

## Several Considerations on Statistical Test Rule Induction Method

Yuichi Kato

Faculty of Science & Engineering, Shimane Univ.  
1060 Nishikawatsu-cho, Matsue city, 690-8504, JAPAN  
e-mail: ykato@cis.shimane-u.ac.jp

Tetsuro Saeki

Information, Yamaguchi University  
677-1 Yoshida, Yamaguchi city, Yamaguchi 753-8511, JAPAN  
e-mail: tsaeki@yamaguchi-u.ac.jp

Shoutarou Mizuno

Faculty of Science & Engineering, Shimane Univ.  
1060 Nishikawatsu-cho, Matsue city, 690-8504, JAPAN  
e-mail: s129902@matsu.shimane-u.ac.jp

**Abstract—** STRIM (Statistical Test Rule Induction Method) has been proposed as a method to effectively induct if-then rules from the decision table. The method was studied independently of the conventional rough sets methods. This paper summarizes the basic concept of STRIM, considers the relationship between STRIM and conventional methods, especially VPRS (Variable Precision Rough Set), and shows that STRIM develops the notion of VPRS into a statistical principle. In a simulation experiment, we also consider the condition that STRIM inducts the true rules specified in advance, as such conditions have not yet been studied, even in VPRS. Examination of the conditions is very important if STRIM is properly applied to a set of real-world data set.

**Keywords-** Rough Sets, Rule Induction, Decision Table, Simulation Experiment, If-then Rules

### I. INTRODUCTION

Rough Sets theory as introduced by Pawlak[1] provides a database called the decision table, with various methods of inducting if-then rules and determining 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 to date [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 seems to narrow the scope of the analysis. In a previous study [8] we devised a model of data generation for the database, and proposed a statistical rule induction method and 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]. In contrast, when conventional methods [4,6,7] were used, significant rules could barely be inducted, and different rule sets were inducted from different sample data sets with the same rules; i.e. using these methods results were highly dependent on the sample set.

This paper examines the principle of STRIM in more depth. Specifically, this paper briefly summarizes the procedures of the two stage STRIM process. The first stage finds rule candidates, and the second arranges these candidates. After summarizing them by use of an example simulation experiment, we then consider the relationship between the principle of rule induction in the first stage and that in the conventional method, especially in VPRS (Variable Precision Rough Set)[5]. STRIM was developed independently from the conventional rough set theory, and our results show that STRIM can be considered to develop the notion of VPRS into a statistical principle, by proving that the admissible classification error in VPRS corresponds to the significance level of the STRIM statistical test.

We further examine the validity of the second process and the arrangement by STRIM, based on a statistical model which clearly shows the standard of the arrangement. In contrast, that achieved by VPRS is shown to be not so clear, and the decision by the analyst studying the matter is required. Both considerations in the two processes are also illustrated by the results of the simulation experiment, and seem to be useful for understanding and/or interpreting the results when analyzing real-world data sets.

II. DATA GENERATION MODEL AND DECISION TABLE

Rough Sets theory is used for inducing if-then rules hidden in the decision table S. S is conventionally denoted  $S=(U, A=C \cup D, V, \rho)$ . Here,  $U=\{u(i) \mid i=1, \dots, |U|=N\}$  is a sample set, A is an attribute set,  $C=\{C(j) \mid j=1, \dots, |C|\}$  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  $\rho: U \times A \rightarrow V$ . For example, if  $a=C(j) \in A$  ( $j=1, \dots, |C|$ ) then  $V_a=\{1, 2, \dots, MC(j)\}$  and if  $a=D$  then  $V_a=\{1, 2, \dots, MD\}$ . Table 1 shows the example of  $|C|=6$ ,  $MC(j) = 6$ ,  $MD = 6$ ,  $\rho(x=u(1), a=C(1))=5$ ,  $\rho(x=u(2), a=C(2))=5$ .

STRIM considers the decision table to be a sample data set obtained from an input-output system including a rule box, as shown in Fig.1, and a hypothesis regarding the decision attribute values, as shown in Table 2. 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  $u^D(i)$  by use of rules contained in the rule box and the hypothesis. For example, specify the following rules in the rule box:

$R(d)$ : if  $Rd$  then  $D=d$ , ( $d=1, \dots, MD=6$ ),

where  $Rd = (C(1)=d) \wedge (C(2)=d) \vee (C(3)=d) \wedge (C(4)=d)$ . Generate  $u^C(i)=(v_{C(1)}(i), v_{C(2)}(i), \dots, v_{C(|C|)}(i))$  of  $u(i)$  by use of random numbers with a uniform distribution, and then  $u^D(i)$  is determined by use of the rules specified in the rule box and the hypothesis. Table 1 is an example generated by this specification. The model of the data generation seems not to be so curious, when comparing with that in the real world. Inducting if-then rules from the decision table then identifies the rules in the rule box by use of the set of inputs-output  $(u^C(i), u^D(i)) (i=1, \dots, |U|=N)$ .

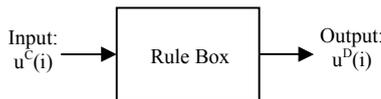


Figure 1. Relationship between the condition attributes' value and the decision attribute's value

TABLE I. AN EXAMPLE OF A DECISION TABLE

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

TABLE II. HYPOTHESIS WITH REGARD TO THE DECISION ATTRIBUTE VALUE

Hypothesis 1	$u^C(i)$ coincides with $R(k)$ , and $u^D(i)$ is uniquely determined as $D=d(k)$ (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 $R(d)(d=d_1, d_2, \dots)$ , 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).

III. SUMMARIES OF RULE INDUCTION PROCEDURES BY STRIM

STRIM inducts if-then rules from the decision table through a 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 2). 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  $U(CP(k)) = \{u(i) \mid u^{C=CP(k)}(i)\}$ . Also derive  $U(m) = \{u(i) \mid u^{D=m}(i)\}$  ( $m=1, \dots, MD$ ). Calculate the distribution  $f: (n_1, n_2, \dots, n_{MD})$  of the decision attributes of  $U(CP(k))$ , where  $n_m = |U(CP(k)) \cap U(m)|$  ( $m=1, \dots, MD$ ). If  $CP(k)$  assumed 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 2, and the distribution  $f$  does not have partiality. On the contrary, 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  $CP(k)$  assumed is a neither necessary nor sufficient condition. Whether  $f$  has partiality or not can be determined objectively by statistically testing the following null hypothesis  $H_0$  and its alternative hypothesis  $H_1$ :  $H_0$ :  $f$  does not have partiality.  $H_1$ :  $f$  has partiality.

Table 3 shows the number of examples of  $CP(k)$ ,  $(n_1, n_2, \dots, n_{MD})$  and an index of the partiality by  $z$  derived from Table 1 with  $N=10000$ , in order to illustrate this concept. For example, the first row means the following: 100000 denotes  $CP(k=1)=(C(1)=1)$  (the rule length is  $RL=1$ ) and its corresponding  $f=(496, 240, 275, 238, 224, 226)$  and  $z=12.52$ , where

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

$$n_d = \max(n_1, n_2, \dots, n_{MD} = n_6), \quad (d \in \{1, 2, \dots, MD=6\}),$$

$$p_d = \text{Pr } o(D = d), \quad n = \sum_{m=1}^{MD} n_m. \text{ In principle, } (n_1, n_2, \dots, n_{MD})$$

obeys a multinomial distribution which is sufficiently approximated by the standard normal distribution by use of  $n_d$  under the condition:  $p_d n \geq 5$  and  $n(1-p_d) \geq 5$ . In the same way, the fifth row 110000 denotes  $CP(k=5)=$

$(C(1)=1 \wedge C(2)=1)$  (RL=2), the 13-th (RL=3) is  $C(1)=1 \wedge C(2)=1 \wedge C(5)=1$ , and so on. Here, if we specify a standard of the significance level such as  $z \geq z_\alpha=3.0$  and reject  $H_0$ , then the  $CP(k)$  assumed becomes a candidate for the rules in the rule box.

The second stage process is that of arranging the set of rule candidates derived from the first process, and finally estimating the rules in the rule box, since some candidates may satisfy the relationship:  $CP(k_i) \supseteq CP(k_j) \supseteq CP(k_l) \dots$ . For example, in the case  $100000 \supset 110000 \supset 110010$  (see Table 3). The basic notion is to represent the  $CP(k)$  of the maximum  $z$ , that is, the maximum partiality. In the above example, STRIM selects the  $CP(k)$  of 110000, which by chance coincides with the rule specified in advance. Fig. 2 shows the STRIM algorithm [8].

Table 4 shows the estimated results for Table 1 with  $N=10000$ . STRIM inducts all of twelve rules specified in advance, and also one extra rule. However, there are clear differences between them in the indexes of accuracy and coverage. Another 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 their problems.

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

trying CP(k)	$(C(1), C(2), \dots, C(6))$	$(n_1, n_2, \dots, n_6)$	$z$
1	(1,0,0,0,0,0)	(469,240,275,238,224,226)	12.52
2	(2,0,0,0,0,0)	(238,454,245,244,232,219)	12.12
3	(3,0,0,0,0,0)	(236,213,477,271,232,222)	13.36
4	(0,0,0,0,0,6)	(289,277,300,255,296,296)	0.97
5	(1,1,0,0,0,0)	(235,10,12,2,7,8)	30.77
6	(1,2,0,0,0,0)	(41,47,41,41,40,49)	1.06
7	(1,3,0,0,0,0)	(46, 52, 57, 60, 51, 40)	1.46
8	(1,0,0,0,0,6)	(84, 36, 52, 38, 34, 39)	5.95
9	(2,1,0,0,0,0)	(46, 35, 42, 51, 37, 37)	1.73
10	(2,2,0,0,0,0)	(8, 227, 6, 11, 4, 6)	30.47
11	(2,3,0,0,0,0)	(49, 43, 44, 55, 49, 44)	1.3
12	(0,0,0,0,6,6)	(52, 50, 60, 46, 52, 43)	1.54
13	(1,1,0,0,1,0)	(38, 1, 3, 1, 0, 1)	12.61
14	(1,1,0,0,2,0)	(38, 3, 1, 0, 3, 3)	11.81
15	(1,1,0,0,5,0)	(49, 3, 2, 1, 1, 1)	14.22
16	(1,1,0,0,6,0)	(46, 0, 3, 0, 0, 1)	14.48
17	(1,1,0,0,0,3)	(45, 4, 3, 0, 3, 1)	12.97
18	(2,2,0,5,0,0)	(1, 45, 1, 3, 1, 0)	13.9
19	(2,2,0,0,1,0)	(0, 55, 0, 2, 0, 1)	16.15
20	(2,2,0,0,2,0)	(0, 38, 1, 1, 2, 2)	12.61
21	(2,2,0,0,4,0)	(4, 37, 1, 2, 0, 1)	12

IV. STUDIES ON THE RELATIONSHIP BETWEEN THE CONVENTIONAL METHOD AND STRIM

The most basic strategy to induct the rules from a decision table is to use the inclusion relationship between the set derived by the condition attributes and the set by the decision attribute; many methods of achieving this have

been proposed to date. Fig. 3, for example, shows the well-known LEM2 algorithm [4] and B at LN (Line No.)=0 is specified like  $B=U(d)=\{u(i)|u^{D=d}(i)\}$  removing the conflicted data set. LEM2 with lower approximation derives  $CP(k)$ , satisfying  $U(d) \supseteq U(CP(k))$ . In the figure,  $t$  corresponds to  $C(j_k)=v_{C(j_k)}$ ,  $[t]$  to  $U(t)=\{u(i)|u^{C=t}(i)\}$ ,  $T$  to  $CP(k)$ , and the final result  $\tau$  to  $v_k CP(k)$  is obtained by repeating from LN=3 to LN=16 until the condition  $U(d)=U(v_k CP(k))$  is satisfied.

However, as previously shown [8], LEM2 is likely to induct many sub-rules of their true rules with longer rule length, since it executes the algorithm until the condition  $U(d)=U(v_k CP(k))$  is satisfied. In 1993 Zialko [5] introduced the variable precision rough set, which inducts  $CP(k)$  satisfying the following conditions:

$$\underline{C}_\varepsilon(U(d)) = \{u(i) | acc \geq 1 - \varepsilon, acc = |U(d) \cap U(CP(k))| / |U(CP(k))| = n_d / n\} \quad (2)$$

where  $\varepsilon \in [0, 1]$  is an admissible classification error, and  $acc$  is accuracy of the rule.  $\underline{C}_\varepsilon(U(d))$ , a  $\varepsilon$ -lower approximation of VPRS, coincides with the ordinary lower approximation by  $\varepsilon=0$ , and adopts the rules with the high index of coverage defined by  $cov = |U(d) \cap U(CP(k))| / |U(d)|$ ; this can squeeze the above sub-rules. VPRS has been widely used for solving real-world problems, and a variety of modified VPRSs have been proposed [9-11]. However, the standard of adopting rules is not so clear. For example,  $(acc, cov)$  of  $CP(k=1)$ ,  $CP(k=5)$  and  $CP(13)$  in Table 3 are (0.281, 0.285), (0.857, 0.143) and (0.864, 0.0231) respectively.  $CP(k=13)$  should be adopted as the most accurate,  $CP(k=1)$  as the widest coverage, and  $CP(k=5)$  as the moderate index of both; however, this requires a decision by the analyst studying the matter. Moreover, no systematical algorithms of VPRS (such as LEM2) have been reported, since  $\varepsilon > 0$  makes it difficult to derive such an algorithm.

We now consider a  $CP(k)$  satisfying (2). The greater part of the decision attribute value of the  $U(CP(k))$  is now included in  $U(d)$ , since the  $CP(k)$  satisfies (2). Accordingly, the distribution of the decision attribute value of the  $U(CP(k))$  has partiality in  $D=d$ , which coincides with the basic concept of STRIM. As shown in 3, STRIM statistically tests whether  $(n_1, n_2, \dots, n_{M_d})$  is partial or not, and gives the decision with a significance level  $z_\alpha$ .  $n_d$  in (2) satisfies the event  $n_d \geq n - n \cdot \varepsilon$  and the probability of the event is evaluated thus:

$$Pr o(n_d \geq n - n \cdot \varepsilon) = Pr o\left(\frac{n_d + 0.5 - np_d}{(np_d(1 - p_d))^{0.5}} \geq \frac{n + 0.5 - n \cdot \varepsilon - np_d}{(np_d(1 - p_d))^{0.5}}\right) \quad (3)$$

Accordingly,  $z_\alpha = \frac{n + 0.5 - n \cdot \varepsilon - np_d}{(np_d(1 - p_d))^{0.5}}$  is obtained comparing (3) with (1) and the following relationship between  $\varepsilon$  and  $z_\alpha$  holds:

$$\varepsilon = 1 - p_d + 0.5/n - z_\alpha(p_d(1 - p_d)/n)^{0.5} \quad (4)$$

The covering index in VPRS also corresponds to the conditions  $p_d n \geq 5$  and  $n(1-p_d) \geq 5$  which are needed for testing in STRIM as sufficient evidence. In R(i=13) in Table 4, substituting  $n=254$ ,  $p_d=1/6$  and  $z_a=3.0$  for (4),  $\varepsilon \doteq 1-1/6$  and  $acc \geq 1-\varepsilon = 1/6=0.167$ , which satisfies  $acc=0.240$  of R (i=13).

```

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(1)(l=1,2,...) so that every r(k) belongs to one Pyramid
at least; // Stage 2
make rePyramid(1)(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 proposed rule; // count n1 n2
if (frequency >= NO) { //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
    
```

Figure 2. An algorithm for STRIM

V. STUDIES ON ARRANGEMENT OF RULE CANDIDATES

There may be rule candidates satisfying relationships  $CP(k_i) \supset CP(k_j) \supset CP(k_l) \dots$  after the first stage process. STRIM selects the candidate with the maximum partiality in the relationship. Let us consider whether STRIM is assured of selecting the true rule in the rule box. Hereafter, CP(F) denotes the true rule in the rule box, and we assume that it satisfies the relationship  $CP(E) \supset CP(F) \supset CP(G)$  and has the maximum  $n_d$  at  $d=1$  like  $CP(E)=\text{"100000"}$ ,  $CP(F)=\text{"110000"}$  and  $CP(G)=\text{"110010"}$  in Table 3 and the distribution  $f: (n_1^F, n_2^F, \dots, n_{M_d}^F)$ . Then:

$z^F = \frac{n_1^F + 0.5 - n^F p_1}{(n^F p_1 (1 - p_1))^{0.5}} \approx \frac{n^F (a^F - p_1)}{\sigma^F}$ . Here,  $0.5 < n^F p_1$ ,  $n_1^F = a^F n^F$  ( $0 < a^F \leq 1$ ),  $n^F = \sum n_i^F$  and  $\sigma^F = (n^F p_1 (1 - p_1))^{0.5}$ . In the same way with regard to G,  $z^G = \frac{n_1^G + 0.5 - n^G p_1}{(n^G p_1 (1 - p_1))^{0.5}} \approx \frac{n^G (a^G - p_1)}{\sigma^G}$ . Here  $U(F) \supset U(G)$ ,  $a^G \doteq a^F$  and  $n^F > n^G$  lead to  $n^G = r n^F$  ( $0 < r < 1$ ). Accordingly,  $z^G \approx \frac{n^G (a^G - p_1)}{\sigma^G} = \frac{r^{0.5} n^F (a^F - p_1)}{\sigma^F} = r^{0.5} z^F$  which means  $z^G < z^F$  and STRIM selects not CP(G) but CP(F). In

the same way with regard to E,  $z^E = \frac{n_1^E + 0.5 - n^E p_1}{(n^E p_1 (1 - p_1))^{0.5}}$ . Here,  $n_1^E = n_1^F + n_1^{EF} = a^E n^E + a^{EF} n^{EF}$ ,  $n^{EF} = |U(EF)|$  and  $U(EF) = U(E) - U(F)$ . U(EF) is an indifferent data set (See Hypothesis 2). Taking their relationships into consideration, the following equation holds:

$$z^E \approx \frac{n^F (a^F - p_1)}{\sigma^E} + \frac{n^{EF} (a^{EF} - p_1)}{\sigma^E} = \frac{\sigma^F}{\sigma^E} z^F + \frac{\sigma^{EF}}{\sigma^E} z^{EF} = s_E^F z^F + s_E^{EF} z^{EF}$$

Here  $n^F < n^E$  and  $n^{EF} < n^E$  lead to  $\sigma^F / \sigma^E = s_E^F < 1$  and  $\sigma^{EF} / \sigma^E = s_E^{EF} < 1$ . Fig. 4 shows the relationship between  $z^F$  and  $z^E$  by use of the following two lines:  $z^E = s_E^F z^F + s_E^{EF} z^{EF}$  (①),  $z^E = z^F$  (②). The cross point of the two line is  $(z_0^F = s_E^{EF} / (1 - s_E^{EF}), z_0^F)$ . Accordingly, STRIM always selects CP(F) if the inequality  $z^F \geq z_0^F$  holds. As conclusion in this section, STRIM necessarily selects CP(F) of the true rule only if  $z^F > z_0^F$  holds. In Table1  $s_E^F = (1.0/M_C)^{0.5}$ ,  $s_E^{EF} = (1 - 1/M_C)^{0.5}$  and  $M_C=6$ , and  $|z_E^F| < z_a = 3.0$  holds with less than 1[%] error. Accordingly, if  $z^F \geq z_0 = s_E^{EF} / (1 - s_E^F) \doteq 5$  holds, then STRIM selects not CP(E) but CP(F). We can confirm the validity of the consideration of this section in Table 3 and Table 4. Especially, note that R (i=13) in Table4 does not satisfy the condition  $z \geq 5.0$ , and doubt of the inducted result thus arises.

```

Line No. Procedure LEM2
0 (input: a set B
output: a single local covering tof set B);
1 begin
2 G:=B;
3  $\tau := \varphi$ ;
4 while G $\neq \varphi$ 
5 begin
6 T:= $\varphi$ ;
7 T(G):={ t | [t]  $\cap$  G $\neq \varphi$  };
8 while T= $\varphi$  or [T]  $\not\subseteq$  B
9 begin
10 select a pair t $\in$ T(G) such that |[t]  $\cap$  G| is maximum;
11 if a tie occurs, select a pair t $\in$ T(G)
12 with the smallest cardinality of [t];
13 if another tie occurs, select first pair;
14 T:=T  $\cup$  {t};
15 G:=[t]  $\cap$  G;
16 T(G):={t | [t]  $\cap$  G $\neq \varphi$  };
17 T(G):=T(G)-T;
18 end {while}
19 for each t $\in$ T do
20 if [T-{t}]  $\subseteq$  B then T:=T-{t};
21  $\tau := \tau \cup \{T\}$ ;
22 G:=B -  $\cup$  T  $\in$   $\tau$ [T];
    
```

```

16 end {while};
17 for each T ∈ τdo
18 if  $\cup S \in \tau - \{T\} | S = B$  then  $\tau := \tau - \{T\}$ ;
end {procedure}.

```

Figure 3. An algorithm for LEM2

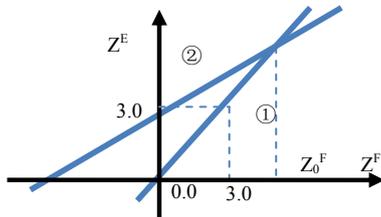


Figure 4. An Relationship between  $z^F$  and  $z^E$

VI. CONCLUSION

This paper showed STRIM [8] to be a powerful tool for inducing if-then rules from a decision table by examining and illustrating the following features and relationships between results from the STRIM model and that from conventional methods, especially VPRS [5] in a simulation experiment:

1) VPRS uses the indexes of accuracy and coverage with an admissible error  $\epsilon$  when it selects the candidates. The accuracy can be recognized as the index of the partiality of the distribution of the decision attribute values for the trying rule, which coincides with the idea of STRIM. The  $\epsilon$  corresponds to the significance level by  $z_\alpha$  in STRIM. The coverage corresponds to the applicable condition for a statistical test by STRIM. However, VPRS does not have the objective standard of both indexes to select rule candidates, since to data the conventional methods do not view the decision table as a sample set obtained from its population.

2) STRIM provides statistical assurance for an analyst searching for the true rules under the proper conditions studied in 5, as the results can statistically show whether those rules inducted are true or not. In contrast, VPRS

provides no such assurance, since it is not a method based on a data generating model, as is STRIM.

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. Rausser: 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, 331-362(1992).
- [3] Y.G. Bao, X.Y. Du, M.G. Deng and N. Ishii: An efficient method for computing all reducts, Transaction 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, 3-18(1992).
- [5] W. Ziarko: Variable precision rough set model, Journal of Computer and System Science, 46, 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, 2012, Springer, Heidelberg (2012)..
- [9] W. Jaworski: Rule induction: Combining rough sets and statistical approaches, In C.-C.Chan et al.(eds.): RSCTC 2008, LNAI vol.5306, pp.170-180,2008, Springer, Heidelberg (2008).
- [10] G. Xiw, J. Zhang, K.K. Lai and L. Yu: Variable precision rough set group decision-making: An application, International Journal of Approximate Reasoning, 49,pp.331-343(2008).
- [11] M. Inuiguchi, Y. Yoshioka, Y. Kusunoki: Variable-precision dominance-based rough set approach and attribute reduction, International Journal of Approximate Reasoning, 50, pp.1199-1214(2009).
- [12] K.Y. Huang, Ting-Hua. Chang and Ting-Cheng Chang: Determination of the threshold  $\beta$  of variable precision rough set by fuzzy algorithms, International Journal of Approximate Reasoning, 52, pp.1056-1072(2011)..

TABLE IV. RESULTS OF ESTIMATED RULES FOR THE DECISION TABLE IN TABLE 1 BY STRIM

estimated rule R(i)	(C(1),C(2),...,C(6),D)	(n <sub>1</sub> ,n <sub>2</sub> ,...,n <sub>6</sub> )	p-value(z)	accuracy	coverage
1	(5,5,0,0,0,5)	(7,8,5,7,271,4)	0(34.15)	0.897	0.162
2	(0,0,1,1,0,0,1)	(243,6,5,6,4,3)	0(32.68)	0.910	0.148
3	(4,4,0,0,0,4)	(10,2,8,252,7,6)	0(32.58)	0.884	0.15
4	(0,0,5,5,0,0,5)	(5,5,6,11,249,7)	0(32.27)	0.880	0.149
5	(6,6,0,0,0,0,6)	(10,12,4,7,253)	0(32.16)	0.866	0.154
6	(3,3,0,0,0,0,3)	(6,3,254,13,8,12)	0(32.00)	0.858	0.15
7	(0,0,2,2,0,0,2)	(4, 243, 2, 8, 5, 14)	0(31.90)	0.865	0.143
8	(0,0,3,3,0,0,3)	(11, 8, 243, 5, 7, 7)	0(31.48)	0.865	0.143
9	(0,0,6,6,0,0,6)	(7, 2, 8, 10, 9, 240)	0(31.41)	0.870	0.146
10	(0,0,4,4,0,0,4)	(8, 12, 13, 245, 7, 7)	0(30.91)	0.839	0.146
11	(1,1,0,0,0,0,1)	(235, 10, 12, 2, 7, 8)	0(30.77)	0.858	0.143

12	(2,2,0,0,0,2)	(8, 227, 6, 11, 4, 6)	0(30.47)	0.866	0.136
13	(0,0,0,1,1,2)	(39, 61, 44, 44, 35, 31)	6.26e-4(3.23)	0.240	0.037