, \{y_2, y_8\}, \{y_3\}, \{y_4\}, \{y_6\}, \{y_7\}\}$.

2.3 Set Approximation

The decision for car in race cannot be defined in a crisp manner using the attributes mentioned in Table 1. In this situation the concept of Rough set emerges. Though it is not possible to classify the cars in the Universe U in a crisp manner, it is possible to understand the cars which will be certainly eligible for a race, the cars which will not certainly eligible for a race and finally, the cars which belong to the boundary between these two certain cases. When such a boundary exists, the set is called Rough [3], [5], [6]. The lower approximation is the cars which are certainly eligible for a race, the upper approximation is the cars possibly eligible for a race and the boundary region is the cars belong to the boundary between cars eligible for a race and doesn't eligible for a race called Rough.

The lower and the upper approximations for $Y \subseteq U$ using the information in $P \subseteq A$ are denoted as $\underline{P}Y$ and $\overline{P}Y$ respectively and defined as follows[7], [8]:

$$\underline{P}Y = \{y \mid [y]_P \subseteq Y\} \qquad (2)$$

and

$$\overline{P}Y = \{y \mid [y]_P \cap Y \neq \varnothing\} \qquad (3)$$

Consider the target set Y shown in Table 1 as a set of cars eligible for a race.

(i.e) Y = {y | decision for eligibility in race (y) = yes}.

Thus $Y = \{y_1, y_3, y_6, y_7\}$ (cars in race) and the attribute subset P = {Name, Engine}. Then the lower approximation of Y is given by

$\underline{P}Y = \{\{y_3\} \cup \{y_6\} \cup \{y_7\}\}$

$= \{y_3, y_6, y_7\}$ and the upper approximation of Y is given by

$\overline{P}Y = \{\{y_1, y_5\} \cup \{y_3\} \cup \{y_6\} \cup \{y_7\}\}$.

$\overline{P}Y = \{y_1, y_3, y_5, y_6, y_7\}$.

2.4 Rough set

The lower and the upper approximation of Y can also be used to define the basic concepts of RST. The boundary region of Y consists of objects that cannot be classified into Y based on the knowledge in the attribute set P. Thus the coefficient of accuracy of approximation is one for crisp set and it is less than one for rough set with respect to the attribute set P. Mathematically, Rough Set can also be characterized by a coefficient of accuracy of approximation, $\alpha_P(Y) < 1$ [3] and

$$\alpha_P(Y) = \frac{|\underline{P}Y|}{|\overline{P}Y|} \qquad (4)$$

For the details provided in Table 1, where P = {Name, Engine}.

$$\alpha_P(Y) = \frac{|\{y_3, y_6, y_7\}|}{|\{y_1, y_3, y_5, y_6, y_7\}|}\ .$$

$$= \frac{3}{5} = 0.6 < 1.$$

The boundary region is the set of objects that can neither be ruled in nor ruled out as a member of the target set. (i.e) $PN_B(Y) = \overline{P}Y - \underline{P}Y$. If the boundary region is non empty then the set is said to be rough with respect to the attribute set P otherwise the set is said to be crisp with respect to the attribute set P.

The negative region is the set of objects can be definitely ruled out as a member of the target set called negative region. (i.e) $U - \overline{P}Y = \{y_2, y_4, y_8\}$. Example: $\overline{P}Y - \underline{P}Y = \{y_1, y_5\}$. The above set approximations are illustrated in the given figure 1.

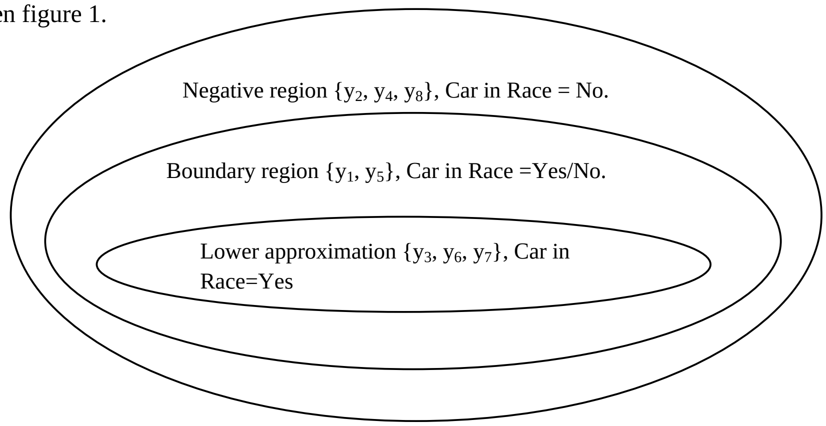

Figure.1 Approximating the set of cars suitable for car race using the four conditional attributes Name, Mileage, Engine and Car condition.

2.5 Reduct

The Reduct P of an Information System, I is a subset of attributes obtained from the attribute set A of I. The attributes rejected from the reduct are redundant and their removal does not worsen the classification problem[3].

Given an information system I = (U, A), the definitions of these notions are as follows. A reduct of I is a minimal set of attributes $B \subseteq A$ such that $IND_I(B) = IND_I(A)$. In other words, a reduct is a minimal set of attributes from A that preserves the partitioning of the universe and hence the ability to perform classifications as the whole attribute set A does[3].

Let I be an information system with n objects[3]. The discernibility matrix of I is a symmetric n x n matrix with entries c_{ij} as given below.

$$c_{ij} = a \varepsilon A \mid a(y_1) \neq a(y_2) \text{, for i, j = 1, 2, ..., n} \qquad (5)$$

Table 2 Discernibility Matrix

	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8
y_1	**{}**	**{n, m, e}**	**{n, m, e}**	**{n, m, e, c}**	**{c}**	**{n, m}**	**{n, m, e}**	**{n, m, e, c}**
y_2	**{n, m, e}**	**{}**	**{n, m, e}**	**{n, e, c}**	**{n, m, e, c}**	**{n, e}**	**{n, m, e}**	**{c}**
y_3	**{n, m, e}**	**{n, m, e}**	**{}**	**{n, m, e, c}**	**{n, m, e, c}**	**{n, m, e}**	**{n, e}**	**{n, m, e, c}**
y_4	**{n, m, e, c}**	**{n, e, c}**	**{n, m, e, c}**	**{}**	**{n, m, e}**	**{n, e, c}**	**{n, m, e, c}**	**{n, e}**
y_5	**{c}**	**{n, m, e, c}**	**{n, m, e, c}**	**{n, m, e}**	**{}**	**{n, m, c}**	**{n, m, e, c}**	**{n, m, e}**
y_6	**{n, m}**	**{n, e}**	**{n, m, e}**	**{n, e, c}**	**{n, m, c}**	**{}**	**{n, m, e}**	**{n, e, c}**
y_7	**{n, m, e}**	**{n, m, e}**	**{n, e}**	**{n, m, e, c}**	**{n, m, e, c}**	**{n, m, e}**	**{}**	**{n, m, e, c}**
y_8	**{n, m, e, c}**	**{c}**	**{n, m, e, c}**	**{n, e}**	**{n, m, e}**	**{n, e, c}**	**{n, m, e, c}**	**{}**

The discernibility matrix is symmetrical with an empty diagonal. From the above matrix, we determine the discernibility function.

An discernibility function f_I of an Information System I is a Boolean function of m Boolean variables $a_1^*, a_2^*, \ldots\ldots, a_m^*$ (corresponding to the attributes $a_1, \ldots\ldots, a_m$) are defined as follows.

$$f_I(\; a_1^*, a_2^*, \ldots\ldots\ldots\ldots\ldots\ldots, a_m^*)$$

$$= \wedge \{\vee c_{ij}^* \mid 1 \le j \le i \le n, c_{ij} \ne \phi\} \text{ where } c_{ij}^* = \{a^* \mid a\varepsilon c_{ij}\} \qquad (6)$$

The discernibility function for the Information System I, where the one letter Boolean variables correspond to the attribute names such as n denotes name of the car, m denotes the mileage of the car, e denotes the engine type and c denotes the car condition is defined as follows.

$$\begin{aligned}
f_I(n, m, e, c) = {} & (n \vee m \vee e)(n \vee m \vee e) \\
& (n \vee m \vee e \vee c)(c)(n \vee m)(n \vee m \vee e) \\
& (n \vee m \vee e \vee c)(n \vee m \vee e)(n \vee e \vee c) \\
& (n \vee m \vee e \vee c)(n \vee e)(n \vee m \vee e)(c) \\
& (n \vee m \vee e \vee c)(n \vee m \vee e \vee c)(n \vee m \vee e) \\
& (n \vee e)(n \vee m \vee e \vee c)(n \vee m \vee e)(n \vee e \vee c) \\
& (n \vee m \vee e \vee c)(n \vee e)(n \vee m \vee c) \\
& (n \vee m \vee e \vee c)(n \vee m \vee e)(n \vee m \vee e) \\
& (n \vee e \vee c)(n \vee m \vee e \vee c).
\end{aligned}$$

where each parenthesized tuple is a conjunction in the Boolean expression and they are called as implicants. After simplification the function is determined as follows:

$$f_I(n, m, e, c) = (n \vee m \vee e)(n \vee m \vee e \vee c)(c)$$
$$(n \vee m)(n \vee e)(n \vee e \vee c)(n \vee m \vee c).$$

Here each tuple is a prime implicants. Then we can use the Karnaugh's map [9] to determine the essential prime implicants.

n, m \ e, c	00	01	11	10
00		1		
01				
11	1	1	1	1
10			1	1

Y = nmec + n me'c.

Y = nmc (e+e').

Y = nmc (Since e+e' = 0).

From the map, we determined the essential prime implicants. The essential prime implicants are the combination of Name, Mileage and Car condition.

{Name, Mileage} and {Name, Car condition} are the different unique minimal set of attributes called the reducts and their relative indiscernibility function helps to determine the decision. These two combination play a vital role to frame the rule. Here the decision is to decide if a car can suitable for a car race or not.

Consider the target set $Y = \{y_1, y_3, y_6, y_7\}$ (car suitable for a race) and the reduct (attribute subset) P = {Name, Mileage}.

IND(Name, Mileage)

$= \{\{y_1, y_5\}, \{y_2, y_8\}, \{y_3\}, \{y_4\}, \{y_6\}, \{y_7\}\} \cap \{\{y_1, y_5\}, \{y_2, y_4, y_6, y_8\}, \{y_3, y_7\}\}.$

$= \{\{y_1, y_5\}, \{y_2, y_8\}, \{y_3\}, \{y_4\}, \{y_6\}, \{y_7\}\}.$

Lower approximation of Y is given by

$\underline{P}Y = \{\{y_3\} \cup \{y_6\} \cup \{y_7\}\}$

$\underline{P}Y = \{y_3, y_6, y_7\}$ and the upper approximation of Y is given by

$\overline{P}Y = \{\{y_1, y_5\} \cup \{y_3\} \cup \{y_6\} \cup \{y_7\}\}.$

$\overline{P}Y = \{y_1, y_3, y_5, y_6, y_7\}.$

Negative region $= U - \overline{P}Y = \{y_2, y_4, y_8\}.$

Boundary region $= \overline{P}Y - \underline{P}Y = \{y_1, y_5\}.$

Consider the target set $Y = \{y_1, y_3, y_6, y_7\}$ (car suitable for a race) and the attribute subset P = {Name, Car condition}.

IND(Name, Car condition)

$= \{\{y_1, y_5\}, \{y_2, y_8\}, \{y_3\}, \{y_4\}, \{y_6\}, \{y_7\}\} \cap \{\{y_1, y_2, y_3, y_6, y_7\}, \{y_4, y_5, y_8\}\}.$

$= \{\{y_1\}, \{y_2\}, \{y_3\}, \{y_4\}, \{y_5\}, \{y_6\}, \{y_7\}, \{y_8\}\}.$

Lower approximation of Y is given by

$\underline{P}Y = \{y_1\} \cup \{y_3\} \cup \{y_6\} \cup \{y_7\} = \{y_1, y_3, y_6, y_7\}.$

Upper approximation of Y is given by

$\overline{P}Y = \{y_1\} \cup \{y_3\} \cup \{y_6\} \cup \{y_7\} = \{y_1, y_3, y_6, y_7\}.$

$\overline{P}Y = \{y_1, y_3, y_6, y_7\}.$

Negative region $= U - \overline{P}Y = \{y_2, y_4, y_5, y_8\}.$

From the indiscernibility relations and set approximations determined, we can obtain the lower and upper approximations, negative region and boundary region for the system. For the target set, the set of all cars suitable for a race $Y = \{y_1, y_3, y_6, y_7\}$, the negative region indicates that the car is not suitable for a race, the lower approximations corresponds to the cars which are certainly suitable for a race and the upper approximation region corresponds to the cars which are possibly eligible for a race or not. The boundary region is the rough set region and shows the cars which belong to the boundary region to eligible for a car race or not for the entire attribute set P considered. The lower approximation and the upper approximation of the reduct gives the crisp set and the rough set. Therefore, we use both the rough and crisp set for modeling the system to decide if a car is suitable for a race or not using rule mining.

F. Rule mining

The rules derived from the lower approximation are certainly valid and hence such rules are known as certain rules. The rules derived from the upper approximation are possibly valid and hence such rules are known as possible rules.

Rough set rules for the system has the general format,

IF antecedent THEN consequent.

The decision for the whole system depends on both the attribute set {Name, Mileage} and {Mileage, Car condition}. The certain rules and possible rules are given as follows:

IF the car name is Ferrari Testarossa and Mileage is 0-100 mph THEN the car eligible for a race.

(i.e) (Name, Ferrari Testarossa) &
(Mileage, 0-100 mph) → (race, yes)

IF the car name is Porsche 908 and Mileage is 100-200 mph THEN the car eligible for a race.

(i.e) (Name, Porsche 908) &
(Mileage, 100-200) → (race, yes).

IF the car name is Maruty and the car condition is Good THEN the car doesn't eligible for a race.

(i.e) (Name, Maruty) &
(Car condition, Good) → (race, no).

IF the car name is Tata sumo and the car condition is Good THEN the car doesn't eligible for a race.

(i.e) (Name, Tata sumo) &
(Car condition, Good)→ (race, no).

IF the car name is Porsche 917 and the car condition is Good THEN the car eligible for a race.

(i.e) (Name, Porsche 917) &
(Car condition, Good) → (race, yes).

IF the car name is Porsche 917 and the car condition is Bad THEN the car doesn't eligible for a race.

(i.e) (Name, Porsche 917) &
(Car condition, Bad) → (race, no).

IF the car name is 2012 Nissan 370Z and the car condition is Good THEN the car eligible for a race.

(i.e) (Name, 2012 Nissan 370Z) &
(Car condition, Good) → (race, yes).

From the Table 1, large number of rules can be framed. But we frame only few rules to decide whether the car is eligible for a car race. These rules are enough to take the decision for the whole system. So Rough Set analysis helps in determining a minimal number of essential rules for making decisions on real-world problems.

III Conclusion:

In this paper, RST concepts were used to model a system to desire upon the car's eligibility for a race based on the Information System acquired for the same. The attributes of the Information system were investigated, reducts identified and indiscernibility, set approximations of the reducts are determined. From the set approximations, definite and possible rules are mined for the system to desire on a car's eligibility for a race. Thus this paper is an illustration on the application of RST concepts for data mining applications.

REFERENCES:

[1] Yingjie Yang, Robert John, 'Roughness bounds in Rough Set Operations' Centre for Computational Intelligence, School of Computing, De Montfort University, The Gateway, Leicester, LE1 9BH, United Kingdom.
[2] Z. Pawlak, 'Rough sets and intelligent data analysis', Information Science 147:1–12, 2002.
[3] Jan Komorowski, 'Rough sets, a tutorial book' Department of Computer and Information Science Norwegian University of Science and Technology (NTNU) 7034 Trondheim, Norway, Institute of Mathematics, Warsaw University Banacha 2, 02-097 Warszawa, Poland, Institute of Mathematics, Warsaw University of Technology Pl. Politechniki 1, 00-665 Warszawa, Poland and Polish-Japanese Institute of Information Technology Koszykowa 86, 02-008 Warszawa, Poland.
[4] Jiawei Han and Micheline Kamber, 'Data Mining: Concepts and Techniques', Second edition, University of lllinois at Urbana-Champaign.
[5] E. Venkateswara Reddy, G.V. Suresh and E.S. Reddy, 'Rough set analysis for uncertain data classification' CSE Department Universal College of Engineering &Technology, Guntur, India.
[6] Zdzislaw Pawlak, 'Rough sets and data mining' Institute of Theoretical and Applied Informatics, Polish Academy of Sciences, ul. Baltycka 5, 44 100 Gliwice, Poland and Rough set – wikipedia free encyclopedia.
[7] Z. Pawlak, 'Rough Sets - Theoretical Aspect of Reasoning about Data', Kluwer Academic Pubilishers (1991).
[8] Z. Pawlak, 'International Journal of Computer and Information Sciences', 11, 1982, p.341.
[9] Karnaugh's map and Boolean algebra – Approximate Boolean Reasoning Approach to Rough Sets and Data Mining Hung Son Nguyen Institute of Mathematics, Warsaw University son@mimuw.edu.pl RSFDGrC, September 3, 2005.