Rough Sets – Application in Data Mining

Dr. Robinson Chellathurai[1], L. Jesmalar[2]

[1]Associate Professor, Scott Christian College (Autonomous), Nagercoil.
[2]Lecturer, Holy Cross College, Nagercoil.
[1]l.jesmalar@yahoo.com

***Abstract*–The chief objective of this study is to show the usefulness of Rough set theory in Data mining, particularly on the segmentation. For this purpose the basic details are presented in the first two sections and the application is illustrated in section 3.**

***Keywords*–Decision systems, indiscernibility, consistent, reduct, cuts,Tolerance relation.**

I. Introduction

Rough set theory (RS) is a mathematical formalism developed by Zdzislaw Pawalk (1982) to analyze data tables. In the RS terminology, a data table is called as an "Information system". If some of the attributes are interpreted as outcomes of classification, it is also called as a "decision system'. An example of such a system is in Table I.

The identity of 9 Xth STD students

TABLE I
A TABLE REPRESENTING A CLASSIFICATION OF STUDENTS

No	Age	Parents education	Class room behavior	Place or Origin	PERFOR MANCE
x_1	16	educated	Good	Urban	Excellent
x_2	16	educated	Good	Urban	Excellent
x_3	14	educated	Good	Urban	Excellent
x_4	10	educated	Good	Urban	Fair
x_5	13	uneducated	Good	Urban	Fair
x_6	15	educated	Above average	Rural	Fair
x_7	13	uneducated	Above average	Rural	Normal
x_8	14	uneducated	Above average	Rural	Normal
x_9	16	uneducated	Average	Rural	Normal

Here PERFORMANCE is the decision attribute. Symbols used: Class room behavior – CRB, Parents education – PE, Place of origin – PO, Performance – PE.

In RS data model information is stored as in Table1. Here each row (tuple) represents a fact or an object. The main object of RS data analysis is to reduce data size.

II. Rough Set Theory

A. Data model

i) In the above column one is the set of objects denoted by U
ii) Columns 2 to 5 are called the attributes whose values for each objects are denoted by V_a
iii) Denote by A, the set of attributes a: $U \rightarrow V_a$

We can split the set of attributes in to two subsets $C \subset A$ and D =A-C, respectively the conditional set of attributes (Age, PE, PO) and the decision (or Performance)attribute. Conditional attribute gives the measured features of the ten objects, while the decision attribute is a posteriori outcome of classification. Here it is PERFORMANCE(PF).

B. Reduction of Dimensionality

A table may be redundant in two ways. The first form of redundancy is easy to observe. Some objects may have same features in all the attributes. This is true in the case of tuples $x_1$and x_2 in Table I. Here for reducing data it is enough if we store only one of the two. This has to be done for all the pairs. Such pairs are termed as *indiscernible tuples*. The Second form of redundancy is more difficult to locate in large data tables. Now in Table I, the values of the conditional attributes allow us to classify every object. If a student 16 years old, his parents are educated, his class room behavior is good and coming from urban area then his class performance is Excellent. It is definite that his performance is Excellent. If we do not consider AGE attribute, we cannot any more able to classify objects of the type ⟨*educated, good, URBAN*⟩.Both x_3 and x_4 have the same features except age but the PERFORMANCE(x_3) = Excellent and that of x_4 is Fair. While age is needed to discriminate from performance, we may be able to remove some other attributes (i.e., columns) without losing the classification power. For example if we erase the ORIGIN, still we can classify all the objects. More Formally,

$$(AGE (x) =AGE(x')) \wedge (PE(x) = PE(x')) \wedge(CRB(x) =CRB(x'))\Rightarrow(PF(x)=PF(x'))$$

Now let us consider the method of managing these two form of redundancy.

TABLE II
VALUES FOR TABLE I

U = { $x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9$,}
A = {AGE, PE, CRB, PO, PF}
V_{AGE} = {10,13, 14,15,16}
V_{PAREDU} = {EDUCATED, UNEDUCATED}
$V_{C.R.BEHAVIOUR}$ = {Average, Above average, Good}
V_{PLACE} = {Rural, Urban}
$V_{PERFORMANCE}$= {Normal, Fair, Excellent}

TABLE III
A MODIFIED TABLE OF TABLE I

No	Age	Parents education	Class room behavior	Place or Origin	PERFORMANCE
x_1	16	educated	Good	Urban	Fair
x_2	16	educated	Good	Urban	Excellent
x_3	14	educated	Good	Urban	Excellent
x_4	10	educated	Good	Urban	Fair
x_5	13	uneducated	Good	Urban	Fair
x_6	15	educated	Above average	Rural	Fair
x_7	13	uneducated	Above average	Rural	Normal
x_8	14	uneducated	Above average	Rural	Normal
x_9	16	uneducated	Average	Rural	Normal

Here the tuples x_1 and x_2 are the same values for the conditional attributes (they are C-indiscernible) but the performances are different. This table is not consistent.

1) Indiscernibility relation and Rough sets

Two tuples are said to be indiscernible with respect to a set of attributes $B \subseteq A$ if

$$a(x) = a(x') \quad \forall \ a \in B$$

Many have proved that this relation is an equivalence relation [1].We denote the B-indiscernibility (Relative to attributes in B)as $[x]_B$ .In Table1 $[x_1]_C = \{x_1, x_2\}$ where C={ Age, PE, CRB, PO}, C is called the conditional set of attributes. We call the partition induced by a B- indiscernibility relation on a set U of objects as $IND_B(U)$.

As suggested earlier, it suffices to keep only one tuple from every indiscernibility class. We denote by CX, the set of tuple X using C-indiscernibility. In the case of Table 1 we can represent the set $X^*_{\text{Excellent}}$ as

$$CX_{\text{Excellent}} = \{[x_1],[x_2]\} \quad \text{------------- (1)}$$

This is not always possible. A table is said to be *consistent* if every C- indiscernibility class has a unique value for the decision attribute. This implies that two tuples which have the same features are equally classified. The above representation is useful only when a table is consistent. For example in Table1 x_1,x_2 are identical in all the attributes but x_3 differ from both only in AGE, however it has the same PERFORMANCE. This might be due to a wrong entry even.

To represent a set which is not precisely definable by indiscernibility? Classes, we can find a lower and an upper approximation of it, in the following way

$$\underline{C}\ X_{GOOD} = \{ x \in U / [x]_C \subseteq X_{GOOD} \}$$

$$\overline{C}\ X_{GOOD} = \{ x \in U / [x]_C \cap X_{GOOD} \neq \Phi \}$$

$\underline{C}$ X is the set of indiscernibility classes which are subsets of X. If X is characterized by a particular decision value, the above implies that all indiscernibility class in $\underline{C}$ X contain objects with that value. Due to this, the data tells us that we are able to classify objects in $\underline{C}$ X. There may also be indiscernibility classes which contain only some tuples in X. In such cases, we cannot classify them. These are objects in $\overline{C}X - \underline{C}X$, also called boundary region. Finally, there may be elements in U - $\overline{C}$X, which contain tuples not in X. (*All tuples with decision value *Excellent* for the decision attribute.)

These also can be (negatively) classified. If the boundary region is empty ie $\underline{C}X = \overline{C}$ X, X is said to be *crisp* or *precise*. Otherwise, it is called a *rough set*, which in practice is a pair of set approximations. Rough set theory can be used to represent which ever set X, but we are usually interested in approximating sets of tuples with respect to the decision attribute.

2) Reducts

We have already seen that some columns of a table can be removed without affecting the classification power of the system. This can be extended to tables where we do not distinguish between conditional and decision attributes. In this case, an attribute can be removed if there are no two tuples which become indiscernible.

A *reduct* is a minimal set of attributes that preserve the indiscernibility relation.
In terms of properties of the table $\langle U, A, V_a \rangle$,a reduct is a set of attributes R such that

i. $R \subseteq A$
ii. $IND_R(U) = IND_A(U)$
iii. $IND_{R-a}(U) \neq IND_A(U) \quad \forall\ a \in R$

While computing equivalence classes are easy, finding minimal reducts S NP –hard. One way of solving this, which in practice is tackled using strong heuristics, is based on Boolean reasoning .given a table, for every pair of tuples we list the attributes which are different. Table IV is a more realistic example of classification of students, which has already been reduced by using indiscernibility classes.

Here tuples have been aggregated in C –indiscernibility classes.CL_2 is not consistent in the decision attribute and hence it has been subdivided into two classes with different values for the decision attribute. The NUM column stands for the number of tuples in the corresponding indiscernibility class.

TABLE IV

No	AG	P E	CB	PO	PF	NUM
CL_1	14	educated	Good	Urban	Excellent	118
CL_{2a}	16	educated	Good	Urban	Excellent	116
CL_{2b}	16	educated	Good	Urban	Fair	123
CL_3	10	educated	Good	Urban	Fair	112
CL_4	13	uneducated	Good	Urban	Fair	110
CL_5	15	educated	Above average	Rural	Fair	111
CL_6	13	uneducated	Above average	Urban	Normal	18
CL_7	14	uneducated	Average	Urban	Normal	11
CL_8	16	uneducated	Average	Rural	Normal	11
					Total	730

TABLE V
INDISCERNIBILITY MATRIX FOR TABLE IV

	CL_2	CL_3	CL_4	CL_5	CL_6	CL_7	CL_8
CL_1	AG	AG	AG,PE	AG,CB,PO	AG,CB,PE	PE, CB, PO	AG, PE, CB, PO
CL_2		AG	AG,PE	AG, CB,PO	AG, PE CB	PE, CB, PO	AG,CB, PO
CL_3			AG, PE	AG,CB, PO	AG,PE, CB,	AG, PE, CB, PO	AG, CB, PO
CL_4				AG, PE, CB, PO	AG, PE, CB, PO	AG,CB, PO	AG,CB, PO
CL_5					AG, PE,PO	AG, PE,CB	AG,PE, CB
CL_6						PO,AG,CB	AG, CB,PO
CL_7							AG,

This matrix is symmetric .Entry $(CL_x, CL_{x'})$ = $a_1, a_2, \ldots a_k$ corresponds to the expression $a_1 \cup a_2 \ldots\ldots \cup a_k$

For every pair of tuples, we can list all attributes which allow us to discern between them. Table 5 represents the indiscernibility matrix for Table4 .Every entry $(CL_x, CL_x{}')$ = $a_1, a_2, \ldots a_k$ of this matrix corresponds to a Boolean expression $E(CL_x, CL_{x'})$ = $a_1 \cup a_2 \ldots\ldots \cup a_k$. If we want to know if two tuples are discernible using only some of the attributes, we can do the following. We assign **true** to the attributes we are considering and false to the remaining ones,then we can evaluate the corresponding expression. $E(CL_x, CL_{x'})$ tells us if we are able to discern CL_x, from $CL_{x'}$

If we want to know when all objects are discernible from each other, we can take conjunction of all the entries of the indiscernibility matrix. We have the following discernibility function for our example

f(AG, PE, CB, PO) = (AG)(AG) (AG∪PE) (AG∪CB∪PO) (AG∪PE∪CB)(AG∪CB∪PO) (AG∪PE∪ CB∪PO) (AG) (AG∪PE) (AG∪CB∪PO) (AG∪PE∪CB) (AG∪PE∪CB∪PO) (AG∪CB∪PO)

Notice that all the tuples are still discernible From each other

(AG∪PE) (AG∪CB∪PO) (AG∪PE∪CB) (AG∪PE∪CB∪PO) (AG∪PE∪CB∪PO) (AG∪PE∪CB∪PO) (CB) (AG∪CB∪PO) (AG∪CB∪PO) (AG∪PE∪PO) (AG∪PE∪CB) (AG∪PE∪CB) (AG∪CB∪PO) (AG∪CB∪PO) ----------------- (2)

TABLE VI
TABLE IV AFTER REDUCTION

NO	AG	CB	AP	NUM
CL_1	14	GOOD	Excellent	118
CL_{2a}	16	GOOD	Excellent	116
CL_{2b}	16	GOOD	Fair	123
CL_3	10	GOOD	Fair	112
CL_4	13	GOOD	Fair	110
CL_5	15	Above average	Fair	111
CL_6	13	Above average	Normal	18
CL_7	14	Average	Normal	11
CL_8	16	Average	Normal	11

Here it seems to be a little complex, but it is really very easy .In reality ,it is another way of writing the indiscernibility matrix given a truth assignment corresponding to the variables we want to keep, every group of disjunctions is true iff we are able to distinguish between two particular indiscernibility classes. For example, suppose we want to use only parents education as the attribute. For the $E(CL_1, CL_2)$ =AGE is false, hence we cannot discern class CL_1 from CL_2.In factPE

(CL_1)=PE(CL_2)=Educated on the other hand E$(CL_1=CL_4)$ = (AG ∪ PE) is true ,hence we can discern class CL_1 from CL_4.In fact PE (CL_1)=EducatedPE(CL_2)=uneducated.

Equation 2 is true iff all group of disjunctions are true. This implies that we are able to discern any two classes from each other using only the attributes whose corresponding variables are true.

The problem of finding a subset of the attributes which preserve the indiscernibility relation can be reduced to the problem of finding the implicants of equation(2).

That is a conjunction of variables such that if these variables are true also the property of the function is true. In particular the object of using the concept of RS is the estimation of prime implicants which implicants with the minimal size.

In the case of (2) the prime implicants is f(AG,PE,CB,PO) =AG∩ CB. This implies that the attributes AGE and CLASS BEHAVIOUR are sufficient to preserve the indiscernibility relation. Table 5 represents the smallest possible redundancy reduction using only information from data.

C. Numerical measures of rough Approximations

The main idea of rough approximations a set can be represented by a lower and upper approximation. Suppose an indiscernibility class has 100 elements. If it is possible to classify only 75 in X. Now this set is well within the boundary of X. Even if we are able to classify only one, then again it is true. This shows that two approximations are not sufficient to represent a set of objects in a satisfactory way.

The word Accuracy is the measure of the roughness of a set. If a set has $\underline{C}$X =Φ and $\overline{C}$X=U, the approximation does nothing since for any x∈ U we cannot decide if x ∈ X. On the other hand if $\underline{C}$X=$\overline{C}$X=X, the set is precise(crisp) and for all the elements we know whether x∈U if x∈ X or not .This property is denoted by the formula:

$$\alpha_c(X) = \frac{|\underline{C}X|}{|\overline{C}X|}$$

By definition $0 \le \alpha_c(X) \le 1$ and if $\alpha_c(X) = 1$,X is said to be precise(crisp) with respect to C.

Rough membership tells us how much a discernibility class belongs to a setX. This property is defined as $\mu_x^C(x) = \frac{|[x]_c \cap x|}{|[x]_c|}$When $[x]_c$ is a proper subset of X, then $\mu_x^C(x)$=1 otherwise $0 \le \mu_x^C(x) < 1$

The above to some extent resubles property of the membership function in Fuzzy sets.

III APPLICATIONS

A. Data Reduction

Indiscernibility relation and reducts can be used to data size. Table 6 shows how more than 700 tuples can be represented in a compact way using only two conditional attributes.

B. Missing value handling

Handling missing values exist even before RS. Nine methods are compared in[GBH01].A draw back in these is that the statistical distribution of the attribute values is not usually known a priori

Otherwise, traditional RS methods may be extended to manage **null** values. All the methods are based on the tolerance relations.

1) Tolerance relations

A tolerance relation on U is a subset of UxU which is reflexive and symmetric , not necessarily transitive. The symmetry condition is also relaxed in [SVOO].In [wan] a tolerance relation is presented, which also classifies objects with some missing values. In[val03]the authors does something similar but also introduced a third approximation for objects which are not similar but have **null** values. Both these approaches have not answered completely to this problem.

C. Feature selection and Extraction

Some of the feature selection methods are presented in[KPS99]and[DG00]one is to evaluate all reducts and to take the so- called **core** which is the intersection of all of them. This is simple to evaluate another way is to divide the data table into sub tables and keep only the reducts which appear **sufficiently often** (This can be done numerically).These are called **approximate reducts** ,and may be useful to reduce noise effects.

These methods distinguish between **useful** and **useless** attributes. In reality, every attribute more or less useful, and we can define a continuous measure to quantify this. This can be done in two steps: In the first step, we evaluate as how much decision attributes depend on condition ones. Then we erase an attribute and we evaluate again the dependency. The more it is important for classification, the more this value will decrease. From the difference of dependency, we obtain a value for the utility of the attribute .This is done in two steps
Step (1).

Dependency between attributes

This dependency is measured as

$$\gamma(B,D) = \frac{\sum_{X \in IND_{D}(U)} |\underline{B}X|}{|U|}$$

If all sets X are crisp, γ(B,D)=1.In fact, this implies that using the attributes in B we can precisely define the partition IND_D(U).We cannot precisely decide about the membership of an object, that is $\underline{B}$X =Φ, γ(B,D)=0.

Now we can calculate as how much removing an attribute changes the original dependency.

$$\sigma_{B,D}\ (a) = \frac{\gamma(B,D)-\gamma(B-\{a\},D)}{\gamma(B,D)}$$

$If\ B-\{a\}$ is a reduct, γ(B,D)= γ(B-{a},D) and $\sigma_{B,D}$ (a)=0.In fact ,the attribute 'a'is not significant at all. If B is a reduct, it may happen that some of its attributes are less useful than others, since they allow to distinguish between a few small classes.

Discretization a main topic in RS methods .It is always possible to find Discretizations which preserve decision class **via** Boolean reasoning.

This can be done as follows.

Step (i)

For every attribute, all possible **cuts** are listed.

A cut is the median point between two adjacent values. For example ,if an attribute take values {1,2,4,5,7,9}, cuts are {1.5,3,4.5,6,8}.

Step (ii)

For every pair of objects with different decision attributes, the set of cuts which distinguishes them is to be identified

Step (iii)

These cuts are used to evaluate a discernibility formula ,as we did it to find the reducts. We now illustrate this by an example.

Consider the table VII

1. First, we enumerate all attribute values and all cuts.

V_{A_1} ={1,2,5,7,8,9} , V_{A_2} ={1,2,4,5}

CutsA_1={1.5,3.5,6,7.5,8.5},

CutsA_2={1.5,3,4.5}

TABLE VII
ATTRIBUTES OF A TABLE WITH CONTINUOUS VALUES

≠	A_1	A_2	D
a_1	1	2	YES
a_2	5	4	YES
a_3	2	5	NO
a_4	7	1	YES
a_5	8	1	NO
a_6	9	5	NO

TABLE VIII
TABLE VII AFTER DISCRETIZATION

≠	A_1	A_2	D
a_1	ca1-1	ca2-1	YES
a_2	A1-1	A2-1	YES
a_3	A1-1	A2-2	NO
a_4	A1-1	A2-1	YES
a_5	A1-2	A2-1	NO
a_6	A1-1	A2-2	NO

For every attribute, we have selected one cut.

This divides the continuous domain of the attribute in to two discrete spaces.

2. Now, we list the cuts which distinguishes tuples from different classes

$$E(a_1,a_3) = \underbrace{1.5}_{A_1} \quad \underbrace{3,4.5}_{A_2}$$

$$E(a_1,a_5) = \underbrace{1.5,3,3.5,6,7.5}_{A_1} \quad \underbrace{1.5}_{A_2}$$

$$E(a_1,a_6) = \underbrace{1.5,3.5,6,7.5,8.5}_{A_1} \quad \underbrace{3,4.5}_{A_2}$$

$$E(a_1,a_3) = \underbrace{1.5}_{A_1} \quad \underbrace{3,4.5}_{A_2}$$

$$E(a_1,a_5) = \underbrace{1.5,3,3.5,6,7.5}_{A_1} \quad \underbrace{1.5}_{A_2}$$

$$E(a_2,a_3) = \underbrace{3.5}_{A_1} \quad \underbrace{4.5}_{A_2}$$

$$E(a_2,a_5) = \underbrace{6,7.5}_{A_1} \quad \underbrace{3}_{A_2}$$

$$E(a_2,a_6) = \underbrace{6,7.5,8.5}_{A_1} \quad \underbrace{4.5}_{A_2}$$

$$E(a_3,a_4) = \underbrace{3.5,6,7.5}_{A_1} \quad \underbrace{4.5,3,1.5}_{A_2}$$

$$E(a_4,a_5) = \underbrace{7.5}_{A_1} \quad \underbrace{NIL}_{A_2}$$

For example $E(a_4,a_5) = \underbrace{7.5}_{A_1}$

In fact a_4 : (7,1) and a_5 : (8,1) can be separated by 7.5 for the first attribute. This is clearly exhibited in Fig. 1.

Black circles represent the objects in class NO, small black dots corresponds to YES ,only two values per attribute are needed to distinguish the two classes. Discretization obtained by Boolean reasoning is illustrated by dotted lines.

D. Data mining

Data mining indicates a family of different tasks. Among them so far the rough set theory is applied to classification, clustering and Association rules. We shall indicate the utility in the segmentation.

1) Segmentation

Segmentation is the process of organizing the time series in to few intervals having uniform characteristics. It is for dimensionality reduction. The segment points divide the time axis into intervals behaving approximately according to a simple model. Since the model's behaviors are almost parallel the parameter coefficients alone are to be stored. Here in similarity the parallel coefficients of the models may not vary between and hence each parameter can be put in a very small interval and the indiscernibility criteria can be applied to reduce the dimension as discussed in this . Thus the Rough set can be a convenient method for segmentation.

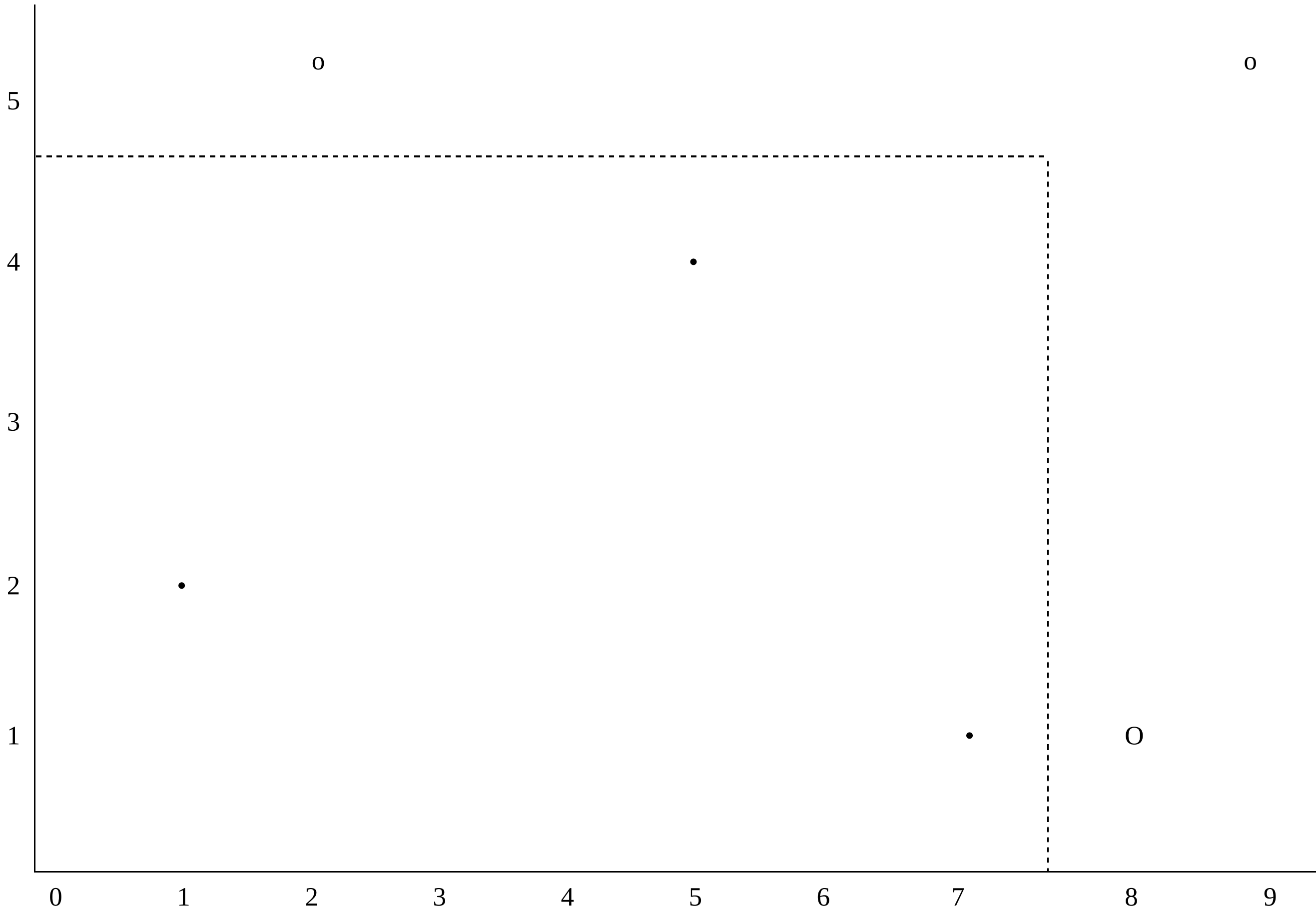

Fig 1: Graphical representation of Table VII

CONCLUSION

The study reveals that the rough set theory can be used in the data mining more efficiently in time series segmentation.

REFERENCES

[1] Dunctsh and Gedia. Roughian – Rough information analysis.URL ; Citeseer.nj.nec.com/ 488866.html

[2] Dunctsh and Gedia.Rough set data analysis in the kdd process URL ; Citeseer.nj.nec.com/ 38773.html

[3] Pawlak, Zdzisław "Rough sets" Int .J.computer and information sci ;11:341-356,1982

[4] R.Slocoinski and D.Vander pooten.A generalized definition of rough approximations based on similarity. IEEE Transactions on Data and knowledge Engineering.

[5] Daniel Lemine; A better Alternative to piecewise linear time series segmentation; Technical report cs. DB/0605103arxiv.org,2006.

[6] Zhaohong Deng, Fu –Lai chung ,shitongWang. Clustering –Inverse: A Generalized model for pattern –Based Time series Segmentation, Journal of intelligent learning systems and Applications,2011,3,26-36.

[7] Duntsh and G. Gedia.Rough set data analysis URL ; Cite-seer.nj.nec.com/ dntsch00rough.html

[8] I.Dunctsh andG. Gedia.andH.S Nguyen.Rough set data analysis in the kdd process URL ; Citeseer.nj.nec.com/ 387773.html

[9] Usama M.Fayyad,Gregory piatetsky-shapiro,Padhraic smyth and Ramasamy uthurusamy,editors. Advances in knowledge discovery and data mining, Chap I pages 1-36,MIT press,Cambridge, MA,1996.

[10] J.Komorowski, L.Polkowski andA.mskowron,editors.Rough- Fuzzy Hybridization pages 3-98.Springer verlag,1999.Cite-seer.nj.nnec.com/ Komorouski 98.rough.html.

[11] J.W.Grzymala –Busse and M.Hu.A comparison of several approaches to missing values in data mining.In Ziarko and Yao [ZY01]

[12] R.Solwinskj and D.Vanderpooten.A generalized definition of rough approximations based on similarity.IEEE Transactions on Data and Knowledge Engineering, 12(20:331-336,2000.

[13] E.Valentini.Ling uaggiodi iterrogazione perbasi di dati multimediali basatosu Fuzzy set e rough set.master's thesis,University of Bologna,2003

[14] Guoyin Wang.Extension of rough set under incomplete information systems.

