Proposal for a New Reduct Method for Decision Tables and an Improved STRIM

Jiwei Fei[1,2], Tetsuro Saeki[1,2(✉)], and Yuichi Kato[1,2]

[1] Yamaguchi University, 2-16-1 Tokiwadai, Ube, Yamaguchi 755-8611, Japan
tsaeki@yamaguchi-u.ac.jp

[2] Shimane University, 1060 Nishikawatsu-cho, Matsue, Shimane 690-8504, Japan
ykato@cis.shimane-u.ac.jp

Abstract. Rough Sets theory is widely used as a method for estimating and/or inducing the knowledge structure of if-then rules from a decision table after a reduct of the table. The concept of a reduct is that of constructing a decision table by necessary and sufficient condition attributes to induce the rules. This paper retests the reduct by the conventional methods by the use of simulation datasets after summarizing the reduct briefly and points out several problems of their methods. Then, a new reduct method based on a statistical viewpoint is proposed and confirmed to be valid by applying it to the simulation datasets. The new reduct method is incorporated into STRIM (Statistical Test Rule Induction Method), and plays an effective role for the rule induction. The STRIM including the reduct method is also applied for a UCI dataset and shows to be very useful and effective for estimating if-then rules hidden behind the decision table of interest.

1 Introduction

Rough Sets theory was introduced by Pawlak [1] and used for inducing if-then rules from a dataset called the decision table. The induced if-then rules simply and clearly express the structure of rating and/or knowledge hiding behind the decision table. Such rule induction methods are needed for disease diagnosis systems, discrimination problems, decision problems and other aspects. The first step for the rule induction is to find the condition attributes which do not have any relationships with the decision attribute, to remove them and finally to reduce the table. Those processes to obtain the reduced table are useful for efficiently inducing rules and called a reduct. The conventional Rough Sets theory to induce if-then rules is based on the indiscernibility of the samples of the table. The reduct by the conventional method also uses the same concept and various types of indiscernibility, methods to find their indiscernibility and algorithms for the reducts are proposed to date [2–7].

This paper retests the conventional reduct methods through the use of a simulation dataset and points out their problems after summarizing the conventional rough sets and reduct methods. Then a new reduct method is proposed to overcome their problems from a statistical point of view. Specifically, the

Y. Tan et al. (Eds.): DMBD 2017, LNCS 10387, pp. 366–378, 2017.
DOI: 10.1007/978-3-319-61845-6_37

Table 1. An example of a decision table.

U	$(C(1)C(2)C(3)C(4)C(5)C(6))$	D
1	563242	3
2	256124	6
3	116226	1
4	416646	6
...	......	...
$N-1$	151252	2
N	513135	4

new method recognizes each sample data in the decision table as the outcomes of random variables of the tuple of the condition attributes and the decision attribute, since the dataset is obtained from their population of interest. Accordingly, the reduct problem can be replaced by the problem of finding the condition attributes which are statistically independent of the decision attribute and/or its values. The statistical independence can be easily tested, for example, by a Chi-square test using the dataset. The validity of the new reduct method is confirmed by applying it to the simulation dataset. The experiment also gives an idea of improving STRIM (Statistical Test Rule Induction Method [8–11]) to include the reduct function and to induce if-then rules more efficiently. The usefulness of the reduct method and the improved STIRM are also confirmed by applying them to a UCI dataset [12] prepared for machine learning.

2 Conventional Rough Sets and Reduct Method

Rough Sets theory is used for inducing if-then rules from a 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 $V = \cup_{a \in A} V_a$ and is characterized by an information function ρ: $U \times A \rightarrow V$. Table 1 shows an example where $|C| = 6$, $|V_{a=C(j)}| = M_{C(j)} = 6$, $|V_{a=D}| = M_D = 6$, $\rho(x = u(1), a = C(1)) = 5$, $\rho(x = u(2), a = C(2)) = 5$, and so on.

Rough Sets theory focuses on the following equivalence relation and equivalence set of indiscernibility:

$$I_C = \{(u(i), u(j)) \in U^2 | \rho(u(i), a) = \rho(u(j), a), \forall a \in C\}.$$

I_C derives the quotient set $U/I_C = \{[u_i]_C | i = 1, 2, ...\}$. Here, $[u_i]_C = \{u(j) \in U | (u(j), u_i) \in I_C, u_i \in U\}$. $[u_i]_C$ is an equivalence set with the representative element u_i and is called an element set of C in Rough Sets theory [2]. Let be

Line No.	Algorithm to compute a single global covering
1	(input: the set A of all attributes, partition $\{d\}^*$ on U; output: a single global covering R);
2	**Begin**
3	compute partition A^*;
4	$P := A$;
5	$R := \emptyset$;
6	**if** $A^* \leq \{d\}^*$
7	**Then**
8	**Begin**
9	**for** each attribute a in A **do**
10	**Begin**
11	$Q := P - \{a\}$;
12	compute partition Q^*;
13	**if** $Q^* \leq \{d\}^*$ **then** $P := Q$
14	**end** {for}
15	$R := P$
16	**end** {then}
17	**end** {algorithm}

Fig. 1. An example of LEM1 algorithm.

$\forall X \subseteq U$ then X can be approximated like $C_*(X) \subseteq X \subseteq C^*(X)$ by use of the element set. Here,

$$C_*(X) = \{u_i \in U | [u_i]_C \subseteq X\}, \tag{1}$$

$$C^*(X) = \{u_i \in U | [u_i]_C \cap X \neq \emptyset\}, \tag{2}$$

$C_*(X)$ and $C^*(X)$ are called the lower and upper approximations of X by C respectively. The pair of $(C_*(X), C^*(X))$ is usually called a rough set of X by C. Specifically, let $X = D_d = \{u(i) | (\rho(u(i), D) = d\}$ called concept $D = d$ then $C_*(X)$ is surely a set satisfying $D = d$ since $C_*(X) \subseteq X$ and it derives if-then rules of $D = d$ with necessity.

The conventional Rough Sets theory seeks a minimal subset of C denoted with $B(\subseteq C)$ satisfying the following two conditions:

(i) $B_*(D_d) = C_*(D_d)$, $d = 1, 2, ..., M_D$.
(ii) $a(\in B)$ satisfying $(B - \{a\})_*(D_d) = C_*(D_d)$ $(d = 1, 2, ..., M_D)$ does not exist.

$B(\subseteq C)$ is called a relative reduct of $\{D_d | d = 1, ..., M_D\}$ preserving the lower approximation and is useful for finding if-then rules since redundant condition attributes have been already removed from C.

LEM1 algorithm [2] and the discernibility matrix method (DMM) [3] are well known as representative ways to perform reducts. Figure 1 shows an example of LEM1, and A and $\{d\}^*$ at Line 1 of the figure respectively correspond to C and $\{D_d | d = 1, ..., M_D\}$ in this paper. LEM1 from Line 6 to 16 in the figure in principle checks and executes (i) and (ii) for all the combinations of the condition attributes.

DMM [3] at first forms a symmetric $N \times N$ matrix having the following (i, j) element δ_{ij}:

$\delta_{ij} = \{a \in C | \rho(u(i), a) \neq \rho(u(j), a)\}$; $\exists d \in D$, $\rho(u(i), d) \neq \rho(u(j), d)$ and $\{u(i), u(j)\} \cap Pos(D) \neq \emptyset$, $= *$; otherwise.

Here, $Pos(D) = \cup_{d=1}^{M_D} C_*(D_d)$ and $*$ denotes "don't care". Then, a relative reduct preserving the lower approximation can be obtained by the following expression:

$$F^{reduct} = \bigwedge_{i,j:i<j} \bigvee \delta_{ij}. \quad (3)$$

3 Retests of the Conventional Reduct Method

Here we retest the ability of the reducts obtained through LEM1 and DMM by use of a simulation dataset. Figure 2 [8–11] shows a way of how to generate simulation datasets. Specifically let (a) generate the condition attribute values of $u(i)$, that is, $u^C(i) = (v_{C(1)}(i), v_{C(2)}(i), ..., v_{C(|C|)}(i))$ by the use of random numbers with a uniform distribution and (b) determine the decision attribute value of $u(i)$ without NoiseC and NoiseD for a plain experiment, that is $u^D(i)$ by use of if-then rules specified in advance and the hypotheses shown in Table 2 and repeat the (a) and (b) processes by N times. Table 1 shows an example dataset generated by the use of those procedures with the following if-then rule $R(d)$ specified in advance:

$$R(d): \text{ if } Rd \text{ then } D = d \quad (d = 1, ..., M_D = 6), \quad (4)$$

where $Rd = (C(1) = d) \wedge (C(2) = d) \vee (C(3) = d) \wedge (C(4) = d)$.

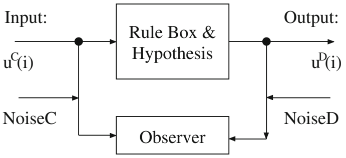

Fig. 2. A data generation model for a decision table contaminated with noise.

Table 2. Hypotheses 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 $R(d)$ $(d = d1, d2, ...)$ 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)

The results of retesting both methods using the $N = 10,000$ dataset showed $F_{LEM1}^{reduct} = F_{DMM}^{reduct} = C(1) \wedge C(2) \wedge C(3) \wedge C(4) \wedge C(5) \wedge C(6)$ while the results were expected to be $F^{reduct} = C(1) \wedge C(2) \wedge C(3) \wedge C(4)$ from the rules (4) specified in advance. The retest experiment was repeated three times by changing the generated dataset and obtained the same results.

These results are clearly derived from the indiscernibility and/or discernibility caused by the element set which could not distinguish the differences between samples by the if-then rules (see Hypothesis 1 in Table 2) or those obtained by chance (see Hypothesis 2 and 3 in Table 2).

4 Proposal of Statistical Reduct Method

As mentioned in Sect. 3, the conventional reduct methods are unable to reproduce the forms of reducts specified in advance from the decision table due to a lack of abilities adaptive to the indifferent and conflicted samples in datasets despite the fact that real-world datasets will have such samples. This paper studies this problem with reducts from the view of STRIM (Statistical Test Rule Induction Method) [8–11]. STRIM regards the decision table as a sample set obtained from the population of interest based on the input-output system as shown in Fig. 2. According to a statistical model, $u(i) = (u^C(i), u^D(i) = (v_{C(1)}(i), v_{C(2)}(i), ..., v_{C(|C|)}(i), u^D(i))$ is an outcome of the random variables of $A = (C, D) = (C(1), C(2), ..., C(|C|), D)$ (hereafter, the names of the attributes are used as the random variables). Then, the following probability model will be specified: For any j, $P(C(j) = v_{C(j)}(k)) = p(j,k)$, $\sum_{k=1}^{M_{C(j)}} p(j,k) = 1$. For any $j1 \neq j2$, $C(j1)$ and $C(j2)$ are independent of each other for simplicity. According to the rules specified in (4), if $C = (1, 1, 2, 3, 4, 5)$ (hereafter (112345) briefly), for example, then $P(D = 1|C = (112345)) = 1.0$ by use of Hypothesis 1 in Table 2. If $C = (123456)$ then $P(D = 1|C = (123456)) = 1/M_D = 1/6$ by use of Hypothesis 2. If $C = (112256)$ then $P(D = 1|C = (112256)) = 1/2$ by use of Hypothesis 3. Generally, the outcome of random variable D is determined by the outcome of C, if-then rules (generally unknown) and the hypothesis shown in Table 2. Consequently, the following expression is obtained:

$$P(D = l, C = u^C(i)) = P(D = l|C = u^C(i))P(C = u^C(i)). \tag{5}$$

Here, $P(D = l|C = u^C(i)$ is the conditional probability of $D = l$ by $C = u^C(i)$ and very dependent on the if-then rules to be induced.

In the special case, if $C(j)$ does not exist in the condition part of the if-then rules of $D = l$, then the event $D = l$ is independent of $C(j)$, that is $P(D = l, C(j)) = P(D = l|C(j))P(C(j)) = P(D = l)P(C(j))$. This independence between $D = l$ and $C(j)$ can be used for a reduct of the decision table for the concept $D = l$. The problem of whether they are independent or not can be easily dealt with using a statistical test of hypotheses by the use of $\{u(i) = (v_{C(1)}(i), v_{C(2)}(i), ..., v_{C(|C|)}(i), u^D(i))|i = 1, ..., N\}$. Specifically, specifications and testing of the following null hypothesis $H0(j,l)$ and its alternative hypothesis $H1(j,l)$ $(j = 1, ..., |C|, l = 1, ...M_D)$ were implemented:

Table 3. Example of contingency table by a statistical reduct ($N = 3000$, $df = 5$).

$D = 1$ $\lvert U(D = 1)\rvert = 503$	C					
	1	2	3	4	5	6
1	**158**	**150**	**135**	**118**	68	89
2	69	76	88	74	79	76
3	63	63	57	81	94	96
4	69	77	74	71	93	71
5	76	67	72	78	84	84
6	68	70	77	81	85	87
χ^2	74.00	78.92	45.84	28.22	4.00	4.83
p-values	1.51E−14	1.41E−15	9.34E−9	3.30E−05	0.550	0.437

$H0(j, l)$: $C(j)$ and $D = l$ are independent of each other.
$H1(j, l)$: $C(j)$ and $D = l$ are not independent of each other.

This paper adopts a Chi-square test since it is a standard method for testing the independence of two categorical variables by use of the contingency table $M_{C(j)} \times 1$. The test statistic χ^2 of $C(j)$ vs. $D = l$ is

$$\chi^2 = \sum_{k=1}^{M_{C(j)}} \frac{(f_{kl} - e_{kl})^2}{e_{kl}}, \tag{6}$$

where, $f_{kl} = |U(C(j) = k) \cap U(D = l)|$, $U(C(j) = k) = \{u(i)|\rho(u(i), C(j)) = k\}$, $U(D = l) = \{u(i)|u^D(i) = l\}$, $e_{kl} = n\hat{p}(j, k)\hat{p}(D, l)$, $n = \sum_{k=1}^{M_{C(j)}} \sum_{l=1}^{M_D} f_{kl}$, $\hat{p}(j, k) = \frac{f_{k\_}}{n}$, $\hat{p}(D, l) = \frac{f_{\_l}}{n}$, $f_{k\_} = \sum_{l=1}^{M_D} f_{kl}$, $f_{\_l} = \sum_{k=1}^{M_{C(j)}} f_{kl}$. χ^2 obeys a Chi-square distribution with degrees of freedom $df = (M_{C(j)} - 1)$ under $H0(j, l)$ and testing condition [13]: $n\hat{p}(j, k)\hat{p}(D, l) \geq 5$. This paper proposes a reduct method to adopt only the $C(j)$s of $H0(j, l)$ that were rejected and to construct a decision table for $D = l$ composed by them, since the test of the hypotheses cannot control type II errors, but only type I errors by a significance level. This paper names the proposed method the statistical reduct method (SRM) to distinguish it from the conventional methods.

A simulation experiment was conducted to confirm the validity of the proposed method using the decision table of the samples of $N = 10,000$ used in Sect. 2, and the following procedures:

Step 1: Randomly select samples by $N_B = 3000$ from the decision table ($N = 10,000$), and form a new decision table.
Step 2: Apply SRM to the new table, and calculate χ^2 every $C(j)$ by $D = l$.

Table 3 shows an example of the contingency table for the case of $D = 1$ vs. $C(j)$ ($j = 1, ..., 6$) and the results of a Chi-square test of them with $df = (M_{C(j)} - 1)$, and suggests the following knowledge:

(1) The p-values of $C(5)$ and $C(6)$ are quite high compared with the other condition attributes and indicate that $C(5)$ and $C(6)$ are independent of $D = 1$, that is, they are redundant and should be removed from the viewpoint of reduct.
(2) The frequencies $f_{kl=1}$ of $C(1) = 1$, $C(2) = 1$, $C(3) = 1$ and $C(4) = 1$ are relatively high compared with those of the rest of the same $C(j)$ $(j = 1, ..., 4)$. Accordingly, the combinations of $C(j) = 1$ $(j = 1, ..., 4)$ will most likely construct the rules of $D = 1$, which coincides with the rules specified in (4).

The above knowledge of (1) and (2) was also confirmed for the case of $D = l$ $(l = 2, ..., 6)$ and coincided with the specifications of Rules (4), and thus through them the validity and usefulness of SRM have been confirmed.

5 Proposal of Improved STRIM

STRIM has been proposed as a method to induce if-then rules from decision tables by use of two stages [8–11]. The first stage is that of searching rule candidates by the following procedures:

Step 1: Specify a proper condition part of trying if-then rules:

$$CP(k) = \bigwedge_j (C(j_k) = v_k). \tag{7}$$

Step 2: Test the condition part on the null hypothesis $(H0)$ that $CP(k)$ is not a rule candidate and its alternative hypothesis $(H1)$ specifying a proper significance level. Specifically, use a test statistic $z = \frac{(n_d + 0.5 - np_d)}{\sqrt{np_d(1-p_d)}}$ which obeys the normal distribution $N(0, 1^2)$ on $H0$, if $np_d \geq 5$, $n(1 - p_d) \geq 5$ (testing condition [14]). Here, $p_d = \frac{1}{|M_D|}$, $n_d = \max(n_1, n_2, ..., n_{M_D})$, $n = \sum_{m=1}^{M_D} n_m$, $n_m = |U(CP(k)) \cap U(D = m)|$, $U(CP(k)) = \{u(i)|u^C(i)$ satisfies $CP(k)\}$, $U(D = m) = \{u(i)|\ u^{D=m}(i)\}$.
Step 3: If $H0$ is rejected then add the trying rule to the rule candidates.
Step 4: Repeat from Step 1 to Step 3 changing the trying rule systematically until the patterns of it are exhausted.

The basic notion of STRIM is that the rule makes a bias in the distribution of decision attribute values $(n_1, n_2, ..., n_{M_D})$. It should be noted that $P(D|CP(k)) = P($ if $CP(k)$ then $D)$ corresponding to (5), and can be estimated by $(n_1, n_2, ..., n_{M_D})/n$ using the sample set.

The second stage is that of arranging the rule candidates having an inclusion relationship by representing them with $CP(k)$ of the maximum bias.

However, the conventional STRIM [8–11] did not have such a reduct function studied in Sect. 4 so that it had to search $CP(k)$s in (7) including even $C(j)$ to be reducted and induced many kinds of rule candidates to burden the second stage. The knowledge from (1) and (2) studied in Sect. 4 can drastically squeeze the

```
Line No. Algorithm to induce if-then rules by STRIM with a reduct function
1   int main(void) {
2   int rdct_max[|CV|] = {0, ... ,0}; //initialize maximum value of C(j)
3   int rdct[|CV|] = {0, ..., 0}; //initialize reduct results by D = l
4   int rule[|C|] = {0, ..., 0}; //initialize trying rules
5   int tail = -1; //initial vale set
6   input data; // set decision table
7   for (di = 1; di<= |D|; di++) { // induce rule candidates every D = l
8   attribute_reduct(rdct_max)
9   set rdct[ck] ; // if (rdct_max[ck]==0) {rdct[ck] = 0;} else {rdct[ck] = 1;}
10  rule_check(rcdct, redct_max, tail, rule); // the first stage process
11  }// end of di
12  arrange rule candidates // the second stage
13  }// end of main
14  int attribute_reduct(int rdct_max[]) {
15  make contingency table for D = l vs. C(j)
16  Test H0(j,l);
17  if H0(j,l) is rejexted then set rdct_max[j,l] = jmax else rdct_max[j,l] = 0; // jmax:the attribute vale of the maximum frequency
18  }// end of attribute_reduct
19  int rule_check(int rdct[], int rdct_max[], int tail,int rule[]) { // the first stage process
20  for (ci = tail+1; cj<|C|; ci++) {
21  for (cj = 1; cj <= rdct[ci]; cj++) {
22  rule[ci] = rdct_max[cj]; // a trying rule sets for test
23  count frequency of the trying rule; // count n1, n2, ...
24  if (frequency>= N0) { //sufficient frequency ?
25  if (|z|>3.0) { //sufficient evidence ?
26  add the trying rule as a rule candidate
27  }// end of if |z|
28  rule_check(ci,rule)
29  }// end of if frequency
30  }// end of for cj
31  rule[ci] = 0; // trying rules reset
32  }// end of for ci
33  }// end of rule_check
```

Fig. 3. An algorithm for STRIM including a reduct function.

search space without idle $C(j)$s and/or its values. Figure 3 shows an algorithm for the improved STRIM described in a C language style including the reduct function "attribute_reduct()" (Line 14–18) studied in Sect. 4. The first stage (Line 7–11) is executed every $D = l$ in a function "rule_check()" (Line 19–33) after operating the reduct (Line 8). The algorithm develops the patterns of trying rules implemented by the dimension "rule[]" (Line 4), for example, for $D = 1$ as (100000) → (110000) → (111000) → (111100) → (110100) → (101100) → (10010) → (010000) ... → (001100) → (000100) by the operation of "rule[ci] = rdct_max[cj]" (Line 22) and the recursive call of "rule_check(ci,rule)" (Line 28) since "rdct_max[] = $[1, 1, 1, 1, 0, 0]$" and "rdct[] = $[1, 1, 1, 1, 0, 0]$" at Line 9 have been obtained (see Table 3). Accordingly, the number of trying rule patterns for $R(d)$ specified in (4) is $(2^4 - 1) \times 6 = 90$. If the function of reduct is not implemented, the number is $(6^6 - 1) \times 6 = 279,930$, which burdens the second stage with a heavy load. From here, the effectiveness of the improved STRIM can be seen.

Table 4. An example of rule candidates.

Trying $CP(k)$	$C(1)C(2)C(3)C(4)C(5)C(6)$	$f = (n_1, n_2, n_3, n_4, n_5, n_6)$	z
1	004400	$(2, 1, 1, 101, 2, 1)$	21.45
2	002200	$(0, 91, 0, 2, 0, 0)$	21.43
...	...	...	...
5	005500	$(1, 0, 1, 2, 78, 2)$	19.07
6	440000	$(2, 0, 0, 76, 2, 1)$	18.70
...	...	...	...
12	001100	$(63, 2, 0, 0, 2, 2)$	16.7
13	000400	$(71, 62, 68, 168, 74, 73)$	9.63
14	004000	$(74, 67, 79, 170, 72, 73)$	9.32
15	303000	$(5, 4, 39, 5, 7, 6)$	9.26
16	400000	$(69, 74, 61, 154, 65, 61)$	8.90
17	404000	$(13, 6, 13, 45, 4, 6)$	8.86
...	...	...	...
20	400400	$(14, 8, 7, 48, 11, 11)$	8.57
...	...	...	...

Table 4 shows examples of the rule candidates arranged in descending order from z value obtained from the distribution of decision attribute values $f = (n_1, n_2, ..., n_{M_D})$ by applying the improved STRIM to the dataset shown in Table 3. We can see the following from the table:

(1) The trying rule $CP(k = 1) = 004400$ makes bias of the distribution of D like $f = (2, 1, 1, 101, 2, 1)$ at $D = 4$ intensively. Accordingly, the candidate is a rule for $D = 4$. The intensity of the bias can be measured by the z value as mentioned in Step 2.

(2) There are inclusion relationships between rule candidates, for example, $U(CP(1) = 004400) \subset U(CP(14) = 004000)$, $U(CP(17) = 404000) \subset U(CP(14) = 004000)$, while the z value of $CP(1) >$ that of $CP(14)$ and the z value of $CP(14) >$ that of $CP(17)$.

The second stage at Line 12 in Fig. 3 arranges their candidates and represents them with only $CP(1)$, which happens to coincide with the specified rule in (4). Table 5 shows the last results induced through the first and second stages with D, f, p-value, accuracy and coverage besides $CP(k)$. Here, accuracy and coverage are defined as follows:

$$\text{accuracy} = \frac{|U(CP(k)) \cap U(D = d)|}{|U(CP(k))|}, \quad \text{coverage} = \frac{|U(CP(k)) \cap U(D = d)|}{|U(D = d)|},$$

and they are often used for showing the indexes of the validity of the induced rules in Rough Sets theory. The improved STRIM induced all of twelve rules

Table 5. Estimated rules for the decision table in Table 3 by improved STRIM.

Trying $CP(k)$	$C(1)C(2)C(3)$ $C(4)C(5)C(6)$	D	$f = (n_1, n_2, n_3,$ $n_4, n_5, n_6)$	p-value(z)	Accuracy	Coverage
1	004400	4	$(2, 1, 1, 101, 2, 1)$	2.09E−102(21.45)	0.94	0.20
2	002200	2	$(0, 91, 0, 2, 0, 0)$	3.37E−102(21.43)	0.98	0.19
3	110000	1	$(91, 3, 2, 1, 10)$	6.57E−92(20.3)	0.93	0.18
4	330000	3	$(1, 1, 89, 3, 0, 0)$	6.33E−91(20.19)	0.95	0.17
5	005500	5	$(1, 0, 1, 2, 78, 2)$	2.15E−81(19.07)	0.93	0.16
6	440000	4	$(2, 0, 0, 76, 2, 1)$	2.92E−78(18.70)	0.94	0.15
7	003300	3	$(1, 4, 77, 0, 0, 1)$	6.70E−77(18.52)	0.93	0.15
8	550000	5	$(1, 1, 2, 0, 75, 3)$	9.15E−77(18.51)	0.91	0.15
9	660000	6	$(0, 3, 1, 3, 0, 76)$	5.32E−76(18.41)	0.91	0.15
10	006600	6	$(3, 3, 3, 3, 0, 73)$	8.69E−67(17.22)	0.86	0.15
11	220000	2	$(0, 60, 1, 2, 0, 2)$	2.46E−63(16.76)	0.92	0.12
12	001100	1	$(63, 2, 0, 0, 2, 2)$	3.63E−63(16.74)	0.93	0.13
13	303000	3	$(5, 4, 39, 5, 7, 6)$	9.56E−21(9.27)	0.59	0.076

Table 6. An arrangement of Car Evaluation dataset of UCI.

Unified attribute value	$C(1)$: buying	$C(2)$: maint	$C(3)$: doors	$C(4)$: person	$C(5)$: lug boot	$C(6)$: safety	D: class (freq.)
1	vhigh	vhigh	2	2	small	low	unacc (1210)
2	high	high	3	4	med	med	acc (383)
3	med	med	4	more	big	high	good (69)
4	low	low	5more	–	–	–	vgood (65)

specified in advance from the decision table with $N = 3,000$, and also one extra rule. However, there are clear differences between them in the indexes of accuracy and coverage.

6 An Example of Application for an Open Dataset

This paper applied SRM for the "Car Evaluation" dataset included in the literature [12]. Table 6 shows the summaries and specifications of the dataset: $|C| = 6$, $|C(1)| = 4$,..., $|\{D\}| = 4$, $N = |U| = |C(1)| \times, ..., \times |C(6)| = 1,728$ which consists of every combination of condition attributes' values and there were no conflicted or identical samples. The frequencies of D extremely incline toward $D = 1$ as shown in Table 6.

Table 7 shows the results obtained by SRM and suggests the following:

(1) Given $1.0E-5$ as the critical p-value, $C(3)$ is commonly redundant at $D = l$ $(l = 1, ..., 4)$.

(2) With regard to the if-then rule of $D = 1$, $C(5)$ is redundant besides $C(3)$. In the same way, so are $C(2)$ and $C(5)$ at $D = 2$, as well as $C(5)$ at $D = 3$.

Table 7. Results by SRM for Car Evaluation dataset.

		$C(1)$	$C(2)$	$C(3)$	$C(4)$	$C(5)$	$C(6)$
$D = 1$	χ^2	22.94	19.24	2.58	111.00	8.81	118.81
	p-value	4.16E−05	2.44E−04	4.62E−01	7.88E−25	1.22E−02	1.59E−26
$D = 2$	χ^2	11.77	10.77	3.19	192.56	6.52	194.25
	p-value	8.21E−03	1.30E−02	3.64E−01	1.53E−42	3.85E−02	6.59E−43
$D = 3$	χ^2	84.33	84.33	0.39	34.70	0.26	36.26
	p-value	3.61E−18	3.61E−18	9.42E−01	2.92E−08	8.78E−01	1.34E−08
$D = 4$	χ^2	70.20	28.60	4.23	33.08	37.69	130.00
	p-value	3.87E−15	2.72E−06	2.38E−01	6.57E−08	6.53E−09	5.90E−29

Table 8. Examples of contingency table and χ^2 test by SRM ((a): $D = 1$ vs. $C(j)$, (b): $D = 4$ vs. $C(j)$ $(j = 1, ..., 6)$).

(a) $D = 1$						
$V_{C(i)}$	$C(1)$	$C(2)$	$C(3)$	$C(4)$	$C(5)$	$C(6)$
1	***360***	***360***	326	***576***	450	***576***
2	324	314	300	312	392	357
3	268	268	292	322	368	277
4	258	268	292	–	–	–
χ^2	22.94	19.24	2.58	111.00	8.81	118.81
p-value	4.16E−05	2.44E−04	4.62E−01	7.88E−25	1.22E−02	1.59E−26
(b) $D = 4$						
$V_{C(i)}$	$C(1)$	$C(2)$	$C(3)$	$C(4)$	$C(5)$	$C(6)$
1	–	–	10	–	–	–
2	–	13	15	30	25	–
3	26	***26***	20	35	***40***	***65***
4	***39***	***26***	20	–	–	–
χ^2	70.20	28.60	4.23	33.08	37.69	130.00
p-value	3.87E−15	2.72E−06	2.38E−01	6.57E−08	6.53E−09	5.90E−29

Table 8 shows examples of the contingency tables of $D = 1$ (a) and $D = 4$ (b), and their χ^2 by SRM, and suggests the following knowledge:

(1) With regard to the if-then rules of $D = 1$, (a) the frequencies of $C(1) = 1$, $C(2) = 1$, $C(4) = 1$ and $C(6) = 1$ are distinctively high. Accordingly, the if-then rules of $D = 1$ are supposed to be constructed by the combinations of them as shown in the knowledge (2) studied in Sect. 4.

(2) In the same way, the if-then rules of $D = 4$ are constructed by the combinations of $C(1) = 4$, $C(2) = 4$, $C(5) = 3$ and $C(6) = 3$.

Table 9. Estimated rules of $D = 1$ (a) and $D = 4$ (b) for Table 6

(a) $D = 1$						
Trying $CP(k)$	$C(1)C(2)C(3)$ $C(4)C(5)C(6)$	D	$f = (n_1, n_2, n_3, n_4)$	p-value(z)	Accuracy	Coverage
1	000100	1	$(576, 0, 0, 0)$	3.51E−56(15.75)	1.00	0.476
2	000001	1	$(576, 0, 0, 0)$	3.58E−56(15.75)	1.00	0.476
3	110000	1	$(108, 0, 0, 0)$	2.52E−12(6.90)	1.00	0.089
(b) $D = 4$						
Trying $CP(k)$	$C(1)C(2)C(3)$ $C(4)C(5)C(6)$	D	$f = (n_1, n_2, n_3, n_4)$	p-value(z)	Accuracy	Coverage
1	400003	4	$(52, 33, 20, 39)$	1.08E−50(14.93)	0.271	0.600
2	000033	4	$(88, 64, 0, 40)$	7.93E−37(12.62)	0.208	0.615
3	000303	4	$(49, 96, 12, 35)$	3.84E−37(10.73)	0.182	0.538

Corresponding to Tables 8, 9 shows estimated rules of $D = 1$ (a) and $D = 6$ (b). With regard to rules of $D = 1$, the improved STRIM clearly induces their rules. To express those rules by use of the original notation in Table 6, if person = "2" ∨ safety = "low" ∨ buying = "vhigh" ∧ maint = "vhigh" then class = "unacc" is obtained with accuracy = 1.0 and coverage = $1,008/1,210 \approx 0.833$. In the same way, three examples of the trying rule of $D = 4$ satisfying the testing condition $n\hat{p}(j, k) \geq 5$ (for $D = 4$, $n \geq \frac{5}{0.04} = 125$) are shown in Table 9 (b) although their $n_d = \max(n_1, n_2, ..., n_{M_D})$ is not satisfied at $D = 4$ (in the table $D = 4$ is forcibly entered). The first rule that if buying = "low" ∧ safety = "high" then class = "vgood" is thought to be proper since the p-value is the best and the indexes of accuracy and coverage are moderate among the trying rules for $D = 4$. Both estimated rules for $D = 1$ and 4 coincide with our common sense.

7 Conclusions

The Rough Sets theory has been used for inducing if-then rules from the decision table. The first step in inducing the rules is to find reducts of the condition attributes. This paper retested the conventional reduct methods LEM1 [2] and DMM [3] by a simulation experiment after summarizing the conventional Rough Sets theory and pointed out their problems. Then, this paper proposed a new statistical reduct method (SRM) to overcome the problems of the conventional method from the view of STRIM [8–11]. STRIM including SRM was developed and its validity and usefulness were confirmed by a simulation experiment and application to an open dataset of UCI for machine learning. The improved STRIM should be recognized to be particularly useful for not only reducts of condition attributes but also inducing if-then rules.

References

1. Pawlak, Z.: Rough sets. Int. J. Inf. Comput. Sci. **11**(5), 341–356 (1982)
2. Grzymala-Busse, J.W.: LERS — a system for learning from examples based on rough sets. In: Słowiński, R. (ed.) Intelligent Decision Support — Handbook of Applications and Advances of the Rough Sets Theory. Theory and Decision Library, vol. 11, pp. 3–18. Kluwer Academic Publishers, Amsterdam (1992)
3. Skowron, A., Rauser, C.M.: The discernibility matrix and functions in information systems. In: Słowiński, R. (ed.) Intelligent Decision Support — Handbook of Applications and Advances of the Rough Sets Theory. Theory and Decision Library, vol. 11, pp. 331–362. Kluwer Academic Publishers, Amsterdam (1992)
4. Pawlak, Z.: Rough set fundamentals; KFIS Autumn Coference Tutorial, pp. 1–32 (1996)
5. Ślęzak, D.: Various approaches to reasoning with frequency based decision reducts: a survey. In: Polkowski, L., Tsumoto, S., Lin, T.Y. (eds.) Rough Set Method and Applications, vol. 56, pp. 235–285. Physical-Verlag, Heidelberg (2000)
6. Bao, Y.G., Du, X.Y., Deng, M.G., Ishii, N.: An efficient method for computing all reducts. Trans. Jpn. Soc. Artif. Intell. **19**(3), 166–173 (2004)
7. Jia, X., Shang, L., Zhou, Z., Yao, Y.: Generalized attribute reduct in rough set theory. Knowl.-Based Syst. **91**, 204–218 (2016). Elsevier
8. Matsubayashi, T., Kato, Y., Saeki, T.: A new rule induction method from a decision table using a statistical test. In: Li, T., Nguyen, H.S., Wang, G., Grzymala-Busse, J., Janicki, R., Hassanien, A.E., Yu, H. (eds.) RSKT 2012. LNCS (LNAI), vol. 7414, pp. 81–90. Springer, Heidelberg (2012). doi:10.1007/978-3-642-31900-6_11
9. Kato, Y., Saeki, T., Mizuno, S.: Studies on the necessary data size for rule induction by STRIM. In: Lingras, P., Wolski, M., Cornelis, C., Mitra, S., Wasilewski, P. (eds.) RSKT 2013. LNCS (LNAI), vol. 8171, pp. 213–220. Springer, Heidelberg (2013). doi:10.1007/978-3-642-41299-8_20
10. Kato, Y., Saeki, T., Mizuno, S.: Considerations on rule induction procedures by STRIM and their relationship to VPRS. In: Kryszkiewicz, M., Cornelis, C., Ciucci, D., Medina-Moreno, J., Motoda, H., Raś, Z.W. (eds.) RSEISP 2014. LNCS (LNAI), vol. 8537, pp. 198–208. Springer, Cham (2014). doi:10.1007/978-3-319-08729-0_19
11. Kato, Y., Saeki, T., Mizuno, S.: Proposal of a statistical test rule induction method by use of the decision table. Appl. Soft Compt. **28**, 160–166 (2015). Elsevier
12. Asunction, A., Newman, D.J.: UCI machine learning repository. University of California, School of Information and Computer Science, Irvine (2007). http://www.ics.edu/~mlearn/MlRepository.html
13. Walpole, R.E., Myers, R.H., Myers, S.L., Ye, K.: Probability and Statistics for Engineers and Scientists, 8th edn., pp. 374–377. Pearson Prentice Hall, New Jersey (2007)
14. Walpole, R.E., Myers, R.H., Myers, S.L., Ye, K.: Probability and Statistics for Engineers and Scientists, 8th edn., pp. 361–364. Pearson Prentice Hall, New Jersey (2007)