ANOMALY DETECTION
-A Soft Computing Approach

T. Y. Lin
Mathematics and Computer Science
San Jose State University
San Jose, California 95192
tylin@sjsumcs.SJSU.EDU

Abstract

Computer are finite discrete machines, the set of real numbers is an infinite continuum. So real numbers in computers are approximation. Rough set theory is the underlying mathematics. A "computer" version of Weistrass theorem states that every sequence, within the radius of error, repeats certain terms infinitely many times. In terms of applications, the theorem guarantees that the audit trail has ***repeating patterns.*** *Examining further, based on fuzzy-rough set theory, hidden* ***fuzzy relationships (rules)*** *in audit data are uncovered. The information about the repeating data and fuzzy relationships reflect "unconscious patterns" of users' habits. They are some deeper "signatures" of computer users, which provide a foundation to detect abuses and misuses of computer systems. A "sliding window information system" is used to illustrate the detection of a "simple" virus attack. The complexity problem is believed to be controllable via rough set representation of data.*

1. Introduction

What are patterns? Do they exist? One could approach "hard" patterns from the point of view of algorithmic information theory. Unfortunately, algorithm information theory asserts that almost all finite sequences have no patterns [1], [2] However, "soft patterns " do exist. In this paper we develop two types of patterns, one is repeating records (within the radius of error), the other is fuzzy relationships (or rules) among data. In the area of intrusion detection, we believe users exhibit "unconscious patterns" [2], [3]. In this paper, we continue our earlier efforts on the fundamental questions: Do soft patterns exist in audit trails? What types of patterns are there? [5], [6] Some experimental results based on DataLogic software will be reported in the future paper. Datalogic is a software system developed by Reduct Inc based on Rough Set Theory.

Let us say few words about the "new" computing and mathematical concepts that will be used in this paper. Recently Zadeh organized a soft computing program at Berkeley, and spoke about soft computing at SIMTEC'93 [7], [8]. About the same time, Pawlak proposed all-embracing soft sets at RSKD'93 [9]. The notion of soft sets is a unified view of classical, rough, and fuzzy sets. Rough sets and fuzzy sets are complementary generalizations of classical sets. Fuzzy set theory allow partial memberships to handle vagueness, while rough set theory allows multiple memberships to deal with indiscernibility. According to Zadeh, *soft computing* includes, at least, fuzzy logic, neural network, probabilistic reasoning, belief network, genetic algorithms, and parts of learning and chaos theories. We believe that the notion of soft sets developed by Pawlak school are also part of soft computing.

Computers are finite discrete machines, however, the set of real numbers is an infinite continuum. So the representation of real numbers in computers must be approximations. What is the mathematical theories behind such approximations. Pawlak' rough set theory turns out to be the right mathematical model for such representations [10], [11]. In this paper, first we examine the properties of numbers represented in computers from the point of view of mathematical analysis. Earlier, we have obtained a "computer" version of Weistrass theorem [5], which states that every sequence in a closed interval repeats, within the radius of error, certain values infinitely many times. In terms of our applications, the theorem implies that in the "infinite" input stream of records of numbers or strings, there are repeating patterns. Note that if the input data are fixed length strings, the data certainly will repeat (since there are only finite number of them). The audit trail can be interpreted as an infinite input stream of records of a database. So the theorem guarantees that

(a) there are repeating records.

These repeating data are not necessarily the only patterns. Some logical relationships among these input data may repeating themselves "infinitely" many times. So based on rough set theory again, we examine further the hidden **repeating fuzzy relationships** among these data. These relationships are often the reflection of some **unconscious patterns** of user's habits [3]. The fuzzy-rough set methodology allow us to find more elaborate hidden phenomena in the audit trail, namely,

(b) The repeating fuzzy relationships(rules).

The information about the "repeating records/rules" (unconscious habits) are often the deeper facts about users. Thus provide us a foundation to detect the abuse and misuse of computer systems.

2. Rough Sets and Numbers in Computers

As remarked earlier, the set of real numbers is an infinite continuum., while computer memory is finite. How real numbers are represented in computers? Let X be an interval X=[a, b] which covers the range that we need. The computer's representation of X is a finite set of points lying between the real numbers a and b.

$$a =< a_1, a_2, ...a_n <b$$

More precisely, X is partitioned into half-open intervals

$$[a_1, a_1+e), [a_2,a_2+e),.....[a_n, b=a_n+e]$$

such that each sub-interval $s_i=[a_i,a_i+e)$ is mapped (truncated) into the left-end point a_i, where e is a small positive number, and a_i are truncated numbers. Such partition defines an equivalence relation R on X. The pair (X, R) is called approximation space by Pawlak [10], [11]. The equivalence relation, x R y, means that x and y are truncated into the same number. Geometrically. it means that x and y are in the same sub-interval. We will call such R *an indiscernibility relation of radius e*. The quotient set X/R is a set of sub-intervals that are often represented by their left-end points. We will call e the radius of truncation.

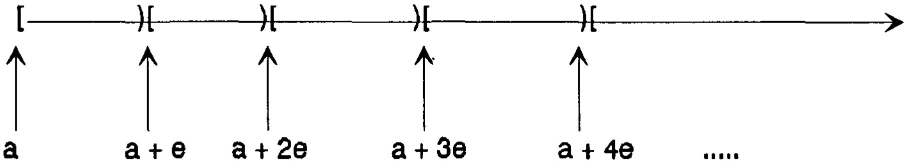

Figure 1

3. Patterns and Kolmogorov Complexity

Intuitively, a sequence with patterns can not be random; a random sequence can not have patterns. So we view patterns from the notion of randomness. Let us recall few points about randomness in algorithmic information theory. We will refer readers to [1], [2] for further details. Let x be a binary string (or sequence), and K(x), Kolmogorov complexity, be the length of shortest program that can generate the string x.

K(x) = min {length(p) : p is any conceivable program that generates the string x}

where the length of a program can be, say, the length of bit string of executable code.

Let f(n) be a function of n, which tends to ∞ as n goes to ∞. Let x be a string of length n, then x is said to be *Kolmogorov random*, if $K(x) \geq n - f(n)$. It is not too difficult to show that for every n, there is a random binary sequences of length n. In fact, it has been shown that[2]

Theorem The "measure" of the set of non-random sequences is zero, more precisely,

Card {x : length(x)=n and random}/ $2^n \leq 2^{-f(n)}$

where Card is the cardinal number.

If we take a non-random binary sequence as sequence with pattern, we have

Corollary "Almost all" finite binary sequences have no patterns.

This is a rather disturbing fact for Intrusion Detection Systems unless one appeals to human behaviors. Fortunately, we have positive results on soft patterns.

4. Soft Patterns in Sequences

In mathematical analysis, Weistrass theorem states that a sequence in a closed interval X has a convergent subsequence, say converges to point p. In other words, given an ε-neighborhood of p, there will be infinitely many terms of the sequence located inside the ε-neighborhood of p

Theorem Every sequence in a closed finite interval has a ε-neighborhood repeating subsequence.

If we truncate numbers with radius 2ε, the sequence will repeat the truncated p infinitely many times. In fact, if we observe that there are only finite truncated numbers in a bounded interval, and finite numbers of character strings within bounded length, then an infinite sequence of data in computers certainly will repeat some terms infinitely many times. The same consideration can be applied to high dimensional case, so we have

Theorem Every sequence of records in computers has a repeating subsequence.

One can view audit trail as a sequence of tuples in a relational database.

Theorem If the audit trail is long enough, then there is repeating records.

The repeating records are part of user's "signature".

5. Rules in Fuzzy Information Systems

Main focus in this section is on extracting rules from audit trails. Audit trails can be viewed as relational databases or better Pawlak information system with **continuous input**. We will be more interested in data so information systems are better framework than relational databases, especially, the entity integrity rule is hardly ever enforced in the audited data. Information systems have been studied intensively by Pawlak school [10]. The main contribution in this paper is extending Pawlak's methodology to fuzzy rough sets [13], [14]. Based on fuzzy view of rough sets, instead of exact rules, we obtain fuzzy rules. In audit trails, we need fuzzy rules, because of constant updating. Exact rules are too expensive to be updated. In our approach, each supporting case is weighted, that is, from the point of view of fuzzy set theory each case has a partial membership. We view an audit trail as a "dynamic" information system; the records are constantly inserted and faded away (aging). An information system in audit trails is a sliding "window".

5.1. Pawlak Information Systems

In this subsection, we use examples to illustrate the idea of how one can extract rules from data . First we have to introduce Pawlak's information system.

A Pawlak information system is a 4-tuple

$$S = (U, T, V, \rho)$$

where
U is the set of objects of S.
T is a set of attributes.
V = the union of all sets V_a of values of attributes a.
$\rho : U \times T \dashrightarrow V$,
called description function, is a map such that
$\rho(x,a)$ is in V_a for all x in U and a in V.

Let B be a non-empty subset of T. Let x, y be two objects. x and y are indiscernible by B in S , denoted by

$$x \cong y \ (\text{mod } B) \text{ if } \rho(x,q) = \rho(y,q) \text{ for every q.}$$

Obviously, ≅ is an equivalence relation, it will be called indiscernibility relation IND(B). The partition induced by B is called a classification of U generated by B. For a non empty subset of B of T, an ordered pair A = (U, B) is an approximation space. A definable set X will be called B-definable [10]. An information system (S, T, V, ρ) is called a **decision table** if T = C U D is a set of attributes, where C and D are disjoint non-empty subsets [11]. The elements in C are called conditional attributes. The elements in D are called decision attributes.

A relation/view instance is a snap shot of a relational database, which represents user's instant perception of entities or objects represented in the database. An information system is such an instance. We should note that the information systems is an extension of relational databases *without the entity integrity constraint.*

Table-1

ID#	LOCATION	TEST	POLL	LEVEL	NEW	CASE	RESULT
ID-1.	Houston	1	0	0	11	1	1
ID-2.	San Jose	1	0	0	11	1	1
ID-3.	Santa Clara	1	1	1	11	1	1
ID-4.	New York	0	1	1	10	0.7	1
ID-5.	Chicago	0	1	1	10	0.7	1
ID-6.	Los Angeles	0	1	1	10	0.7	1
ID-7.	San Franscico	0	1	1	10	0.7	1
ID-8.	Seattle	0	1	1	10	0.7	1
ID-9.	Philadelphia	0	1	1	10	0.7	1
ID-10.	Atlanta	0	1	1	10	0.7	1
ID-11.	St Louis	0	1	1	10	0.7	1
ID-12.	Cincinnati	0	1	1	10	0.7	1
ID-13.	Washington	0	1	1	12	1	2
ID-14	New Orleans	1	1	0	12	1	2
ID-15.	Baltimore	1	1	0	12	1	2
ID-16.	Boston	1	1	0	12	1	2
ID-17.	San Diego	1	1	0	12	1	2
ID-18.	Palo Alto.	1	1	0	23	1	3
ID-19.	Berkeley	1	0	0	23	1	3
ID-20.	Davis	1	0	0	23	1	3
ID-21.	Austin	1	0	0	23	1	3

From this relation, we will form two information systems, more precisely, two decision tables; they are adopted from (with changes) [12]

Example 1. Extracting Exact Rules

(5.1) We will consider an equivalence relation defined by the attribute RESULT, called decision attribute.

ID-i $\cong$ ID-j iff
ID-i.RESULT=ID-j.RESULT,

We have the following three equivalence classes, called decision classes

DECISION1={ID-1,ID-2,...,ID-12}={1},
DECISION2={ID-13,ID-14,...ID-17}={2},
DECISION3={ID-18,ID-19,..ID-21}={3}

(5.2) For the conditional attributes (NEW, CASE), we consider the following equivalence relation

ID-i $\cong$ ID-j iff
ID-i.NEW=ID-j.NEW, ID-i.CASE=ID-j.CASE,

Then we have the following three equivalence classes, called condition classes.

#1CASE1={ID-1, ID-2, ID-3},
#1CASE2={ID-4, ID-5,...., ID-12 },
#1CASE3={ID-13,ID-15,.....ID-17},
#1CASE4={ID-18,ID-15,.....ID-21}

Later, we will consider the case that each conditional class is a fuzzy set.

Comparing condition & decision classes, we get the inclusions

#1CASE1------------>DECISION1
#1CASE2------------>DECISION1
#1CASE3------------>DECISION2
#1CASE4------------>DECISION3

In other words, it discovers the following exact rules:

If NEW=11 and CASE=1, then RESULT=1
If NEW=10, and CASE=0.7, then RESULT=1
If NEW=12, and CASE=1, then RESULT=2
If NEW=23, and CASE=1, then RESULT=3

5.2. Fuzzy Rules in Information Systems

Example 2. Extracting Fuzzy Rules

In stead of giving an fuzzy information system, we give a fuzzy view of the information system. The equivalence classes are regarded as fuzzy sets, and hence we have derived fuzzy rules. Our results can be viewed as fuzzy version of [12]. The decision attributes be the same as in (5.1)

(5.3) We will consider the equivalence relation defined by conditional attributes {TEST, POLL, LEVEL}

ID-i ≅ ID-j iff
ID-i.TEST=ID-j.TEST,
ID-i.POLL=ID-j.POLL, and
ID-i.LEVEL=ID-j.LEVEL,

Then we have the following three condition classes

#2CASE1={ID-1, ID-2, ID-19, ID-20, ID-21},
#2CASE2={ID-3},
#2CASE3={ID-4,ID-5,.....ID-13}
#2CASE4={ID-18,ID-15,.....ID-21}

Comparing the condition classes with decision classes, we found

(5.3.1) one exact inclusion

#2CASE2 $\subseteq_{(0)}$ DECISION1,

So, it discovers the exact rule

If TEST=1, POLL=1, and LEVEL=1,
then RESULT=1---------------------(R1)

(5.3.2) fuzzy inclusions (see Appendix 7.3)

The equivalence class is a classical set, however, we will treat it as a fuzzy set, namely, it is represented by its characteristic function with real values in 0 or 1. The fuzzy inclusions are represented by the inequalities of membership functions. Further, we will allow certain errors as long as they are within the radius ε of tolerance(errors). In fact, we will call such inclusions ε-fuzzy inclusions, denoted by $\subseteq_\varepsilon$ [6], In this example, we choose ε=0.1. Then, we have an ε-fuzzy inclusion other than 5.3.1

#3CASE3 $\subseteq_{(0.1)}$ DECISION1.

To see the ε-fuzzy inclusion, write

Y=DECISION1,
X=#3CASE3,
Z=X∩Y={ID-4,ID-5,.....ID-12}

First note that if Z=X, then X ⊆ Y, so if we want to show that X $\subseteq_{(0.1)}$ Y, we have to show that Z and X are nearly equal. Let FZ =FX∩FY, FW=FX∪FY, which express the sets, Z=X∩Y, W=X∪Y, in terms of **characteristic functions** (membership functions). So we have to show that

$$\text{(Eq-1)} \quad \sum_{u=u_1}^{u_{21}} |FX(u)/Card(FW) - FZ(u)/Card(FW)| \leq \varepsilon$$

or equivalently

$$\text{(Eq-1')} \quad \sum_{u=u_1}^{u_{21}} |FX(u) - FZ(u)| \leq \varepsilon \bullet Card(FW)$$

$$\sum_{u=u_1}^{u_{21}} |FX(u)/13 - FZ(u)/13| = 1/13 \leq \varepsilon$$

where (1) u_i = ID-i, i=1,2,..21, are the records
(2) u is a variable that varies through the records, $u_1, u_2, \ldots u_{21}$
(3) Card is the (fuzzy set theoretical) cardinal numbers [15].

The left hand-sided sum (Eq-1) is called **deviation number.** So, the ε-fuzzy inclusion discovers the following approximate rules

If TEST=0, POLL=1, and LEVEL=1,
then (approximately) RESULT=1-------(R2)

In conclusion, we see that the exact view gives us an exact rule, while the fuzzy view give us two fuzzy rules (one exact, one fuzzy). Here we would like to comment that one could take the attitude that the two fuzzy membership functions FX and FZ are the different representations of the same fuzzy set (both are admissible membership functions). In next computation, we will take the aging into account.

5. 3 Sliding Window Information System (SWIS)

Since an audit data has a continuous input, we have to consider the ages of the data. We will consider an information system of sliding window, data are continuously come and faded(aging) away.

Aging Rule: For simplicity, we assume the record id is numbered by the time of its arrival. For example, ID-1 arrived at time 1 and ID-2 arrived at time 2, and so forth. The aging rule is described by an aging function which is function of time. In this example, the "age" of the newest record is 1, the next 20 records are 0.9, the 21st (in reverse chronological order) is 0.1, the 22nd record and so on are 0.

We will present two examples here to illustrate the idea

SWIS-Example 1. Extracting Rules from a Sliding Window Information System (SWIS). A SWIS is a fuzzy information system which consist of the pairs of record ids (from Table-1) and their ages (the degrees of memberships)

(ID-1, 0.0) (ID-2, 0.1), (ID-3, 0.9), (ID-4, 0.9) ...
(ID-21, 0.9),

together with two new data:

(ID-22, 0.9), (ID-23,1)

The new data are the tuples

(ID-22, San Macro, 1, 0, 0, 23, 1, 3),and

(ID-23, Hayward, 1, 0, 0, 23, 1, 3)

The decision attributes are the same as in (5.1) but the values may have changed. Use the same equivalence relation, we have the following three equivalence classes, called decision classes

DECISION1={ID-1,ID-2,...,ID-12}={1},
DECISION2={ID-13,ID-14,...ID-17}={2},
DECISION3={ID-18,ID-19,..ID-21,
ID-22, ID-23}={3}

(5.3.3) As before, we consider the equivalence relation defined by conditional attributes {TEST, POLL, LEVEL}, we have the following three condition classes

#3CASE1={ID-1, ID-2, ID-19, ID-20,
ID-21, ID-22, ID-23},
#3CASE2={ID-3},
#3CASE3={ID-4,ID-5,.....ID-13}
#3CASE4={ID-18,ID-15,.....ID-21}

Again, let ε=0.1. Then, we have a new ε-fuzzy inclusion other than (5.3.1)

$$\#3CASE3 \subseteq_{(0.1)} DECISION1.$$

To see the ε-fuzzy inclusion, write

Y=DECISION1,
X=#3CASE3,
Z=X∩Y={ID-4,ID-5,.....ID-12}

Let FZ =FX∩FY, and FW=FX∪FY, which express Z=X∩Y and W=X∪Y in terms of **aging functions** (membership functions). In such case the value of FX(u) in (Eq-1) is the "age" of the record u. Note that Card(FW) = 0.9*11=9.9. We can compute the formula of (Eq-1) as follows:

$$\sum_{u=u_1}^{u_{23}} |FX(u)g(u)/Card(FW)-FZ(u)g(u)/Card(FW)|$$

$$\le \sum_{u=u_1}^{u_{23}} |FX(u)g(u)/9.9 -FZ(u)g(u)/9.9| =0.9/9.9\le \varepsilon$$

Note that FX(u)g(u) is the membership function of the record u. Recall that the left-most sum is the deviation number. If the deviation number of a rule is fluctuated within the tolerance, such as ε, and other "signature" data (repeating records) are unchanged, then the system is normal, otherwise it may signal that there are intrusions. Full experimental results will be reported in the near future. For now, let us examine the case when there is an intrusion.

SWIS-Example 2. Blind-append-virus. In this example, we examine the case when a virus blindly repeat "infinitely" many times of a user's command. In other words, the same record repeatedly enter the audit trail. Let us first recall the **aging rules.** The "age" of the newest record is 1, the next 20 records are 0.9, the 21st (in reverse chronological order) is 0.1, the 22nd record and so on are 0.

Let a fuzzy information system be a set of following pairs

(ID-1, 0.0) (ID-2, 0.0), ... (ID-16, 0.1)
(ID-17, 0.9),..... (ID-36,0.9) (ID-37,1)

The new data from ID-22 to ID-28 are the "same":

(ID-22,	San Macro,	1,	0,	0,	23,	1,	3)
(ID-23,		1,	0,	0,	23,	1,	3)
	:						
(ID-37,		1,	0,	0,	23,	1,	3)

Proceed as before, the decision classes

DECISION1={ID-1,ID-2,...,ID-12}={1},
DECISION2={ID-13,ID-14,...ID-17}={2},
DECISION3={ID-18,ID-19,..,ID-37}={3}

(5.3.3) The condition classes are

#4CASE1={ID-1, ID-2, ID-19, ID-20,
ID-21, ..., ID-37},
#4CASE2={ID-3},
#4CASE3={ID-4,ID-5,.....ID-13}
#4CASE4={ID-18,ID-15,.....ID-21}

The ε-fuzzy inclusions are

(Inc-1) #3CASE3 $\subseteq_{(0.1)}$ DECISION1.

(Inc-2) #3CASE1 $\subseteq_{(0.1)}$ DECISION3

Note that Card(FW) = 0.0, so the inequality may not be true

$$\sum_{u=u_1}^{u_{37}} |FX(u)g(u) -FZ(u)g(u)| \ ?\ \varepsilon*Card(FW)=0.0$$

Note that FX(u)g(u) is the membership function of the record u. (Eq-1') no longer stay within the radius of tolerance. The fuzzy inclusion is no longer true. So *the old fuzzy rule disappears from the sliding window*. Moreover, *a new rule is appearing*. Write D=DECISION3, C=#3CASE1, Z=C∩D={ID-4, ID-5,.....ID-12}. Note that Card(FW) = 0.9*19+1=18.1

$$\sum_{u=u_1}^{u_{37}} |FC(u)g(u)/Card(FW) -FZ(u)g(u)/Card(FW)| \leq$$

$$\sum_{u=u_1}^{u_{23}} |FC(u)g(u)/18.1 -FZ(u)g(u)/18.1| = 0.9/18.1 \leq \varepsilon$$

Note that FC(u)g(u) is the membership function of the record u. (Eq-1) satisfy the radius of tolerance. So a new rule (or a new pattern) is forming. The user's "signature" changed, so an intrusion is occurring. We will not address the complexity problem. However, as we have discuss in the beginning part of this paper, we have shown that , based on rough set theory, there are only finitely many different records in a sequence of records. We believe that the complexity problem can be controlled via rough set representation of data. We will discuss this in near future.

6. Applications to Audit Trails

From Section 4, we are assured that, we can find the repeating records for each user. We could keep a log on the following information:

(a) The repeating records, and *its frequency*, and *occurrences*

(b) The *fuzzy rules* in the audit data.

Now, as the data are continuously collected and faded away (aging), the sliding window slides. If the fuzzy rules stay constant, and deviation numbers fluctuate within the tolerance level, then the system is normal. Otherwise, any significant change on the data (a) or (b) signal an abuse or misuse of the system. So the fuzzy-rough set methodology provides us a foundation for anomaly detection.

7. Appendix- Rough Sets

7.1. Equivalence Relations

A binary relation is an equivalence relation iff it is reflexive, symmetric and transitive. For every equivalence relation there is a partition and vice versa. Let R be a given equivalence relation over U. The family of all equivalence classes is a set, it is called quotient set and denoted by U/R. There is a natural projection from U to U/R.

NQ: U ------> U/R.

defined by NQ(u) = [u] (read as natural quotient), where [u] is the equivalence class containing u. We should note here that [u] has dual roles; it is an element, not a subset, of U/R, but it is a subset of U. In [2], elements in U/R are called names of equivalence classes. Let us denote the complete inverse image of NQ by

INV.NQ(q) = {u : NQ(u) = q} =[u]

or more generally, for a subset X of U/R

INV.NQ(X) = {u : NQ(u) is in X}

Note that INV.NQ(q) is an equivalence class and INV.NQ(X) is a union of equivalence classes.

Example 1. Let Z be integers. Let R denote the equivalence relation called congruence mod m. That is,

x R y if x - y is divisible by m.

Let m = 4. Then the equivalence classes are

[0] = {... -8, -4, 0, 4, 8...}
[1] = {... -7, -3, 1, 5, 9,..}
[2] = {... -6, -2, 2, 6, 10,.}
[3] = {... -5, -1, 3, 7, 11,.}

In other words, [0], [1], [2], [3] is a partition for the integers Z. The quotient set of this equivalence relation is denoted by Zm. Z4 = {[0], [1], [2], [3]}.

7.2. Rough Sets

Let U be the universe of discourse. Let RCol be a finite Collection of equivalence Relations R over U. In general we will use Pawlak's terminology and notations . An ordered pair

K=(U, RCol)

is called a knowledge base over U (In most cases, there is only one equivalence relation R in RCol, so K = (U, R)). A subset X of U is called a concept. For an equivalence relation R, an equivalence class is called R-elementary concept, R-elementary set, R-basic category or R-elementary knowledge (about U in K). The empty set is assumed to be elementary. A set which is a union of elementary sets is called R-definable or R-exact. A finite union is called composed set in U. The set of equivalence classes is the quotient set U/R. There is a neat correspondence between the elementary sets of U and the quotient set U/R. Each elementary set in U corresponds to an element in U/R.

Let SCol be a nonempty SubCollection of RCol. The intersection of all equivalence relations in SCol, denoted by IND(SCol), is an equivalence relation and will be called an indiscernibility relation over SCol. The quotient set U/IND(SCol) will be abbreviated by U/SCol. Equivalence classes of IND(SCol) are called basic categories (concepts) of knowledge K. A concept X is exact in the knowledge base K if there exists an equivalence relation R in IND(K) such that X is R-exact, where IND(K) is the collection of all possible equivalence relations in K, that is,

IND(K)={IND(SCol): for all SCol's in RCol}.

For each X, we associate two subsets, upper and lower approximation:

L_APP(X)= {u : [u] is a subset of X}
U_APP(X)= {u : [u] and X has non-empty intersection}

A subset X of U is definable iff U_APP(X) = L_APP(X). The lower approximation of X in U is the greatest definable set in U contained in X. The upper approximation of X in U is the least definable set in U containing X.

As Pawlak pointed out that the equivalence classes form a topology for U (it will be called Pawlak topology). So we can rephrase the upper and lower approximations as follows:

L_APP(X)= Interior point of X
 = The largest open set contained in X
U_APP(X)= Closure of X
 = The smallest closed set containing X.

Rough set theory serves two functions: one is a generalization of the equality which leads to classification, the other is the approximation in Pawlak topology.

8. Appendix -Fuzzy Sets

The theory of fuzzy sets deals with subsets where the membership function is real valued, not Boolean valued. Intuitively the fuzzy subsets have no well defined boundaries in the universe of discourse. Let U be the universe of discourse. Then a fuzzy set FX is an ordered pairs:

FX = (U, FX)

where FX: U ---> [0,1] is a function.. If both FX(0) and FX(1) are nonempty, we call the fuzzy set normal [16]. Note that FX is a fuzzy set and FX() is a membership function of FX. When context is clear, we may use FX both as the fuzzy set or the membership function. If the membership function assumes only real values 0 and 1, the fuzzy sėt is a classical set. An element x is said to be fuzzily belonged to FX if FX(x)>0 and x is said to be absolutely not belong to FX if FX(x)=0.

8.1. Quasi Classical Sets

Let X be a classical set. We would like to consider the membership function

c*X: U ------> [0,1]

defined by (c*X)(u) = c*(X(u)) for constant c, where * is the multiplication of real numbers. Then c*X is a special type of fuzzy set, we will call it quasi classical set. The meaning of such quasi classical set is that an object x in U is either not in X or the degree (possibility, probability) of its membership is c. We also would like to consider the "union" of quasi classical sets:

(a*X U a*Y)(x) = MAX (a*X(x), b*Y(x))

The union of quasi-classical sets are the so-called "step functions"

8.2. Fuzzy Rough Sets

Let R be an equivalence relation over U. Let FCol(U/R) be the Collection of all Fuzzy sets over U/R. Then the natural projection induces a subfamily of fuzzy sets on U.

$$U \xrightarrow{NQ} U/R \xrightarrow{FX} [0,1]$$

SubFCol(U) = {NQ*FX: FX is in FCol(U/R)}

where * is the composition of functions. This subfamily SubFCol is the family of all R-exact fuzzy sets. SubFcol is precisely, the "step functions" We would like to have more explicit description of this SubFCol of fuzzy sets on U. Let the membership function of the equivalence classes (R-elementary sets) be

EC_i: U ------------> [0, 1], i= 1,2,..n.

Since ECi(i-th equivalence class) is a classical set, its membership function assumes 0 and 1 only; it may be referred to as classical equivalence class. A fuzzy set in U

FX: U -------> [0,1]

is R-definable iff FX is in SubFCol(U). That is, FX is constant function on every EC_i. In other words, FX is a linear sum of classical sets. Using functional notations, FX is R-definable iff

$$FX = c_1*Ec_1 + c_2*EC_2 + ...c_n*EC_n.$$

The R-definable fuzzy set may also be called R-exact. A fuzzy set (concept) is R-undefinable iff it is not R-definable; it may also be called R-inexact. For each FX, we associate two subsets, upper and lower approximation:

U_APP(FX)= inf{FY: FX ≤ FY for all FY in CQE}
L_APP(FX)= sup{FY: FX ≥FY for all FY in CQE}

Such pairs are called fuzzy rough sets.

8.3 Real World Fuzzy Sets

Let U= $\{u_1,u_2,.....u_n\}$be the universe. For a given small number ε (called radius of tolerance). Let FX and FY be two membership functions. Then both functions are said to be representing the same real world fuzzy set, if for given ε,

$$\sum_{u=u_1}^{u_n} |FX(u) - FY(u)|/|Card(FW)| \leq \varepsilon$$

where FW= FX∪FY, and Card(FW) is the cardinality of fuzzy set FW. Roughly, the "total difference" is relatively small compared to "the total measure. However, this admissibility is *not* an equivalence relation.

References

[1] Michael van Lambalgen, The Axiomatization of Randomness, The Journal of Symbolic logic, Vol. 55, No 3, Sept., 1990.

[2] Ming Li and Paul Vitanyi, Two decades of applied Kolmogorov Complexity, Proceeding of third IEEE Structure in Complexity theory Conference, 1988.

[3] T. F. Lunt and Ann Tamaru, F. Gillman, R. Jagganathan, C. Jatali, H. Javitz, and A. Valdes, and P. Neumann, A real-time Intrusion Detection Expert System. SRI technical report, 1990

[4] H. Javitz, A. Valdes, T. F. Lunt and Ann Tamaru, Next Generation Intrusion Detection Expert System. SRI technical report, 1993

[5] T. Y. Lin, Rough Patterns and Intrusion Detection Systems, Journal of Foundation of Computer Sciences and Decision Supports, 1993

.[6] T. Y. Lin, Coping with imprecise information -- "fuzzy logic", Downsiaing Expo, Santa Clara, Aug. 4-6, 1993.

[7] L. Zadeh, Soft Computing, International Simulation Technology Multi-Conference, Nov7-10, San Francisco, 1993, SIMTEC'93

[8] M. Wildberger, AI & Simulation, Simulation, Vol 62., January 1994,

[9] Z. Pawlak, Hard and Soft Sets, Proceeding of The International EWorkshop on Rough Sets and Knowledge Discovery, Banff, 1993.

[10] Z. Pawlak , Rough sets. Int. J. Computer and Information Sci. 11, 341-356, 1982

[11] Z. Pawlak, Rough sets - Theoretical Aspects of Reasoning about Data, Kluwer Academic Publishers, 1991.

[12] W. Ziarko, 1993, Variable Precision Rough Set Model, Journal Computer and System Science, 39-58, 1993

[13] Didier Dubois and Henri Prade. Rough fuzzy sets and fuzzy rough sets. International Journal of General Systems, pages 191-209, 1990.

[14] T.Y. Lin, Topological and Fuzzy Rough Sets, Kluwer Academic Publishers, 1992 (A chapter of Decision support by Experience - Application of the Rough Sets Theory, R. Slowinski (ed.)

[15] Abraham Kandel (1986), Fuzzy Mathematical Techniques with Applications, Addision-Wesley, Reading Massachusetts, 1986

[16] Zimmermann, Fuzzy Set Theory --and its Applications, Second Ed., Kluwer Acdamic Publisher, 1991.