A Logical Method to Predict Outcomes After Coronary Artery Bypass Grafting

Tsutomu Sasao
Meiji University,
Kawasaki, 214-8571 Japan
sasao@ieee.org

Anders Holmgren
Umeå University
Umeå, Sweden
anders.holmgren@umu.se

Patrik Eklund
Umeå University
Umeå, Sweden
peklund@cs.umu.se

***Abstract*—This paper analyzes data from coronary artery bypass grafting (CABG) using decision functions to represent rules. The data was collected at the University Hospital in Umeå, Sweden. The data contains pre-, intra-, and postoperative detail from 2975 heart operations during 1993-96. Each instance is represented by 14 preoperative variables, 4 intraoperative variables, and 9 postoperative variables. A logical method is used to predict the postoperative variables using preoperative variables. First, each postoperative variable is represented as a decision functions of preoperative variables. Then, for each postoperative variable, a minimal set of preoperative variables is derived. And finally, each postoperative variable is represented by a minimum set of rules using preoperative variables. With this method we can predict postoperative outcome, where prediction using preoperative data only is of particular interest e.g. for surgery scheduling.**

***Index Terms*—multi-valued logic, partially defined function, classification, decision tree, imbalanced data set, variable minimization, discretization, domain reduction, rule reduction**

I. Introduction

Given a set of data, data mining is a technique to find a set of useful rules to represent the data. C4.5 [13] and CART (Classification and regression tree) [3] are algorithms to derive decision trees from the set of integer vectors. C4.5 uses entropy to find the decision variables, while CART uses Gini index. Rules can be derived from the decision trees.

This paper shows an alternative method to derive such rules. The method consists of four steps.

1) **Discretization**. Convert the data consisting of real numbers into that of integers.
2) **Domain reduction**. Merge the intervals to reduce the dynamic range of the variables.
3) **Variable reduction**. Reduce the number of variables to represent the partially defined function by minimum covering.
4) **Rule reduction**. Simplify the table, and derive sum-of-products expressions (SOPs) using logic minimization for partially defined functions.

With this method, we analyzed the data in coronary artery bypass grafting (CABG). The data contains pre, intra, and postoperative details of heart operations of 2975 instances. Each instance is represented by 14 preoperative variables, 4 intraoperative variables, and 9 postoperative variables.

A logical method is used to represent the postoperative variables by preoperative variables only. In this way, we can predict the outcome of operations, which is quite useful for scheduling of surgery.

The rest of this paper is organized as follows: Section II introduce CABG. Section III introduce the method used in this paper. Section IV explains the data set used in this paper. Section V shows how to convert numerical data into integer data. Section VI shows the experimental results. Section VII analyzes operative deaths in detail. Section VIII shows the outline of the system, and Section IX concludes the paper.

II. Coronary Artery Bypass Grafting

Coronary Artery Bypass Grafting (CABG) is a surgical procedure used to treat coronary heart disease. CABG disease is the narrowing of the coronary arteries : the blood vessels that supply oxygen and nutrients to the heart muscle. One way to treat the blocked or narrowed arteries is to bypass the blocked portion of the coronary artery with a piece of a healthy blood vessel from elsewhere in the body. Blood vessels, or grafts, used for the bypass procedure may be pieces of a vein from the leg or an artery in the chest.

In an operation, multiple bypass grafts may be used. So, the surgery is complex and takes many hours. During operation, a heart-lung bypass machine is often used. After operation, the patient is sent to the intensive care unit (ICU). After the ICU, the patient is sent to a patient bedroom in the hospital. Since the facility and stuff are limited, doctors have to estimate the outcome of the operations.

The outcome depends not only on the status of the heart disease, but also on the status of kidneys, liver, and lungs.

The outcomes include, D30 (death within 30 days), and REOPBEED (reoperation caused by bleeding). Although such undesirable events are rare, doctors have to estimate the risks of such events. For example, the probabilities of D30=1 and REOPBLEED=1 are less than 1.5%, and 3.7% respectively, in this data set. In data mining, rare events are hard to predict. Such data sets are called **imbalanced** [9].

III. Logical Method to Derive Rules

Logical methods to derive rules from a set of instances have been developed for many years. Related research can be found in [1], [2], [6], [10], [16], [21]. In this part, we introduce the idea by using two examples.

Example 3.1: In a hypothetical hospital, a doctor made diagnosis for 6 patients. In Table 3.1, x_1, x_2, x_3 and x_4 show

TABLE 3.1
EXAMPLE WITH FOUR VARIABLES

x_1	x_2	x_3	x_4	f
0	0	0	1	0
0	1	1	0	0
1	0	0	0	0
0	1	0	1	1
1	0	0	1	1
1	1	0	0	1

symptoms: say, x_1 shows high fever, x_2 shows headache, x_3 shows sore throat, x_4 shows general aches and pain, and f shows the influenza.

From this table, two sets of rules can be generated.

The first set of rules is
"If x_1 and x_4 are true, or if x_2 is true and x_3 is false, then $f = 1$."

The second set of rules is
"If x_1 and x_2 are true, or x_1 and x_4 are true, or x_2 and x_4 are true, then $f = 1$." By using logical expressions, they are represented as follows:

Rules 1: $\mathcal{F}_1 = x_1x_4 \vee x_2\bar{x}_3$.

Rules 2: $\mathcal{F}_2 = x_1x_2 \vee x_1x_4 \vee x_2x_4$.

Note that Rules 1 require four variables, while Rules2 require three variables.

For the patient having the symptoms $(x_1, x_2, x_3, x_4) = (1, 1, 1, 1)$, both rules produce $f = 1$. However, for the patient having the symptoms $(x_1, x_2, x_3, x_4) = (0, 1, 1, 1)$, Rules 1 derive $f = 0$, while Rules 2 derive $f = 1$. ■

In Table 3.1, six combinations are shown. An input combination such that $f(x_1, x_2, x_3, x_4) = 1$ is a **positive instance**, while an input combination such that $f(x_1, x_2, x_3, x_4) = 0$ is a **negative instance**. Such combinations form the **training data** in machine learning. On the other hand, the input combination $(x_1, x_2, x_3, x_4) = (1, 1, 1, 1)$ is missing in Table 3.1. There are $2^4 - 6 = 10$ missing combinations. For such combinations, the function values are not known. Such input combinations are called **unseen data**. We want to predict the outcomes for unseen data.

We are going to construct an SOP that is consistent with the training data.

For example, $\mathcal{F}_1 = x_1x_4 \vee x_2\bar{x}_3$ is an SOP of Table 3.1. The SOP $\mathcal{F}_1$ shows that $f(1, 1, 1, 1) = 1$, which is not shown in Table 3.1. From a simplified SOP, one can predict the outcomes for unseen data [20]. Also, note that when $(x_1, x_2, x_3x_4) = (0, 0, 0, 0)$, $\mathcal{F}_1$ predicts that $f(x_1, x_2, x_3, x_4) = 0$, which is not contained in Table 3.1. In $\mathcal{F}_1$, products x_1x_4 and $x_2\bar{x}_3$ corresponds to **rules**.

Example 3.2: In the same hypothetical hospital, the same doctor made diagnosis for 8 patients. In Table 3.2, $x_1, x_2, \ldots, x_6$ show the results of test: say, x_1 shows PET (Positron Emission Tomography); x_2 shows tumor marker; x_3 shows ultrasonic echo; x_4 shows MRI (Magnetic Resonance Imaging); x_5 shows endoscopy; x_6 shows CT (Computed Tomography) ; and f shows the malignant tumor.

TABLE 3.2
EXAMPLE WITH SIX VARIABLES

x_1	x_2	x_3	x_4	x_5	x_6	f
1	0	0	0	1	1	0
0	0	1	0	1	1	0
0	0	0	1	0	0	0
0	1	0	0	1	0	0
1	0	0	1	1	1	1
1	1	1	1	1	1	1
1	1	0	0	1	1	1
0	1	1	0	1	0	1

TABLE 3.3
THREE VARIABLES ARE NOT SUFFICIENT

x_2	x_3	x_4	f
0	0	0	0
0	1	0	0
0	0	1	0
1	0	0	0
0	0	1	1
1	1	1	1
1	0	0	1
1	1	0	1

From this table, two sets of rules can be derived:
Rules 1:

$$\mathcal{F}_1 = x_2\bar{x}_3\bar{x}_4x_6 \vee \bar{x}_2\bar{x}_3x_4x_6 \vee x_2x_3\bar{x}_4\bar{x}_6 \vee x_2x_3x_4x_6.$$

Rules 2: $\mathcal{F}_2 = x_1x_2x_6 \vee x_1\bar{x}_3x_6 \vee \bar{x}_1x_2x_3\bar{x}_6$.

Both sets of rules require four variables (tests). Rules 1 require x_2, x_3, x_4 and x_6, while Rules 2 require x_1, x_2, x_3 and x_6. Note that x_2, x_3 and x_4 are essential, but they are not sufficient.

Table 3.3 shows the relation between (x_2, x_3, x_4) and f. When $(x_2, x_3, x_4) = (0, 0, 1)$, the value of f can be both 0 and 1. In other words, the third and the fifth entries are **inconsistent** or **conflicting**. In this case, $\{x_2, x_3, x_4\}$ are not sufficient to represent f. ■

Methods to derive simplified set of rules from the set of instances are shown in [7], [8], [14]. A method to derive minimal sets of variables to represent a given set of instances is shown in p.84 of [14], p. 122 of [15], and p. 31 of [17].

In these examples, for simplicity, only two-valued variables were used. However, multiple-valued variables can be also used. Also, the number of classes can be greater than two, i.e., f can take many values [16], [18].

IV. DATA SET

Table 4.1 shows the details of the variables in the data set. In the last column, the number inside of the parenthesis denotes the number of distinct values. Note that V01 (age) is denoted by years and months.

These variables (V01 $\sim$ V27) are categorized into

- Preoperative: V01 $\sim$ V14,
- Intraoperative:V15 $\sim$ V18, and
- Postoperative: V19 $\sim$ V27.

The original data set consists of 2975 instances. After removing instances with incomplete entries in V01 $\sim$ V18, the number of remaining instances became 1480.

For each postoperative variable, we constructed a decision function. Since V20,V21,V22 take numerical values, we set cut points as follows:

- V20 (INTENSH): 24 hours,
- V21 (DAYSPOST): 10 days, and
- V22 (RESPTIME): 24 hours.

In this way, we had 9 decision functions of 18 input variables. Note that V01 (AGE), V08 (PRECREA), V15 (CLAMPTIME), and V16 (ECCTIME) take more than a hundred distinct values.

V. PRE-PROCESSING OF DATA [19]

Among 27 variables, V01, V08, V15, V16 are numerical variables and take more than a hundred distinct values, which are hard to manipulate by a logic minimization program. So, we try to reduce the domain of the variables.

Discretization [11] converts the data consisting of real numbers into that of integers. To do this, the values of the variables are sorted in ascending order, and for each distinct value, unique integer starting from 1 is assigned so that the magnitude relation is kept. For example, Table 4.2 can be converted into Table 4.3.

Domain reduction merges the domain to reduce the dynamic range of the variables. Consider the function $f(x)$, where x takes integer values. If $f(a) = f(a+1)$, then the domains for a and $a{+}1$ are merged. For example, Table 4.3 can be reduced to Table 4.4. In this way, a table with continuous variables are converted into one with integer variables.

Two instances are **inconsistent** or **conflicting** if the attributes are the same, but belong to different classes. The set of instances is **consistent** if there is no inconsistent pair in the set. We assume that the given set of instances is consistent.

VI. EXPERIMENTAL RESULTS

We reduced the number of values for V01, V08, V15 and V16, so that the reduction never affects the accuracy of decision [19].

A. Rules using Minimal Set of Variables

Each postoperative variable was represented as a partially defined function of both preoperative and intraoperative variables. Then, the number of variables was minimized, and finally the SOP was simplified by MINI10 [20] to reduce the number of the products (i.e., rules). The first five columns of Table 6.1 shows the results.

Note that these functions can be represented with at most three variables. Unfortunately, they contain at least one intraoperative variable (V15, V16, V17, or V18). Note that intraoperative variables are available during surgery. Prediction of postoperative variables (i.e., outcomes) without using intraoperative variables are preferable for surgery scheduling.

B. Rules Using Only Preoperative Variables

Rules that predict prognosis of operations using only preoperative variables are extremely helpful. Information on the preoperative variables are easily available.

Thus, we tried to find rules that consist of preoperative variables only. We applied a program to derive all possible minimal sets of variables necessary to represent the function. Then, we selected a solution that contains preoperative variables only. And, finally, we represented the function by a minimum SOP. The last three columns of Table 6.1 shows the results. For most functions, the necessary number of rules or variables increased. The number of rules in Table 6.1 shows one for the simpler rules between the positive and the negative classes[1]. All the functions were represented with preoperative variables only. This is quite helpful for surgery scheduling. We can see that V01 (Age), V04 (Function Class), and V08 (Preoperative S-Creatinine) are important variables. This result is consistent with those of other studies [4], [5], [12]: They used eGFR (estimated glomerular filtration rate), which is more sensitive test of kidney function compared to S-Creatinine.

C. AOQUAL

An interesting question is that whether AOQUAL (V18: aorta quality) can be represented by preoperative variables or not. Fortunately, the answer is yes. There are 15 minimal solutions. One of the solutions is shown in the bottom of Table 6.1. Note that AOQUAL takes three values.

D. Conflict Rate for the Training Set

In the previous subsections, we selected instances whose variables V01$\sim$V18 are complete. Only 1480 or fewer instances out of 2975 instances were used to find the necessary set of variables, (i.e., were used for the training data). The analysis showed that to represent the functions, only a few variables are necessary.

For example, to represent D30, only V01, V04 and V08 are used. So, we selected 2756 instances whose entries for V01, V04, V08, and D30 are complete, and checked if V01, V04, and V08 are sufficient to represent D30. Unfortunately, there exist one pair of conflicting instances. (i.e. inconsistent data pair). That is, the entries for V01, V04, and V08 are the same, but that of D30 are different. However, if we ignore one of these instances, D30 can be represented by V01, V04, and V08.

Let the Conflict Rate be

$$\text{Conflict Rate} = \frac{\text{Number of Conflicting Pairs}}{\text{Size of the Instance Set}}.$$

Table 6.2 summarizes the results for all the postoperative variables. Note that the conflict rates are very low. From this, we can expect low error rates for unseen instances.

Especially, for POPKIDNEY and RETINTENS, no conflict exists. In all cases, the number of rules increased to cover more instances.

[1]In this paper, the positive class corresponds to undesirable events, such as death. Undesirable events are rare in many cases.

TABLE 4.1
HEART OPERATION DATA SET

Variable	Acronym	Explanation	Values
V01	AGE	Years and Months	Numerical (354)
V02	AP	Angina pectoris	STABLE, INSTABLE, ACUTE, OTHER
V03	REOP	Reoperation	YES, NO
V04	FUNCT CLASS	Function class	I, II, IIIA, IIIB, IV
V05	LV FUNCT	Left ventricle function	GOOD, WEAKENED, BAD
V06	NVESSELS	Number of vessels diseased	Numerical (6)
V07	HSTAMST	Left main stenosis	YES, NO
V08	PRECREA	Preoperative S-Creatinine	Numerical (125)
V09	CEREBRDIS	Cerebrovascular disease	YES, NO
V10	PREVCABG	Previous CABG operation	YES, NO
V11	SMOKER	Smoker	YES, NO
V12	LUNGDIS	Lung disease	YES, NO
V13	LIVERDIS	Liver disease	YES, NO
V14	DIABETES	Diabetes	YES, NO
V15	CLAMPTIME	Aorta closed (min)	Numerical (119)
V16	ECCTIME	Heart/lung machine (min)	Numerical (167)
V17	PANAST	Number of anastomoses	Numerical (8)
V18	AOQUAL	Aorta quality	NORMAL, SLIGHTLY CHANGED, SEVERELY CHANGED
V19	D30	Died within 30 days after operation	YES, NO
V20	INTENSH	Hours in intensive care	Numerical
V21	DAYSPOST	Length of stay in hospital	Numerical
V22	RESPTIME	Respiratortime (hours)	Numerical
V23	REOPBLEED	Reoperation caused by bleeding	YES, NO
V24	POPATRFLIM	Postoperative atrial fibrillation	YES, NO
V25	POPCONF	Postoperative confusion	YES, NO
V26	POPKIDNEY	Postoperative kidney insufficiency	YES, NO
V27	RETINTENS	More than once in intensive care	YES, NO

TABLE 4.2
WITH CONTINUOUS VARIABLES

ID	X_1	X_2	X_3	f
1	10.6	25	4.9	1
2	11.2	33	4.9	1
3	11.5	18	4.0	1
4	11.6	22	5.5	1
5	11.6	25	4.4	1
6	11.7	28	4.4	1
7	11.7	37	4.7	1
8	11.7	30	3.7	2
9	11.9	30	4.8	1
10	11.9	35	3.6	2
11	12.1	30	3.8	1
12	12.2	32	4.3	1
13	12.2	34	4.1	2
14	12.2	35	4.4	2
15	12.4	23	3.5	2
16	12.5	37	3.5	2
17	12.6	32	3.3	2
18	12.8	41	3.9	2
19	12.9	28	3.7	2
20	13.3	36	4.1	2

TABLE 4.3
WITH INTEGER VARIABLES

ID	Y_1	Y_2	Y_3	f
1	1	4	13	1
2	2	8	13	1
3	3	1	7	1
4	4	2	14	1
5	4	4	10	1
6	5	5	10	1
7	5	12	11	1
8	5	6	4	2
9	6	6	12	1
10	6	10	3	2
11	7	6	5	1
12	8	7	9	1
13	8	9	8	2
14	8	10	10	2
15	9	3	2	2
16	10	12	2	2
17	11	7	1	2
18	12	13	6	2
19	13	5	4	2
20	14	11	8	2

TABLE 4.4
AFTER DOMAIN REDUCTION

ID	Z_1	Z_2	Z_3	f
1	1	3	8	1
2	1	7	8	1
3	1	1	4	1
4	1	1	8	1
5	1	3	7	1
6	2	4	7	1
7	2	9	8	1
8	2	5	1	2
9	3	5	8	1
10	3	8	1	2
11	4	5	2	1
12	5	6	6	1
13	5	8	5	2
14	5	8	7	2
15	6	2	1	2
16	6	9	1	2
17	6	6	1	2
18	6	10	3	2
19	6	4	1	2
20	6	8	5	2

E. Accuracy for the Test Sets

In the previous experiment, rules were generated using 1480 or fewer instances. Such sets of instances are **training sets**. For example, in the case of REOPBLEED, 1480 instances were used for the **training set** to generate 49 rules. 1289 instances were used for the **test set**: each instance in this set was incomplete, but the entries for V01,V06, V08 and V23 are complete. As shown in Fig. 6.1, the number of incorrectly classified instance was counted to compute the error rate. The error rate of the test set is defined as

$$\text{Error Rate} = \frac{\#\text{ of Incorrectly Classified Instances}}{\text{Size of the Test Set}}.$$

Table 6.3 shows the error rates. The number of rules in Table 6.3 includes both that for the positive and the negative cases [2]. Thus, they are greater than those of Table 6.1.

VII. DETAILED ANALYSIS FOR D30

In this part, we analyze the influence of $V01$ (Age), $V04$ (Function class according to New York Heart Association), and $V08$ (Preoperative S-Creatinine), to $V19$ (D30: Died within 30 days after operation). Among 2975 instances, 44 instances *died*, while 2931 instances *survived.*

[2] We used rules for both the positive and the negative cases to compute the error rate.

TABLE 6.1
RULES USING PREOPERATIVE VARIABLES ONLY

Acronym	# of Inst.	Minimal Variables # of Rules	# of Var.	Var.	Preoperative Variables only # of Rules	# of Var.	Var.
D30	1480	5	2	V01,V15	5	3	**V01, V04, V08**
INTENSH (24 hours)	1480	51	3	V01,V05,V16	76	6	**V01, V04**, V06, V07,**V08**, V12
DAYSPOST (10 days)	1477	55	3	V01,V05,V16	90	6	**V01, V04**, V07, **V08**, V11, V14
POPATRFLIM	1115	27	3	V01,V08,V16	45	6	**V01, V04**, V05, V07, **V08**, V11
POPCOF	1115	9	3	V01,V08,V16	11	3	**V01**, V07,**V08**
POPKIDNEY	1115	13	2	V01,V16	5	3	**V01, V04, V08**
REOPBLEED	1480	8	3	V01,V15,V16	22	3	**V01**, V06,**V08**
RESPTIME(24 hours)	1457	36	3	V01,V08,V16	66	3	**V01, V04, V08**
RETINTENS	1480	9	3	V01,V08,V16	20	4	**V01, V04**, V07,**V08**
AOQUAL	1480				151	6	**V01**, V02, V07, **V08**, V10, V11

TABLE 6.2
RULES USING PREOPERATIVE VARIABLES: CONFLICT RATE

Acronym	# of Instances	# of Rules	# of Conflicts	Conflict Rate ($\times 10^{-3}$)
D30	2756	10	1	0.36
INTENSH (24 hours)	2442	96	3	1.23
DAYSPOST (10 days)	2400	90	6	2.50
POPATRFLIM	1965	89	6	3.06
POPCOF	2214	33	5	2.26
POPKIDNEY	2209	11	0	0.00
REOPBLEED	2769	24	7	2.53
RESPTIME (24 hours)	2724	84	10	3.67
RETINTENS	2758	37	0	0.00
AOQUAL	2437	202	10	4.10

TABLE 6.3
ERROR RATES FOR THE TEST SETS

Acronym	Size of Test Set	# of Rules	# of Errors	Error Rate (%)
D30	1275	13	19	1.49
INTENSH (24 hours)	962	98	8	0.83
DAYSPOST (10 days)	923	146	22	2.38
POPATRFLIM	850	101	59	6.94
POPCOF	1099	35	46	4.18
POPKIDNEY	1094	18	16	1.46
REOPBLEED	1289	46	44	3.41
RESPTIME (24 hours)	1267	93	54	4.26
RETINTENS	1278	61	61	4.77
AOQUAL	957	151	101	10.55

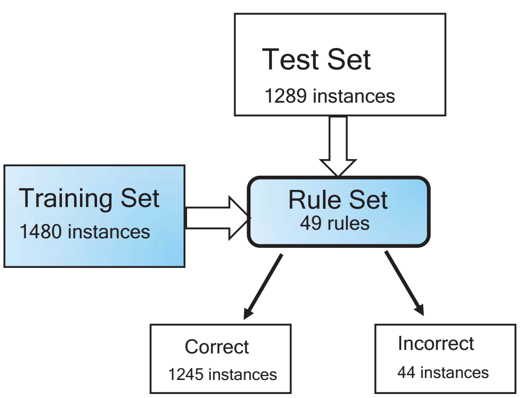

Fig. 6.1. Method to compute error rate (In the case of REOPBLEED).

We selected 1480 instances that had complete entries for all the preoperative, intraoperative variables, and V19. Among 1480 instances, 13 instances *died*, while 1467 instances *survived*. Fig. 6.2 shows the positional cube notation [19] of the rules for D30, generated from 1480 instances. The number of rules for *died* instances is 8, while that for *survived* instances is five. To analyze the properties of instances, the products are *slimmed* [7], i.e., the number of 1's in a row is reduced.

The rules consist of four parts: Y_1, Y_2, Y_3 and $D30$. Y_1 takes 25 values, and corresponds to $V01$ (Age); Y_2 takes 4 values, and corresponds to $V04$ (Function class); Y_3 takes 25 values, and corresponds to $V08$ (S-Creatinine); and $V19$ takes 2 values, and corresponds to $D30$ (Died or not). The first 8 rows cover 13 instances for *died*, while the last 5 rows cover 1467 instances for *survived*. $V01$ (age) ranges from 35.2 to 85.0, while Y_1 ranges from 1 to 25. $V04$ (Function class) takes one of values in $\{I, II, IIIA, IIIB, IV\}$, while Y_2 ranges from 1 to 4, and $Y_2 = 2$ corresponds to Class IIIA. $V08$ (S-Creatinine) ranges from 48 to 689, while Y_3 ranges from 1 to 25. For example, the first cube in Fig.6.2 corresponds to the product

$$Y_1^{\{12\}} Y_2^{\{3\}} Y_3^{\{18\}}.$$

It shows that if $Y_1 = 12$ and $Y_2 = 3$ and $Y_3 = 18$, then $D30 = 1$. It also shows that if $V01$ (Age) is 71.7 and $V04$ (Function Class) is IIIA or IIIB, and $V08$ (S-Creatinine) is 124.0, then $D30 = 1$. The first 8 rows not only cover 13 instances for *died*, but also many unseen instances.

From Fig. 6.2, we can observe that *died* instances occurred only when $Y_2 \neq 1$, which correspond to class III or IV. Independent research [5] also mentions that "New York Heart Association class III or IV is a significant predictor" for $D30$.

Note that 1467 *survived* instances are represented by five rules. For example, the last row of Fig. 6.2 covers $11 \times 2 \times 2 = 44$ instances, since Y_1 part contains 11 ones, Y_2 part contains 2 ones, and Y_3 part contains 2 ones. In addition to $\{V01, V04, V08\}$, $\{V01, V05, V08\}$,

Y_1	Y_2	Y_3	$D30$
1234567890123456789012345	1234	1234567890123456789012345	12
0000000000001000000000000	0010	0000000000000000010000000	10
0000000000001010100000010	0010	0000000001000000000000000	10
0000000001010101000101010	0001	0000000000001000000000000	10
0001000000101010101010101010	0001	0000010001000100110000000	10
0001000000001000101000010	0100	0100000000000000010000000	10
0001000000001010101000010	0010	0000000000000000000001000	10
0101101011010101010101010	0111	0001000000000001000100010	10
0001000000000001010000010	0010	0000000100000000010000000	10
1110110101111010100111101	1111	1010101100001000000000000	01
1011110111111011011011101	1101	1011111111101010011011111	01
1010010101101010000010101	0011	0001000001000000010001000	01
1010011111111111111110101	1110	1100010010111111101110101	01
1010010101101010000010101	0011	0100000000010000000000000	01

Fig. 6.2. Generated rules for D30.

$\{V01, V02, V06, V08\}$, and $\{V01, V06, V08, V14\}$ are minimal sets of preoperative variables to represent $D30$. This shows that $V01$ and $V08$ are essential.

VIII. Outline of the Rule Generation System

A. Requirements for the Data

The training set must be a consistent set of enough instances. If the training set has a pair of inconsistent instances, then one of the pair must be removed from the training set.

B. Specifications of the Current System

- The data set is represented by an EXCEL csv file, where entries are numbers.
- Input variables can be real numbers or integers, while the output variable must be positive integers showing the class.
- The system finds the most important set of variables, and produces a set of rules to classify unseen instances.
- The system also generates all possible minimal sets of variables to represent the function. A user can select the best one.
- The generated rules are represented by a positional cube notation [7], [14].

C. Limitation of the Method

A logical method efficiently selects a minimal set of variables, and derives a set of rules that covers all the instances for each class. If there exists an instance that belongs to a certain class, then a rule that covers the instance is generated. However, the frequency of instances is not considered.

On the other hand, in a statistical method, the frequency of instances is considered. If the frequency of instances is very low compared with other instances, then such instances may be neglected.

IX. Conclusions and Comments

This paper showed a method to derive rules for a given set of instances. Unlike conventional methods that use decision trees, it first reduces the domain, and then produces a sparsely defined decision function. Then, the number of variables is minimized. And, finally, multiple-valued input expressions are simplified to reduce the number of rules. The method produces a complete set of rules for a given set of instances. That is, all the instances are covered by the rules.

For each postoperative variable, a minimal set of variables to represent the variable was generated. Analysis shows that the error rates are very low.

The analysis of D30 shows that V01 (Age), V04 (Function class according to New York Heart Association) and V08 (preoperative S-Creatinine) are important to predict the outcomes. These results are consistent with those of other studies [4], [5], [12].

The merit of logical approach is that the explanation of the decision is clear to both patients and medical doctors [1], [6].

The prediction method developed in this paper complements traditional statistical methods, and provides opportunity for future analysis.

Acknowledgments

The first author was supported in part by a Grant-in-Aid for Scientific Research of the JSPS. The second author maintains the Thoracic database of the Department of Public Health and Clinical Medicine at Umeå University, from which data used in this paper has been selected. Reviewer's comments significantly improved the readability.

References

[1] G. Alexe, S. Alexe, T. O. Bonates, and A. Kogan, "Logical analysis of data: the vision of Peter L. Hammer," *Annals of Mathematics and Artificial Intelligence*, Vol. 49, pp. 265-312, July 2007.

[2] E. Boros, P. L. Hammer, T. Ibaraki, A. Kogan, E. Mayoraz, and I. Muchnik, "An implementation of logical analysis of data," *IEEE Transactions on Knowledge and Data Engineering*, Vol. 12, No. 2, pp. 292-306, March 2000.

[3] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, *Classification and Regression Trees*, CRC Press 1984.

[4] D. M. Charytan, S. S. Yang, S. McGurk, and J. Rawn, "Long and short-term outcomes following coronary artery bypass grafting in patients with and without chronic kidney disease," *Nephrol Dial Transplant*, Vol. 25. No. 11. pp. 3654-63, Nov. 2010.

[5] M. B. Chonchol, et al, "Long-term outcomes after coronary artery bypass grafting: preoperative kidney function is prognostic," *J.of Thoracic and Cardiovascular Surgery*, Vo. 134, No. 3, pp. 683-9, Sep. 2007.

[6] P. L. Hammer and T. O. Bonates, "Logical analysis of data–An overview: From combinatorial optimization to medical applications," *Annals of Operations Research,*, Vol. 148, No. 1. pp. 203-225, Nov. 2006.

[7] S. J. Hong, R. G. Cain, and D. L. Ostapko, "MINI: A heuristic approach for logic minimization," *IBM J. Res. and Develop.*, pp. 443-458, Sept. 1974.

[8] S. J. Hong,"R-MINI: An Iterative approach for generating minimal rules from examples," *IEEE Trans. Knowl. Data Eng.* Vol. 9, No. 5. pp. 709-717, 1997.

[9] B. Krawczyk, "Learning from imbalanced data: open challenges and future directions," *Progress in Artificial Intelligence*,Vo. 5, pp. 221-232, 2016.

[10] M. Lejeune, V. Lozin, I. Lozina, A Ragab, S. Yacout, "Recent advances in the theory and practice of logical analysis of data," *European Journal of Operational Research*, Vol. 275, Issue 1, No. 16, pp. 1-15, May 2019.

[11] H. Liu, F. Hussain, C. L. Tan and M. Dash, "Discretization: An Enabling Technique," *Data Mining and Knowledge Discovery*, Vol. 6, pp. 393-423, 2002.

[12] K. Minakata, et al, "Preoperative chronic kidney disease as a strong predictor of postoperative infection and mortality after coronary artery bypass grafting," *Circulation Journal*, Vol. 78, No. 9, pp. 2225-2231, 2014.

[13] J. R. Quinlan, *C4.5: Program for Machine Learning*, San Mateo, Morgan Kaufmann 1993.

[14] T. Sasao, *Switching Theory for Logic Synthesis*, Kluwer Academic Publishers, 1999.

[15] T. Sasao, *Memory-Based Logic Synthesis*, Springer, 2011.

[16] T. Sasao, "On a minimization of variables to represent sparse multi-valued input decision functions," *International Symposium on Multiple-Valued Logic* (ISMVL-2019), Fredericton, Canada, pp. 182-187, May 21-23, 2019

[17] T. Sasao, *Index Generation Functions,* Morgan & Claypool, 2020.

[18] T. Sasao, "On the minimization of variables to represent partially defined classification functions," *International Symposium on Multiple-Valued Logic*, (ISMVL-2020), Miyazaki, Japan, pp. 115-123, May 20-22, 2020.

[19] T. Sasao, "A method to generate classification rules from examples," *International Symposium on Multiple-Valued Logic*, (ISMVL-2022), pp.176-181, May 2022.

[20] T. Sasao, "Easily reconstructable logic functions," *International Symposium on Multiple-Valued Logic*, (ISMVL-2023), May 22-24, 2023.

[21] E. Triantaphyllou, *Data Mining and Knowledge Discovery via Logic-Based Methods:* Theory, Algorithms, and Applications, Springer, 2010.