World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:9, No:6, 2015

Studies of Rule Induction by STRIM from the Decision Table with Contaminated Attribute Values from Missing Data and Noise — In the Case of Critical Dataset Size —

Tetsuro Saeki, Yuichi Kato, Shoutarou Mizuno

Open Science Index, Computer and Information Engineering Vol:9, No:6, 2015 publications.waset.org/10001804/pdf

***Abstract*—**STRIM (Statistical Test Rule Induction Method) has been proposed as a method to effectively induct if-then rules from the decision table which is considered as a sample set obtained from the population of interest. Its usefulness has been confirmed by simulation experiments specifying rules in advance, and by comparison with conventional methods. However, scope for future development remains before STRIM can be applied to the analysis of real-world data sets. The first requirement is to determine the size of the dataset needed for inducting true rules, since finding statistically significant rules is the core of the method. The second is to examine the capacity of rule induction from datasets with contaminated attribute values created by missing data and noise, since real-world datasets usually contain such contaminated data. This paper examines the first problem theoretically, in connection with the rule length. The second problem is then examined in a simulation experiment, utilizing the critical size of dataset derived from the first step. The experimental results show that STRIM is highly robust in the analysis of datasets with contaminated attribute values, and hence is applicable to real-world data.

***Keywords*—**Rule induction, decision table, missing data, noise.

I. Introduction

ROUGH Sets theory as introduced by Pawlak [1] provides a database called the decision table, with various methods for inducting if-then rules, and determination of the structure of rating and/or knowledge in the database. Such rule induction methods are needed for disease diagnosis systems, discrimination problems, decision problems, and other aspects, and consequently many effective algorithms for rule induction by rough sets have reported in the literature [2] – [7]. However, these methods and algorithms have paid little attention to mechanisms of generating the database, and have generally focused on logical analysis of the given database. This narrows the scope of the analysis. In a previous study [8] we devised a model of data generation for the database, proposed a statistical rule induction method, and presented an algorithm named STRIM. In a simulation experiment based on the model of the data generation with if-then rules specified in advance, STRIM successfully inducted the specified true rules from different databases generated from the same specified rules [8], whereas the conventional methods [2], [4], [7] could not.

In contrast to the ideal data set generated by the simulation experiment, real-world data sets are often small, and/or contain missing and contaminated values in the decision table. The problem of the size of the dataset has seldom been studied to date, as the conventional methods mainly focus on the logical aspect of the rule induction problems and their related features, and hence the issue of dataset size has been of little interest. Conversely, many studies have proposed methods for handling tables containing missing values. One study [9] summarizes this work, based on reference to about fifty studies. The conclusions reached in [9] were:

1) Those studies could be divided into two types: sequential or parallel methods.
2) In the sequential methods, missing attribute values are first replaced by known values during preprocessing, and the rules are then inducted using the ordinary rule induction method [10]. Over ten preprocessing methods have been used [9].
3) In parallel methods, no preprocessing occurs: i.e., the rules are inducted from the original table by devising the ordinary rule induction methods, in order to accept and handle the missing attribute values. The parallel method distinguishes two types of missing attribute values: ”lost” and ”do not care” conditions. The former condition is handled as a value outside the range of the attribute values, whereas the latter is handled as any value within the range.

After summarizing the rule induction method by STRIM, our present paper theoretically studies the dataset size problem, and derives the expression $N_{lst}(w, RL)$ of the minimum size of the data set which can induct a true rule with the rule length RL and the probability w [%]. This paper also examines contamination of the decision table having not only missing values in the condition attributes but also contaminated values in the decision attribute, by applying it as a parallel method to examine the capacity of STRIM at the critical data size $N_{lst}(w = 100, RL = 2)$. The validity of the expression $N_{lst}(w, RL)$ and the capacity of STRIM for contaminated datasets are confirmed by a simulation experiment. These studies yield

T. Saeki is with Yamaguchi University, 2-16-1 Tokiwadai, Ube city, Yamaguchi 755-8611, Japan (corresponding auther provide e-mail: tsaeki@yamaguchi-u.ac.jp).
Y. Kato and S. Mizuno are with Shimane University, 1060 Nishikawatsu-cho, Matsue city, Shimane 690-8504, Japan (e-mail: ykato@cis.shimane-u.ac.jp).

Open Science Index, Computer and Information Engineering Vol:9, No:6, 2015 publications.waset.org/10001804/pdf

useful information for analyzing real-world datasets, since the conventional method can give no such guiding principle. The results further illustrate the advantages of STRIM over the conventional method.

II. Data Generation Model and the Decision Table

Rough Sets theory is used for inducting if-then rules hidden in the decision table S. S is conventionally denoted $S = (U, A = C \cup D, V, \rho)$. Here, $U = \{u(i)|i = 1, ..., |U| = N\}$ is a sample set, A is an attribute set, $C = \{C(j)|j = 1, ..., |C|\}$ is a condition attribute set, $C(j)$ is a member of C and a condition attribute, and D is a decision attribute. V is a set of attribute values denoted by $V = \bigcup_{a \in A} V_a$ and is characterized by an information function ρ: $U \times A \rightarrow V$. For example, if $a = C(j) \in A$ $(j = 1, ..., |C|)$ then $V_a = \{1, 2, ..., M_{C(j)}\}$ and if $a = D$ then $V_a = \{1, 2, ..., M_D\}$. Table I shows an example where $|C| = 6$, $M_{C(j)} = 6$, $M_D = 6$, $\rho(x = u(1), a = C(1)) = 6$, $\rho(x = u(2), a = C(2)) = 2$.

STRIM considers the decision table to be a sample dataset obtained from an input-output system including a rule box (Fig. 1), and a hypothesis regarding the decision attribute values (Table II). A sample $u(i)$ consists of its condition attributes values of $|C|$-tuple $u^C(i)$ and its decision attribute $u^D(i)$. $u^C(i)$ is the input into the rule box, and is transformed into the output $u^D(i)$ using the rules contained in the rule box and the hypothesis. For example, specify the following rules in the rule box:

$R(d)$: if Rd then $D = d$, $(d = 1, ..., M_D = 6)$,

where $Rd = (C(1) = d) \wedge (C(2) = d) \vee (C(3) = d) \wedge (C(4) = d)$. Generate $u^C(i) = (v_{C(1)}(i), v_{C(2)}(i), ..., v_{C(|C|)}(i))$ of $u(i)$ by use of random numbers with a uniform distribution, and then $u^D(i)$ is determined using the rules specified in the rule box and the hypothesis.

In contrast, $u(i) = (u^C(i), u^D(i))$ is measured by an observer, as shown in Fig. 1. Existence of $NoiseC$ and $NoiseD$ leads to missing values in $u^C(i)$, and changes $u^D(i)$ to create other values of $u^D(i)$, respectively. This model is closer to the real-world system. However, Table I shows an example generated by this specification without both noises, for a plain explanation of the system. Inducting if-then rules from the decision table is then identifying the rules in the rule box, by use of the set of observed inputs-outputs $(u^C(i), u^D(i))$ $(i = 1, ..., |U| = N)$.

III. Summaries of Rule Induction Procedures by STRIM

STRIM inducts if-then rules from the decision table through two processes, in separate stages. The first stage process is that of statistically discriminating and separating the set of indifferent data from the set of uniquely determined or conflicted data in the decision table (See Table II). Specifically, assume $CP(k) = \bigwedge_j (C(j_k) = v_j) (\in V_{C(j_k)})$ as the condition part of the if-then rule, and derive the set $U(CP(k)) = \{u(i)|u^{C=CP(k)}(i)\}$. Also derive $U(m) = \{u(i)|u^{D=m}(i)\}$ $(m = 1, ..., M_D)$. Calculate the distribution f: $(n_1, n_2, ..., n_{M_D})$ of the decison attribute values of $U(CP(k))$, where $n_m = |U(CP(k)) \cap U(m)|$ $(m = 1, ..., M_D)$. If the assumed $CP(k)$ does not satisfy the condition $U(Rd) \supseteq U(CP(k))$ (sufficient condition of specified rule Rd) or $U(CP(k)) \supseteq U(Rd)$ (necessary condition), $CP(k)$ only generates the indifferent data set based on Hypothesis 2 in Table II, and the distribution f does not have partiality. Conversely, if $CP(k)$ satisfies either condition, f has partiality, since $u^D(i)$ is determined by Hypothesis 1 or 3. Accordingly, whether f has partiality or not determines whether the assumed $CP(k)$ is neither a necessary nor sufficient condition. Whether f has partiality or not can be determined objectively by statistical test of the following null hypothesis $H0$ and its alternative hypothesis $H1$:

$H0$: f does not have partiality. $H1$: f has partiality.

Table III shows the number of examples of $CP(k)$, $(n_1, n_2, ..., n_{M_D})$ and an index of the partiality by z derived from Table I with $N = 10000$, in order to illustrate this concept. For example, the first row means: 100000 denotes $CP(k = 1) = (C(1) = 1)$ (the rule length is $RL = 1$) and its corresponding $f = (495, 231, 231, 254, 248, 245)$ and $z = 13.75$, where

$$z = \frac{n_d + 0.5 - np_d}{(np_d(1 - p_d))^{0.5}}, \quad (1)$$

$n_d = \max(n_1, n_2, ..., n_{M_D} = n_6)$, $(d \in \{1, 2, ..., M_D = 6\})$, $p_d = Pro(D = d)$, $n = \sum_{m=1}^{M_D} n_m$. In principle, $(n_1, n_2, ..., n_{M_D})$ obeys a multinomial distribution that is adequately approximated by the standard normal distribution, by use of n_d under the condition (testing condition): $p_d n \geq 5$ and $n(1-p_d) \geq 5$ [11]. In the same way, the fourth row 110000 denotes $CP(k = 4) = (C(1) = 1) \wedge (C(2) = 1)$ $(RL = 2)$, the eighth 110400 is $(C(1) = 1) \wedge (C(2) = 1) \wedge (C(4) = 4)$ $(RL = 3)$, and so on. Here, if we specify a standard of the significance level such as $z \geq z_\alpha = 3.0$ and reject $H0$, then the assumed $CP(k)$ becomes a candidate for the rules in the rule box.

The second stage process arranges the set of rule candidates derived from the first process, and finally estimates the rules in the rule box, since some candidates may satisfy the relationship: $CP(ki) \subseteq CP(kj) \subseteq CP(kl) \cdots$. For example, in the case 100000 $\supset$ 110000 $\supset$ 110400 (see Table III). The basic notion is to represent the $CP(k)$ of the maximum z; that is, the maximum partiality. In the above example, STRIM selects a $CP(k)$ of 110000, which by chance coincides with the rule specified in advance. Fig. 2 shows the STRIM algorithm [8].

Table IV shows the estimated results for Table I with $N = 10000$. STRIM inducts all of twelve rules specified in advance, and one extra rule. However, there are clear differences between them in their indices of accuracy and coverage. A simulation experiment in other work [8] also showed that conventional methods such as LEM2 [4] and FDMM [6], [7] with lower approximation could barely induct the significant rules. The rules inducted by these methods were highly dependent on the sample set, but STRIM clearly resolved these problems.

TABLE I
AN EXAMPLE OF A DECISION TABLE

U	$C(1)$	$C(2)$	$C(3)$	$C(4)$	$C(5)$	$C(6)$	D
1	6	1	4	4	6	5	4
2	1	2	2	3	1	4	2
3	3	5	3	5	5	6	4
4	5	2	4	3	1	5	3
5	2	6	5	5	2	1	5
...	...	...	...	...	...	...	...
$N-1$	1	2	5	2	6	5	4
N	2	1	6	6	3	4	6

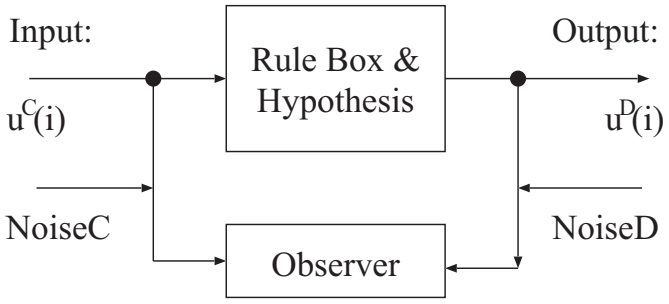


Fig. 1. Rough Sets system contaminated with noise: Input is a tuple of the condition attributes' value and its output is the decision attribute's value

TABLE II
HYPOTHESIS WITH REGARD TO THE DECISION ATTRIBUTE VALUE

Hypothesis 1	$u^C(i)$ coincides with $R(d)$, and $u^D(i)$ is uniquely determined as $D=d$ (uniquely determined data).
Hypothesis 2	$u^C(i)$ does not coincide with any $R(d)$, and $u^D(i)$ can only be determined randomly (indifferent data).
Hypothesis 3	$u^C(i)$ coincides with several $R(d)$ $(d=d1,d2,...)$, and their outputs of $u^C(i)$ conflict with each other. Accordingly, the output of $u^C(i)$ must be randomly determined from the conflicted outputs (conflicted data).

IV. CONSIDERATION ON TESTING CONDITION

As described in Section III, the data set applicable to STRIM must satisfy the testing condition: $p_d n \geq 5$ and $n(1-p_d) \geq 5$. The least number satisfying the condition is denoted with $N0$, and then consider the following event with a given probability w:

$$P(n \geq N0) \quad = \quad P(z \geq z0) = w \tag{2}$$

Here, $z = \dfrac{n+0.5-Np_c}{\sqrt{Np_c(1-p_c)}}$, $z0 = \dfrac{N0+0.5-Np_c}{\sqrt{Np_c(1-p_c)}}$. $p_c = P(C = CP(k)) = \prod_j P(C(j_k) = v_k)$ is the outcome probability of the condition part $CP(k)$ in the decision table. For example, if $CP(K) = (C(1)=1) \wedge (C(2)=1)$ $(RL=2)$ then $p_c = P(C(1)=1) \cdot P(C(2)=1)$. Assuming that z obeys the standard normal distribution, $z0$ is explicitly determined, and the least N denoted with N_{lst} satisfying (2) is given by:

$$N_{lst} \quad = \quad \frac{-b \pm \sqrt{b^2-4ac}}{2a} \tag{3}$$

where $+$: $z0 \leq 0$, $-$: $z0 > 0$, $a = p_c^2$, $b = -\{(2p_c(N0+0.5) + z0^2 p_c(1-p_c)\}$ and $c = (N0+0.5)^2$.

Accordingly, N_{lst} in (3) is mainly determined by parameters w and RL. So let us denote N_{lst} in (3) with $N_{lst}(w, RL)$.

Fig. 3 shows $N_{lst}(w, RL)$ evaluated by (3) at $w = 0.1$ [%] $(z0 = 3.0)$, $= 2.3$ [%] $(z0 = 2.0)$, $= 15.9$ [%] $(z0 = 1.0)$, $= 50.0$ [%] $(z0 = 0.0)$, $= 84.1$ [%] $(z0 = -1.0)$, $= 97.7$ [%] $(z0 = -2.0)$, $= 99.9$ [%] $(z0 = -3.0)$ every $RL = 1$, 2 and 3 in the specification of section II; where $P(C(j) = v_k) = 1/6$ ($j = 1$, ..., $|C| = 6$). For example, Fig. 3 yields the following useful information:

1) Supposing $RL = 2$, $N = 1865$ at least is needed to induct true rules with the probability of almost $w = 100$ [%]. This meaning is denoted with $1865 = N_{lst}$ $(w = 100$ [%], $RL = 2)$.
2) If a data set of $N = 1000$ is given, then the probability of inducting the true rules with $RL = 2$ is estimated to be about $w = 30$ [%]. This meaning is denoted with 30 [%] $= w = N_{lst}^{-1}$ $(N_{gvn} = 1000$, $RL = 2)$.

To confirm the consideration outlined in this section, a simulation experiment was conducted using the decision table containing samples of $N = 10000$ generated in section II, and the following procedures:

Step 1: Randomly select samples by N_{lst} $(w, RL = 2)$ from the decision table $(N = 10000)$, and form a new decision table;

Step 2: Apply STRIM to the new table, and count the number of inducted true rules specified in advance;

Open Science Index, Computer and Information Engineering Vol:9, No:6, 2015 publications.waset.org/10001804/pdf

Open Science Index, Computer and Information Engineering Vol:9, No:6, 2015 publications.waset.org/10001804/pdf

TABLE III
AN EXAMPLE OF A CONDITION PART AND CORRESPONDING FREQUENCY OF THEIR DECISION ATTRIBUTE VALUES

trying $CP(k)$	$C(1)$	$C(2)$	$C(3)$	$C(4)$	$C(5)$	$C(6)$	$(n_1, n_2, ..., n_6)$	z
1	1	0	0	0	0	0	(495, 231, 231, 254, 248, 245)	13.75
2	2	0	0	0	0	0	(247, 491, 244, 229, 229, 254)	13.64
3	0	0	0	0	0	5	(287, 246, 263, 286, 300, 295)	1.38
4	1	1	0	0	0	0	(253, 4, 2, 5, 7, 2)	33.78
5	1	2	0	0	0	0	(50, 51, 45, 61, 48, 47)	1.72
6	2	1	0	0	0	0	(45, 53, 51, 50, 34, 51)	0.98
7	2	2	0	0	0	0	(5, 244, 5, 1, 5, 4)	33.11
8	1	1	0	4	0	0	(41, 0, 0, 5, 0, 0)	13.39
9	1	1	0	5	0	0	(39, 0, 0, 0, 7, 0)	12.59
10	2	1	0	0	0	6	(6, 17, 10, 7, 4, 8)	3.29
11	2	2	1	0	0	0	(5, 44, 0, 0, 0, 0)	13.93
12	2	2	2	0	0	0	(0, 44, 0, 0, 0, 0)	15.03
13	3	3	0	6	0	0	(0, 46, 0, 0, 0, 2)	14.91
14	0	0	4	5	5	0	(8, 6, 6, 10, 13, 4)	2.22
15	0	0	4	5	6	0	(8, 7, 11, 8, 9, 6)	1.13
16	0	0	4	5	0	1	(7, 11, 4, 8, 9, 6)	1.60
17	0	0	4	5	0	3	(5, 8, 7, 10, 11, 9)	1.20
18	0	0	4	5	0	6	(6, 7, 6, 9, 8, 7)	0.95
19	0	0	4	6	2	0	(13, 8, 7, 7, 4, 6)	2.40
20	0	0	4	6	4	0	(10, 7, 9, 7, 13, 7)	1.72
21	0	0	4	6	5	0	(15, 10, 5, 9, 8, 6)	2.46
22	0	0	0	6	6	5	(7, 7, 8, 4, 9, 12)	1.83

TABLE IV
RESULTS OF ESTIMATED RULES FOR THE DECISION TABLE IN TABLE I (WITHOUT NOISE) BY STRIM

estimated $R(i)$	$C(1)$	$C(2)$	$C(3)$	$C(4)$	$C(5)$	$C(6)$	D	$(n_1, ..., n_6)$	p-value (z)	accuracy	coverage
1	3	3	0	0	0	0	3	(6, 4, 280, 6, 2, 2)	0(35.71)	0.933	0.168
2	0	0	5	5	0	0	5	(4, 3, 6, 3, 278, 7)	0(35.31)	0.924	0.168
3	0	0	4	4	0	0	4	(4, 6, 7, 267, 7, 3)	0(34.19)	0.908	0.158
4	4	4	0	0	0	0	4	(5, 4, 4, 255, 3, 2)	0(34.10)	0.934	0.151
5	6	6	0	0	0	0	6	(4, 2, 1, 4, 3, 250)	0(34.10)	0.947	0.149
6	0	0	3	3	0	0	3	(2, 3, 246, 3, 1, 2)	0(34.10)	0.957	0.147
7	1	1	0	0	0	0	1	(253, 4, 2, 5, 7, 2)	0(33.78)	0.927	0.152
8	0	0	1	1	0	0	1	(247, 3, 6, 3, 3, 5)	0(33.34)	0.925	0.149
9	2	2	0	0	0	0	2	(5, 244, 5, 1, 5, 4)	0(33.11)	0.924	0.148
10	0	0	6	6	0	0	6	(5, 4, 5, 3, 3, 243)	0(33.04)	0.924	0.145
11	0	0	2	2	0	0	2	(4, 237, 5, 5, 5, 5)	0(32.22)	0.908	0.144
12	5	5	0	0	0	0	5	(3, 1, 5, 6, 225, 4)	0(31.75)	0.922	0.136
13	0	0	0	0	1	2	3	(50, 36, 65, 45, 38, 35)	3.61e-4(3.38)	0.242	0.039

Step 3: Repeat Step 1 and Step 2 Nr times;
Step 4: Calculate the rate of true rules inducted out of Nr trials.

Fig. 4 shows the comparison of $N_{lst}(w, RL = 2)$ ($w = 0.1$ [%] ($z0 = 3.0$), $= 2.3$ [%] ($z0 = 2.0$), ..., $= 99.9$ [%] ($z0 = -3.0$)) between theoretical values studied in this section and the experimental values obtained from the above procedures by $Nr = 100$. The experimental value adequately represents the theoretical value, and confirms the validity of the theoretical considerations.

V. EXPERIMENTAL STUDIES ON MISSING VALUES IN THE CONDITION ATTRIBUTE

Missing values in parallel methods are of two types, as noted in section I: "lost" and "do not care" values. These are distinguished and denoted by "?" and "*" respectively [9]. Modified STRIM (denoted by mSTRIM) can accept and handle missing values in accordance with reference [9] as follows: With respect to $\forall x \in U$ and $\forall a \in C(j_k)$ of $CP(k)$, if $\rho(x, a) =$? then $x \notin U(CP(k))$ and if $\rho(x, a) =$ * then $x \in U(CP(k))$. A simulation experiment similar to that described in section IV was conducted to examine how mSTRIM can accept and handle the missing values against the data set of the critical size, by changing Step1, and adding the following procedures:

Step1: Randomly select $u(i)$ by $1865 = N_{lst}(w = 100, RL = 2)$ from the decision table ($N = 10000$) in Section II, and determine whether $u(i)$ has a number of N_{miss} of missing values with the probability q [%] (mixing rate of noise). If $u(i)$ has the N_{miss} of missing values, then randomly select the N_{miss} of condition attributes, and replace them with "?" or "*", and build a new decision table.

Fig. 5 shows the experimental results by the rate of true rules inducted by mSTRIM and $Nr = 100$ with respect to q [%] separately by $N_{miss} = 1$ (a), $= 2$ (b), $= 3$ (c). The rate by "*" and "?" is denoted by ♦ and ■ respectively. Based on this figure, mSTIRM is very robust against the missing noise of both "*" and "?" type, since mSTRIM can induct true rules by more than 95 [%] until $q =$ about 60 [%] at $N_{miss} = 1$,

```
int main(void) {
int rule[|C|]={0,...,0}; //initialize trying rules
int tail=-1; //initial vale set
input data; // set decision table
rule_check(tail,rule); // Stage 1
make Pyramid(l) (l=1,2,...) so that every r(k) belongs to one Pyramid at least;
// Stage 2, r(k): rule candidate
make rePyramid(l) (l=1,2,...); // Stage 2
reduce rePyramid; // Stage 2
} // end of main

int rule_check(int tail,int rule[|C|]) { // Stage 1
  for (ci=tail+1; cj<|C|; ci++) {
    for (cj=1; cj<=|C[ci]|; cj++) {
      rule[ci]=cj; // a trying rule sets for test
      count frequency of the trying rule; // count n1, n2, ...
      if (frequency>=N0) { //sufficient frequency ?
        if (|z|>3.0) { //sufficient evidence ?
          store necessary data such as rule, frequency of n1 and n2, and z
        } // end of if |z|
      rule_check(ci,rule);
      } // end of if frequency
    } // end of for cj
    rule[ci]=0; // trying rules reset
  } // end of for ci
} // end of rule_check
```

Fig. 2. An algorithm for STRIM (Statistical Test Rule Induction Method)

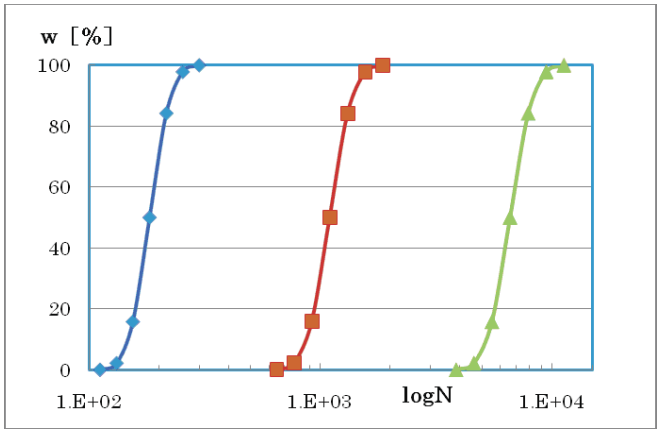


Fig. 3. Theoretical $N(w, RL)$ evaluated by (3) at $w = 0.1$ [%] ($z0 = 3.0$), $= 2.3$ [%] ($z0 = 2.0$), $= 15.9$ [%] ($z0 = 1.0$), $= 50.0$ [%] ($z0 = 0.0$), $= 84.1$ [%] ($z0 = -1.0$), $= 97.7$ [%] ($z0 = -2.0$), $= 99.9$ [%] ($z0 = -3.0$) (♦: $RL = 1$, ■: $RL = 2$, ▲: $RL = 3$)

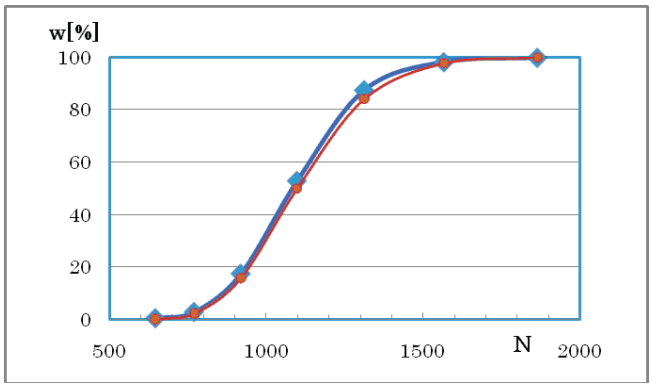


Fig. 4. Comparison of $N(w, RL = 2)$ between theoretical and experimental values at $w = 0.1$ [%] ($z0 = 3.0$), $= 2.3$ [%] ($z0 = 2.0$), ..., $= 99.9$ [%] ($z0 = -3.0$)) (♦: experimental value, ●: theoretical value)

30 [%] at $N_{miss} = 2$ and 20 [%] at $N_{miss} = 3$ respectively, although the size of the dataset used was at the critical level.

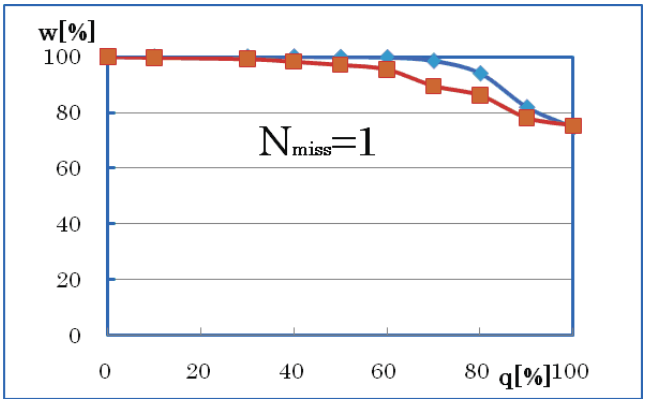
VI. Experimental Studies on Contaminated Values in the Decision Attribute

The study of inducting if-then rules from decision tables containing decision attribute values changed by some cause has seldom been tried to date, even though real-world datasets may contain such values. The model of changing values by $NoiseD$ and observing such datasets is shown in Fig. 1. To examine the performance of STRM for inducting true rules from such a dataset against the critical data size, we conducted a simulation experiment similar to that in sections IV and V, by changing Step1 and adding other procedures:

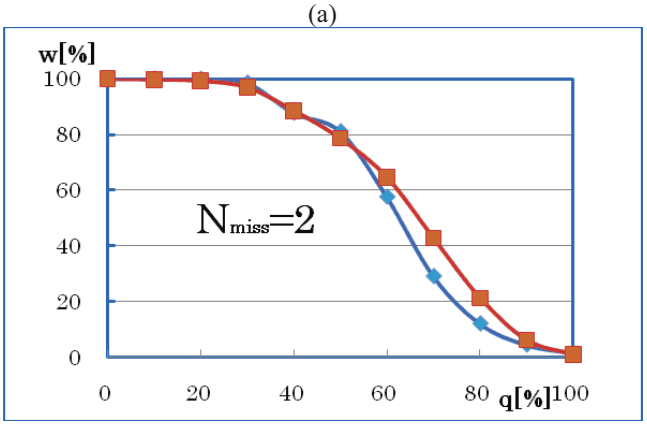
Step1: Randomly select $u(i)$ by $1865 = N_{lst}(w = 100, RL = 2)$ from the decision table ($N = 10000$), change $u^D(i)$ ($i = 1, ..., 1865$) randomly into one

Open Science Index, Computer and Information Engineering Vol:9, No:6, 2015 publications.waset.org/10001804/pdf

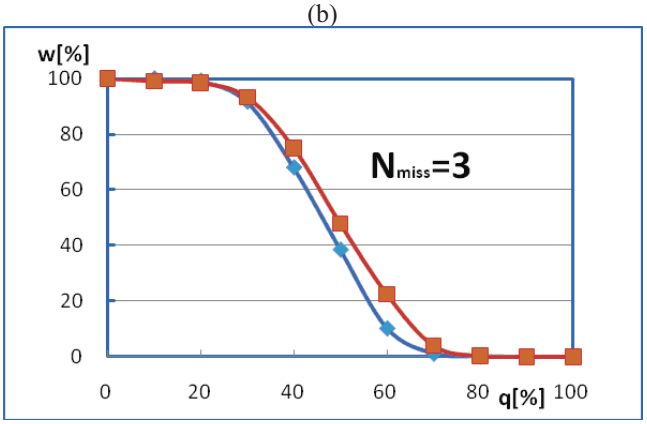
Open Science Index, Computer and Information Engineering Vol:9, No:6, 2015 publications.waset.org/10001804/pdf



(a)



(b)



(c)

Fig. 5. Experimental results with respect to "?": ■, "*": ♦

of $\{1, 2, ..., M_D\}$ with the probability q [%], and build a new decision table.

The experimental results by the rate of true rules inducted by STRIM and $Nr = 100$ with respect to q [%] (Fig. 6) show that STIRM is highly robust against the contaminated noise in the decision attribute, since it can estimate true rules by better than 95 [%] until $q =$ about 60 [%], even though the dataset was at the critical size.

VII. Conclusion

We conducted a simulation experiment to examine the ability and capacity of STRIM for the data size of the

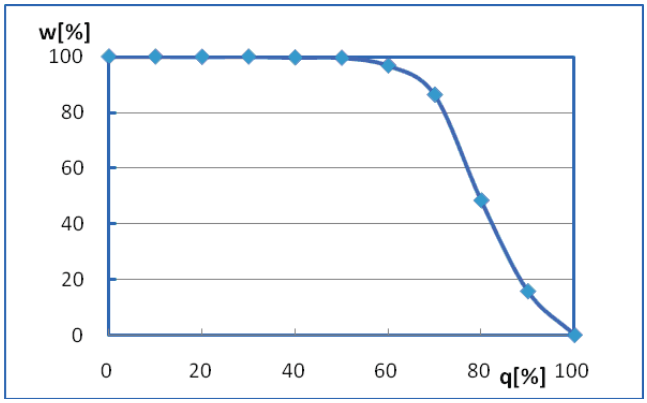


Fig. 6. Experimental results with respect to noise contaminated in decision attribute

decision table, and handling and accepting missing values in the condition attributes and contaminated values in the decision attribute. The results of these experiments were given in the form of the rate of true rules inducted. The results overall show that STRIM is highly robust even where such contaminated values exist. This suggests that STRIM is highly applicable to real-world datasets, and carries advantages for the if-then induction problem.

References

[1] Z. Pawlak: Rough sets, Internat. J. Inform. Comput. Sci., Vol. 11, No. 5, pp. 341-356 (1982).
[2] A. Skowron and C. M. Rauser: The discernibility matrix and functions in information systems, In: R. Słowiński(ed), Intelligent Decision Support, Handbook of Application and Advances of Rough Set Theory, Kluwer Academic Publishers, pp. 331-362 (1992).
[3] Y. G. Bao, X. Y. Du, M. G. Deng and N. Ishii: An efficient method for computing all reducts, Transactions of the Japanese Society for Artificial Intelligence, Vol. 19, No. 3, pp. 166-173 (2004).
[4] J. W. Grzymala-Busse: LERS — A system for learning from examples based on rough sets. In: Intelligent Decision Support. Handbook of Applications and Advances of the Rough Sets Theory, ed. By R. Słowiński, Kluwer Academic Publishers, pp. 3-18 (1992).
[5] W. Ziarko: Variable precision rough set model, Journal of Computer and System Science, Vol. 46, pp. 39-59 (1993).
[6] N. Shan and W. Ziarko: Data-based acquisition and incremental modification of classification rules, Computational Intelligence, Vol. 11, No. 2, pp. 357-370 (1995).
[7] T. Nishimura, Y. Kato and T. Saeki: Studies on an effective algorithm to reduce the decision matrix, RSFDGrC 2011, LNAI Vol. 8743, pp. 240-243, (2011).
[8] T. Matsubayashi, Y. Kato and T. Saeki: A new rule induction method from a decision table using a statistical test, In: T. Li et al. (Eds.): RSKT 2012, LNAI 7414, pp. 81-90, Springer, Heidelberg (2012).
[9] J. W. Grzymala-Busse and W. J. Grzymala-Busse: Handing missing attribute values, O. Maimon, L. Rokach (eds.), Data Mining and Knowledge Discovery Handbook, 2nd ed., Springer, pp. 33-49 (2010).
[10] J. W. Grzymala-Busse: MLEM2; A new algorithm for rule induction from imperfect data, Proceedings of 9th International Conference on Information Processing and Management of Uncertainty in Knowledge — Based Systems, pp. 243-250 (2002).
[11] R. E. Walpole, R. H. Myers, S. L. Myers, K. Ye: Probability and Statistics for Engineers and Scientists, Eighth edition, Pearson Prentice Hall, pp.187-191 (2007).