A knowledge discovery from incomplete coronary artery disease datasets using rough set

Noor Akhmad Setiawan*

Department of Electrical and Electronic Engineering, Universiti Teknologi PETRONAS, Bandar Seri Iskandar, Tronoh 31750, Perak, Malaysia and
Department of Electrical Engineering, Universitas Gadjah Mada, Jl. Yacaranda Sekip Unit IV Yogyakarta, Indonesia
E-mail: noorwewe@yahoo.com
E-mail: noorwewe@ugm.ac.id
*Corresponding author

P.A. Venkatachalam and M.H. Ahmad Fadzil

Department of Electrical and Electronic Engineering, Universiti Teknologi PETRONAS, Bandar Seri Iskandar, Tronoh 31750, Perak Darul Ridzuan, Malaysia
E-mail: drpav@rediffmail.com
E-mail: fadzmo@petronas.com.my

Abstract: Incompleteness of datasets is one of the important issues in the area of knowledge discovery in medicine. This study proposes a rough set theory (RST)-based knowledge discovery from coronary artery disease (CAD) datasets when there are only small number of objects and contain missing data (incomplete). At first, RST combined with artificial neural network (ANN) is developed to impute the missing data of the datasets. Then, the knowledge that is discovered from imputed datasets is used to evaluate the quality of the imputation. After that, RST is applied to extract rules from the imputed datasets. This will result in a large number of rules. Rule selection based on the quality of extracted rules is investigated. All the evaluation and selection are based on the complete datasets. Finally, the selected small number of rules is evaluated. The discovered selected rules are used as a classifier on the diagnosis of the presence of CAD to demonstrate their good performance.

Keywords: coronary artery disease; CAD; imputation; incomplete data; knowledge discovery; rough set; rule selection.

Reference to this paper should be made as follows: Setiawan, N.A., Venkatachalam, P.A. and Ahmad Fadzil, M.H. (2011) 'A knowledge discovery from incomplete coronary artery disease datasets using rough set', Int. J. Medical Engineering and Informatics, Vol. 3, No. 1, pp.60–77.

Biographical notes: Noor Akhmad Setiawan is currently a PhD student in the Department of Electrical and Electronic Engineering, Universiti Teknologi PETRONAS, Malaysia. He is also a Senior Lecturer in the Electrical
1 Introduction

The problem of developing a model that can explain or categorise the data is found in many areas. There are two methods to model the data: mathematical and empirical. Empirical method is used when there are difficulties to obtain mathematical model. Knowledge discovery is one of the empirical methods that is widely used. Knowledge discovery is very important in the medicine decision support systems. The discovered knowledge is used to diagnose and predict result or conclusion. Usually the knowledge is in the form of ‘if-then’ rules (Ohm, 1999).

One of the important steps of knowledge discovery tasks is data preprocessing. Missing data imputation is part of data preprocessing. The quality of a knowledge discovery can be degraded when the data consist of many missing values. Therefore, imputation method has significant impact to the quality of a knowledge discovery. A large number of high quality complete datasets are not always available. It is very difficult to get both large number and complete datasets. The knowledge discovery must be able to handle incomplete and very limited number of datasets.

Coronary artery disease (CAD) is a kind of heart disease. It happens when the coronary arteries which supply blood to the heart itself are narrowing or blocked. Almost of all heart attacks are caused by narrowing or blockage of the coronary arteries resulting from fatty deposits called plaque. Clinical diagnosis of CAD based only on symptoms may be difficult. Some tests should be performed to get more accurate diagnosis. There are many factors, called risk factors, that have relation to increased risk of CAD, but their significance and prevalence have not been precisely determined. The interpretation of all symptoms, risk factors and clinical tests to draw conclusion of the presence of CAD is not always an easy task (Phibbs, 2007). Therefore, it is necessary to build a rule-based decision support systems that can help or assist physician to draw the conclusion. The rules must be derived from the evidences of the patients. The rules should be of a reasonable number in order to understand easily.
Rough set theory (RST) is a new mathematical method to deal with imprecise, vague and uncertain datasets. This theory was originally proposed by Pawlak (1982). It is based on indiscernibility relation. Objects are said to be indiscernible if they cannot be distinguished from other objects using only certain attributes. A set is rough if its elementary set cannot be clearly and precisely defined using the available knowledge. Medical data is usually uncertain, imprecise and vague. Thus, RST can be used to deal with this uncertainty.

There are several research works on heart disease diagnosis. Many kinds of heart diseases from various sources have been investigated. A multi-layer perceptron (MLP)-based medical decision support system has been developed to diagnose five types of heart diseases (hypertension, coronary heart disease, rheumatic valvular heart disease, chronic cor pulmonale and congenital heart disease) simultaneously (Yan et al., 2006). However artificial neural network (ANN) itself cannot explain the knowledge inside it, even though the system has high accuracy. Decision system based on RST was developed for appropriate medical treatment of heart diseases (Ilczuk and Wakulicz-Deja, 2005). Data of 2,039 patients have been used. This work did not cover missing value handling. Bayesian network model of heart disease is proposed (Jayanta and Marco, 2000). The system could predict the probabilities of heart diseases and dependency among attributes related to heart diseases. Diagnosis of ischemic heart disease using various machine learning techniques was found in Kukar et al. (1997). The data of 4,000 patients was used. Extension of MLP for coronary heart disease diagnosis by making it interpretable was shown in Bologna et al. (1997). Most of these research works uses large number of patients.

CAD datasets from University California Irvine (UCI) database repository (Newman et al., 1998) attract many researchers to build decision systems. The researches on CAD diagnosis based on UCI datasets were found in Detrano et al. (1989), Gennari et al. (1989), Polat et al. (2007), Duch et al. (2000), and Tan et al. (2005). The UCICAD datasets contain 920 patients that are collected from Cleveland Clinic Foundation USA; Hungarian Institute of Cardiology, Budapest, Hungary; Veterans Administration Medical Center, Long Beach, California, USA and University Hospital, Zurich, Switzerland. Most of these researchers used Cleveland dataset that consists of 303 patients because it contains only six patients have incomplete attribute values. There are no published works imputation schemes that are applied to Hungarian, Long Beach and Switzerland datasets which contain many missing values.

This study proposes a framework to discover the knowledge of CAD datasets taken from UCI repository. The proposed method is able to handle the incompleteness of datasets using ANN and RST (ANNRST) hybridisation missing data imputation. The evaluation of imputation is not only limited to the accuracy but also considering the quality of the resulting knowledge of imputed datasets. Cleveland dataset is used to test the quality of the resulting knowledge extracted from imputed Hungarian and Long Beach datasets.

2 Rough set theory

RST as described by Pawlak (1982) deals with the analysis of classification of data tables that may consist of uncertainty.
2.1 Basic concepts of RST

The basic concepts of RST are information system or decision system and set approximation. This study deals with decision system. Thus, only decision system will be covered in this brief explanation. Consider an example of simple decision system as shown in Table 1. The decision system is defined as:

\[ DS = (U, C \cup D), \]

where \( D \not\subset C \) is called decision attribute or simply as decision.

<table>
<thead>
<tr>
<th>( x \in U )</th>
<th>( c_1 )</th>
<th>( c_2 )</th>
<th>( d )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( x_1 )</td>
<td>0</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>( x_2 )</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>( x_3 )</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>( x_4 )</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>( x_5 )</td>
<td>2</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>( x_6 )</td>
<td>0</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>( x_7 )</td>
<td>2</td>
<td>2</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 1: Example of decision table

\( U \) and \( C \) are finite non-empty set called the universe and the set of condition attributes or simply conditions.

Let \( A \subseteq C \), each subset defines an equivalence relation called indiscernibility relation which is defined as:

\[ IND_x(A) = \{(x, x') \in U \times U | \forall c \in A, c(x) = c(x')\}. \]

Using only condition in \( A \), the indiscernibility relation (2) will induce a partition of \( U \) into sets. Each object in the set cannot be discerned from other object in same set. For example \( A = \{c_2\} \), then the partition induced by (2): \( U/IND_x(A) = \{ \{x_1, x_6\}, \{x_2\}, \{x_3, x_4\}, \{x_5, x_7\} \} \). The sets of classified objects are called equivalence classes denoted as \([x]_i\).

Set approximation is used when a decision concept such as \( d \) cannot be defined in a crisp manner. It is shown in Table 1 of objects \( x_3 \) and \( x_4 \). For \( A \subseteq C \), the approximations of \( X \subseteq U \) using only information in \( A \) are a lower-approximation \( A_X \) and an upper-approximation \( \overline{A}_X \) that are defined by:

\[ A_X = \{x \mid [x]_d \subseteq X\} \]

\[ \overline{A}_X = \{x \mid [x]_d \cap X \neq \emptyset\} \]

The set of lower-approximation consists of objects which certainly belong to \( X \). The set of upper-approximation consists of objects which possibly belong to \( X \). The set that contains objects that cannot be classified as definitely inside \( X \) or outside \( X \) is called boundary region of \( X \):

\[ BR_x(X) = \overline{A}_X - \underline{A}_X \]
A set is said to be rough if $BR_d(X) \neq \emptyset$. Let $A = \{c_2\}$ and $X = \{x_1, x_4, x_6\}$ as a result of $d = 1$. Then $\overline{X} = \{x_1, x_5, x_4, x_6\}$ and $BR_d(X) = \{x_3, x_4\}$. It follows that outcome of $d = 1$ is rough since the boundary region is not empty.

Let $D$ as decision and $A \subseteq C$ as conditions, then $A$-positive region of $D$ is defined as:

$$POS_A(D) = \bigcup_{x \in U \setminus D} A$$

(6)

where $U/D$ represents the partition of $U$ according to decision $D$. For example, in Table 1, with $A = \{c_2\}$ then $POS_A(D) = \bigcup \{\{x_1, x_6\}, \{x_2\}, \{x_5, x_7\}\} = \{x_1, x_2, x_5, x_6, x_7\}$.

The condition attributes of decision system $DS$ may be redundant so they can be reduced. The reduction of $DS$ will result in reducts. A reduct is a minimal set of attributes $A \subseteq C$ such that $POS_A(D) = POS_C(D)$. A reduct is combination of conditions that can discern between objects as well as all conditions. Reducts can be computed using discernibility matrix and discernibility function (Skowron and Rauszer, 1992). A discernibility matrix of decision system $DS$ can be defined as $n \times n$ matrix $M$ with its elements:

$$m_{ij} = \begin{cases} c \in C | c(x_i) \neq c(x_j), & \text{if } d(x_i) \neq d(x_j) \\ \Phi, & \text{if } d(x_i) = d(x_j) \end{cases}$$

(7)

for $i, j = 1 \ldots n$ and $d \in D$.

The elements of $M$ are the conditions that are needed to discern object $i$ from object $j$ relative to the decision. Table 2 shows an example of decision relative discernibility matrix of Table 1. Discernibility function can be derived based on discernibility matrix. A discernibility function $f_C$ for decision system $DS$ consists of $k$ Boolean variables of Boolean function which is defined as:

$$f_C(c_1, \ldots, c_k) = \bigwedge_{1 \leq j \leq n, m_{ij} \neq \emptyset} \left( \bigvee m_{ij}^* \right)$$

(8)

where $m_{ij}^* = \{ c \in m_{ij} \}$. All the minimal reducts of the decision system may be obtained by finding the set of all the prime implicants of discernibility function. According to Table 2, the discernibility function of decision system $DS$ is $f_C(c_1, c_2) = \{ c_1 \lor c_2 \} \land \{ c_2 \}$. After simplification using Boolean algebra properties then the discernibility function becomes $f_C(c_1, c_2) = \{ c_2 \}$. Thus, there is only single reduct $\{ c_2 \}$. The concept of reduct can be used as attribute reduction or feature selection.

Sometimes reducts computed based on objects relative discernibility are more interesting than full discernibility reduct, especially for decision rule generation. Reducts that are relative to object $x \in U$ can be found using modified (8):

$$f_C(c_1, \ldots, c_k, x_j) = \bigwedge_{x_j \in U} \left( \bigvee m_{ij}^* \right)$$

(9)

For example object $x_3$ relative reducts can be calculated using object relative discernibility function (9) $f_C(c_1, c_2, c_3) = \{ c_1 \lor c_2 \}$, hence relative reducts are $\{ c_1 \}$ and $\{ c_2 \}$. All relative reducts of $x_3$ to $x_2$ can be found.
### Table 2

Example of decision relative discernibility matrix

<table>
<thead>
<tr>
<th>$x \in U$</th>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$x_3$</th>
<th>$x_4$</th>
<th>$x_5$</th>
<th>$x_6$</th>
<th>$x_7$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x_1$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$x_2$</td>
<td>$c_2$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$x_3$</td>
<td></td>
<td>$c_1, c_2$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$x_4$</td>
<td></td>
<td></td>
<td>$c_1, c_2$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$x_5$</td>
<td>$c_1, c_2$</td>
<td></td>
<td></td>
<td>$c_1, c_2$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$x_6$</td>
<td></td>
<td>$c_2$</td>
<td>$c_1, c_2$</td>
<td></td>
<td>$c_1, c_2$</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$x_7$</td>
<td></td>
<td>$c_1, c_2$</td>
<td></td>
<td></td>
<td>$c_1, c_2$</td>
<td>$c_1, c_2$</td>
<td>$c_1, c_2$</td>
</tr>
</tbody>
</table>

### 2.2 Decision rule generation

Once all of the relative reducts are determined, a set of decision rules can be generated from those reducts. Various algorithms are available to generate rules from those reducts.

Consider $DS = (U, C \cup D)$ as decision table. $\forall x \in U$ defines series $c_1(x), \ldots, c_k(x), d(x)$, where $\{c_1, \ldots, c_k\} = C$ and $\{d\} = D$. Hence, the decision rules can be generated in the form of $c_1(x), \ldots, c_k(x) \rightarrow d(x)$. $C$ can be the condition attributes of reduced form of decision table (reduct). For example using information from Table 1 by considering one of its relative reducts $= \{c_2\}$, the rules can be generated as antecedent consequent form:

- If $c_2 = 3$ then $d = 1$. \hspace{1cm} (10)
- If $c_2 = 0$ then $d = 0$. \hspace{1cm} (11)
- If $c_2 = 2$ then $d = 0$. \hspace{1cm} (12)
- If $c_2 = 1$ then $d = 1$ or $d = 0$. \hspace{1cm} (13)

Rules (10–12) is said to be deterministic. Rule (13) is indeterministic because its consequence is uncertain. The definition of rule support is described as how many objects match the corresponding rule. For example rule (12) has support $= 2$ because there are two objects that match the rule antecedent and consequent. Those objects are $x_5$ and $x_7$. Support can be used as rule filtering criterion when there are too many rules generated. Accuracy of certain rule can be calculated as its support divided by the number of objects that match its rule antecedent. Thus, rule (12) has accuracy $\frac{2}{4}$. Coverage of a rule can also be calculated by dividing support by the number of objects that match its rule consequence. Hence, rule (12) has coverage $\frac{2}{4}$.

### 3 A brief description of ANN

This study will use ANN combined with RST to impute missing attribute values of CAD datasets. This section will give brief explanation of ANN especially MLPs.

MLP is an ANN that is widely used in medical decision support system (Yan et al., 2006). It has been applied successfully to many difficult problem in many areas
An MLP is a network of nodes or neurons as processing elements arranged into layers. As shown in Figure 1, a typical MLP usually consists of three layers: input layer, hidden layer and output layer. Single hidden layer is sufficient to make MLP become a universal classifier.

The neurons are connected between layers by their corresponding weights. The propagation of the signals that pass from input to output can be modelled as:

\[ v_k = f_k \left( \sum_{i=1}^{p} w_{ik} x_i \right) \]

\[ y_j = f_j \left( \sum_{k=1}^{q} w_{jk} v_k \right) \]
where $x_1, \ldots, x_p$ are input signal including the bias. $w_k$ is the connection weight between the input $x_i$ and the neuron $k$. $v_1, \ldots, v_q$ are hidden layer’s output signal that will become the input of the output neuron $j$. $w_{kj}$ is the connection weight between the hidden layer’s output $v_k$ and the neuron $j$. $f_k(.)$ and $f_j(.)$ are the activation functions of the $k$th and $j$th neuron respectively. The sigmoid function is usually used in MLP. One of sigmoid functions is hyperbolic tangent function defined in (15).

$$f(a) = \tanh(a)$$

Learning or training in MLP is updating the weight values in the form of:

$$w_{ij}^{(n+1)} = w_{ij}^{(n)} + \Delta w_{ij}^{(n)}$$

$$\Delta w_{ij}^{(n)} = -\eta \left[ \frac{\partial E}{\partial w_{ij}} \right]$$

$$E = \frac{1}{2} \sum_j (t_j - y_j)^2$$

where $t_j$ is the target of learning at neuron $j$ and $\eta$ is learning rate parameter. There are many learning methods based on backpropagation (BP). Resilient backpropagation (Rprop) (Riedmiller and Braun, 1993) is used in this study.

### 4 Methodology

In this study, the methodology that is used consists of three parts: missing attribute value imputation, rule extraction and selection and evaluation of both imputation method and decision rules.

#### 4.1 Missing attribute value imputation

In this study, all datasets of the coronary heart disease from UCI are used. Imputation is applied to the selected objects from Hungarian and Long Beach datasets. We proposed a combination of ANNRST to impute the missing attribute values.

A study of ANNRST effectiveness is conducted by simulating missing attribute values on an attribute of complete dataset before applying to real missing attribute values. Studies on imputation of missing data in different field have been conducted (Troyanskaya et al., 2001; Grzymala-Busse and Hu, 2001; Wasito and Mirkin, 2006; Al Shalabi et al., 2006; Nelwamondo and Marwala, 2007). ANNRST uses ANN as a missing data estimator, and RST as an attribute reducer of the input (Banerjee et al., 2007; Fernandez-Riverola et al., 2007). The ability of ANN to estimate the missing attribute values is demonstrated in Junninen et al. (2004). Using RST, the complexity of ANN topology can be reduced by RST attribute reduction. Hence, the topology will be simpler but has the ability of estimating the missing attribute values that is close to the ability of ANN before attribute reduction. The required number of training samples usually increases with the dimensionality of the input space. There are other advantages having smaller input space (Du and Swamy, 2006).
Table 3  Summary of attributes (UCI heart disease dataset)

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
<th>Value description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Age</td>
<td>Age</td>
<td>Numerical</td>
</tr>
<tr>
<td>Sex</td>
<td>Sex</td>
<td>1 If male</td>
</tr>
<tr>
<td>cp</td>
<td>Chest pain type</td>
<td>0 If female</td>
</tr>
<tr>
<td>trestbps</td>
<td>Resting systolic blood pressure on admission to the hospital (mmHg)</td>
<td>Numerical</td>
</tr>
<tr>
<td>chol</td>
<td>Serum cholesterol (mg/dl)</td>
<td>1 If yes</td>
</tr>
<tr>
<td>fbs</td>
<td>Fasting blood sugar over 120 mg/dl?</td>
<td>0 If no</td>
</tr>
<tr>
<td>restecg</td>
<td>Resting electrocardiographic results</td>
<td>0 Normal</td>
</tr>
<tr>
<td>thalach</td>
<td>Maximum heart rate achieved</td>
<td>1 Upsloping</td>
</tr>
<tr>
<td>exang</td>
<td>Exercise induced angina?</td>
<td>0 If no</td>
</tr>
<tr>
<td>oldpeak</td>
<td>ST depression induced by exercise relative to rest</td>
<td>Numerical</td>
</tr>
<tr>
<td>slope</td>
<td>The slope of the peak exercise ST segment</td>
<td>1 Upsloping</td>
</tr>
<tr>
<td>ca</td>
<td>Number of major vessels coloured by fluoroscopy</td>
<td>Numerical</td>
</tr>
<tr>
<td>thal</td>
<td>Exercise thallium scintigraphic defects</td>
<td>3 Normal</td>
</tr>
<tr>
<td>num</td>
<td>Diagnosis of heart disease (angiographic disease status/presence of CAD)</td>
<td>0 If less than 50% diameter narrowing in any major vessel (CAD no)</td>
</tr>
</tbody>
</table>

As an illustration, consider there is a dataset in form of decision table that consists of \( m \) condition attributes: \( \{c_1, c_2, \ldots, c_m\} \) and single decision attribute: \( \{d\} \). Let \( \{c_2\} \) be the attribute that has missing values. Using ANN, a missing value estimator can be built to estimate the missing values of attribute \( \{c_2\} \) by using \( \{c_1, c_3, \ldots, c_m\} \) and \( \{d\} \) as the input attributes of input layer and \( \{c_2\} \) as the output attribute of output layer. Thus, there will be \( m \) input attributes and one output attribute. Using RST the decision table can be reduced using the reduct concept. Then, the condition attributes can be reduced to: \( \{c_1, c_2, \ldots, c_n\} \) where \( n < m \). For missing values of \( \{c_2\} \), an ANN imputer can be built with smaller size of input attributes: \( \{c_1, c_2, \ldots, c_n\} \) and \( \{d\} \). The diagram of ANNRST is shown.
in Figure 2, where $c_k$ denotes the $k$th attribute that has missing values. A study of imputation using simulation of missing data is carried on to obtain the effectiveness of ANNRST. Cleveland, Hungarian and Long Beach datasets are used in this study. The attributes of CAD datasets are shown in Table 3. The attributes that have real missing values which are the slope of the peak of ST segment ($slope$), number of major vessels coloured by fluoroscopy ($ca$) and exercise thallium scintigraphic defects ($thal$) are not used. The objects that most of their attributes have missing values are also removed. For example, object $x_i$ has missing value on most of its attributes (not just slope, ca and thal), hence $x_i$ is removed from the decision table. After those preprocessing, the decision table consists of 597 objects with ten conditions and single decision. ROSETTA (Ohrn, 1999) is used for RST data analysis. Boolean reasoning algorithm is applied to discretise the numerical attributes (Nguyen and Skowron, 1996). The decision table of dataset then is reduced by RST using Johnson’s (1974) reduct computation.

**Figure 2** Topology of ANNRST

![Topology of ANNRST](image)

The algorithm of discretisation operates by first creating a Boolean function $f$ from the set of candidates cuts, and then computing a prime implicant of this function. Johnson’s algorithm is applied to compute the prime implicant.

The attributes of reduct are age ($age$), resting systolic blood pressure ($trestbps$), serum cholesterol ($chol$), maximum heart rate achieved ($thalach$) and ST depression induced by exercise relative to rest ($oldpeak$). Fasting blood sugar over 120 mg/dl ($fbs$), resting electrocardiographic results ($restecg$) and exercise inducing angina ($exang$) are arbitrarily chosen as the attribute that has missing values. Ten to 300 missing values are simulated on $fbs$, $restecg$ and $exang$ attributes. The ANN topology that is used is an MLP with six inputs (all reduct conditions and decision), single hidden layer and single output ($fbs$). The number of hidden layer is chosen to be $\log_2(T)$, where $T$ is the number of training samples. In this case, the maximum missing data is 300. It means 297 samples are used as training process. The early stopping method is used with the ratio of training and validation is 0.8:0.2. Hence, the actual training samples are 238 sets. Then the hidden layer is $\log_2(238) = 7.89 \approx 8$ neurons (Wanas et al., 1998). Rprop learning method
(Riedmiller and Braun, 1993) and normalisation of inputs value by scaling the inputs between values of −1 to 1 are applied. Fifty training simulations with different weight initialisation are carried on the dataset. Nguyen-Widrow weight initialisation is used (Nguyen and Widrow, 1990). Learning rate $\eta^- = 0.5$ and $\eta^+ = 1.2$ with $\Delta_{ij}^{(0)} = 0.07$ are chosen (Riedmiller and Braun, 1993).

For comparison, ANN without RST attribute reduction, k-nearest neighbour (k-NN) and concept most common value filling (CMCF) are applied to the dataset for data imputation.

The algorithm of k-NN can be described as follows: take the row that contains missing value as the target, determine its k-NN by computing the Euclidean distance between incomplete row and complete rows, identify the k closest and impute the missing value in row that contains missing value by averaging the corresponding complete rows of the k closest. In this case, $k = 1$ because the decision table consists of non-numerical values.

Most common attribute value filling is the simplest methods to deal with missing attribute values. The value that occurs most frequent is selected to be the value of the unknown missing attribute value. CMCF is a restriction of most common attribute value filling by its concept or decision. In CMCF, the value that occurs most frequent within concept is selected to be the imputed value. CMCF is also called maximum relative frequency method or maximum conditional probability method.

ANN imputation hence uses ten input attributes (nine from conditions and one from decision) to impute fbs, restecg and exang attributes.

The accuracy among different methods is calculated. Maximum accuracy among 50 training-testing simulations of ANN and ANNRST is considered as the accuracy for these two methods. Average accuracy for 50 training-testing simulations with different weight initialisations using Nguyen-Widrow for ANN and ANNRST is calculated to find the stability of these two methods. The higher average accuracy means the higher probability to get the better accuracy.

4.2 Rule extraction and selection

The rule extraction method explained in Subsection 2.2 is applied. Filtering method based on rule support is applied to select the rules to reduce the number of rules. Support = $m$ means that rules that have support below $m$ are removed. The determination of $m$ is based on the acceptable accuracy and coverage of the classifier. This acceptable accuracy and coverage will be determined based on empirical data of the experiment. Selected rule-based classifier performance is calculated using classifier quality metrics.

4.3 Evaluation

Many researchers have evaluated the performance of imputation using the accuracy and error-based metrics on simulated missing data (Junninen et al., 2004; Troyanskaya et al., 2001; Wasito and Mirkin, 2006). The comprehensive evaluation based on classifiers to measure the quality of imputation has also been conducted in Al Shalabi et al. (2006) and Grzymala-Busse and Hu (2001). We proposed method by using the most complete dataset (Cleveland) to evaluate the quality of imputation is introduced in this study. The knowledge discovered of datasets that have many imputed missing values (Hungarian
and Long Beach) is used to classify the most complete dataset. This method is called 'imputed classifies complete'.

Evaluations are based on accuracy and coverage of the classifiers. RST is used to extract the rules from Hungarian and Long Beach datasets as the training data. The resulting three training datasets, namely ANNRST, k-NN and CMCF datasets are used for rule extraction. Using ROSETTA, there will be a lot of extracted rules. Simple filtering method based on rule support is conducted. The imputation method effects on the classifier performance on the testing data while rule filtering is applied on these rules are demonstrated. The quality metrics of classifier, namely accuracy, coverage, sensitivity, specificity are calculated. Area under curve (AUC) of receiver operating characteristics (ROC) (Fawcett, 2006) is also calculated. Michalski rule quality (Agotnes, 1999) is adopted and modified as a metrics to measure the quality of classifier based on testing data. Michalski quality = \( \mu \cdot \text{Accuracy} + (1 - \mu) \cdot \text{Coverage} \), where \( 0 \leq \mu \leq 1 \). We proposed this rule filtering method as an evaluator of the imputation performance.

5 Results and discussion

ANNRST performance compared to ANN, k-NN and CMCF on simulated datasets for attributes \( fbs \), \( exang \) and \( restecg \) are shown in Figure 3 and Figure 4. It can be seen in Figure 3 that the accuracy of ANNRST is comparable with that of the ANN and CMCF and better than that of the k-NN.

Figure 3 Comparison of imputation accuracy among methods for attributes, (a) \( fbs \) (b) \( exang \) (c) \( restecg \)
The average accuracy of ANNRST on 50 simulations running with different weight initialisation as shown in Figure 4 is slightly better than that of the ANN. It means that ANNRST is more stable than ANN to achieve better estimation accuracy.

**Figure 4** Comparison of imputation average accuracy among methods for attributes, (a) $fbs$ (b) $exang$ (c) $restecg$

The average of ten to 300 missing data imputation accuracies for attributes $fbs$, $exang$ and $restecg$ are 70.73% and 70.39% for ANN and ANNRST respectively. For average accuracy, the averages are 63.70% and 64.22% for ANN and ANNRST respectively.

The result of imputation evaluation using the extracted knowledge from imputed Hungarian and Long Beach datasets is used to classify Cleveland dataset is shown in Table 4. The classifier that are used are RST, decision tree (D-tree), k-NN and linear transfer function classifier (LTF-C), a classifier based on ANN which are available in RSES (Bazan and Szczuka, 2001). The objective of ANNRST is to impute missing values that have strong relationship with complete datasets. This objective can be achieved by considering Hungarian and Long Beach datasets which are almost incomplete as the training data and Cleveland which are almost complete as the testing data that will be classified. It is shown that ANNRST is better than CMCF and k-NN on overall accuracy.

RST classifier is used in this study for extracting rules from datasets. With three sets of data produced by ANNRST, k-NN and CMCF imputation, RST extracted 3,881, 5,566 and 1,537 rules respectively. In this section, investigation of imputation effects on the quality of RST classifier is carried on while rule filtering based on support is applied. Support $= m$ means that rules that have support below $m$ are removed. The results are
shown in Figure 5 and Figure 6. The results from Figure 5 show that ANNRST gives better performance than k-NN and CMCF.

**Table 4**  Accuracy and coverage of ‘imputed classifies complete’ on three datasets with four classifiers

<table>
<thead>
<tr>
<th>Classifiers</th>
<th>Accuracy (coverage)</th>
<th>Coverage</th>
</tr>
</thead>
<tbody>
<tr>
<td>RST</td>
<td>0.815 (1)</td>
<td>0.856 (0.482)</td>
</tr>
<tr>
<td>D-tree</td>
<td>0.759 (1)</td>
<td>0.779 (0.432)</td>
</tr>
<tr>
<td>k-NN</td>
<td>0.729 (1)</td>
<td>0.776 (1)</td>
</tr>
<tr>
<td>LTF-C</td>
<td>0.792 (1)</td>
<td>0.779 (0.617)</td>
</tr>
<tr>
<td>CMCF</td>
<td>0.792 (1)</td>
<td>0.779 (0.617)</td>
</tr>
</tbody>
</table>

The Michalski quality measure as a trade-off between accuracy and coverage shows the performance of ANNRST. The increasing of accuracy of classifier will usually be followed by decreasing of coverage. The effect of rule filtering on sensitivity, specificity and AUC of ROC can be seen in Figure 6. ANNRST is better than the others except for specificity, CMCF shows better. Sensitivity means how accurate the classifier recognises the CAD suffered patients. Specificity means how accurate the classifier recognises non-suffered patients. As a trade-off between sensitivity and specificity AUC of ROC has been introduced. It can be seen that ANNRST is still better than the others in AUC-ROC except after removing rules have support below 20.

**Figure 5**  Filtering effect of RST classifier on three different datasets on (a) accuracy, (b) coverage, and (c) Michalski quality measure ($m = 0.5$)
The number of rules among classifier is not the same. For example after filtering rules that have support below 20 the number of rules are 206, 92 and 67 for ANNRST, CMCF and k-NN. Figure 7 shows the graph of the Michalski quality measure and AUCROC of classifier versus the number of rules. Based on the investigation, ANNRST Hungarian and Long Beach datasets are chosen as the datasets that the knowledge will be discovered from. Rule filtering by removing the rules that have support below 20 results in 206 rules with their accuracy and coverage are 0.822 and 0.964.

Figure 6 Filtering effect of RST classifier on three different datasets on (a) sensitivity, (b) specificity, and (c) AUC of ROC

Figure 7 Filtering effect of RST classifier on three different datasets with the same rule number on (a) Michalski quality measure ($m = 0.5$) and (b) AUC of ROC
This set of rules from support-based filtering is chosen. The comparison of between methods of rule selection applied on testing datasets can be seen on Table 5. Two support-based filtering were implemented.

This selection method is based on support of rules on training dataset and testing dataset. The other methods are obtained from Agotnes (1999). Table 5 shows that the support-based methods give comparable performance to the others. Support-based selection has better coverage than other methods with comparable accuracy.

<table>
<thead>
<tr>
<th>Selection methods</th>
<th>Accuracy</th>
<th>Coverage</th>
<th>Number of rules</th>
</tr>
</thead>
<tbody>
<tr>
<td>Support-based (training data)</td>
<td>0.847</td>
<td>0.799</td>
<td>29</td>
</tr>
<tr>
<td>Support-based (testing data)</td>
<td>0.844</td>
<td>0.868</td>
<td>27</td>
</tr>
<tr>
<td>Michalski $m = 0.5$</td>
<td>0.845</td>
<td>0.785</td>
<td>27</td>
</tr>
<tr>
<td>Torgo</td>
<td>0.845</td>
<td>0.785</td>
<td>27</td>
</tr>
<tr>
<td>Brazdil</td>
<td>0.845</td>
<td>0.785</td>
<td>27</td>
</tr>
<tr>
<td>Pearson</td>
<td>0.845</td>
<td>0.785</td>
<td>27</td>
</tr>
<tr>
<td>Cohen</td>
<td>0.863</td>
<td>0.65</td>
<td>29</td>
</tr>
</tbody>
</table>

6 Conclusions

A framework to discover knowledge on incomplete CAD using RST has been proposed. Extensive study on missing data imputation has been carried on. On the study of the effectiveness of ANNRST imputation method which is applied on simulated missing attribute values, it can be concluded that ANNRST has comparable performance with ANN. ANNRST has also better stability in training process. ANNRST gives better average accuracy in 50 simulations running. ANN and ANNRST have better accuracy than k-NN and CMCF imputation. The quality of imputed datasets by ANNRST is also better than that of the k-NN and CMCF when its extracted knowledge is tested using Cleveland datasets. The Study of the effects of imputation in rule filtering process has been conducted. It has been shown that ANNRST imputation gives better performance than that of the k-NN and CMCF when rule filtering is applied. Hence, the datasets with ANNRST imputation is chosen to be the training data of the RST rule discovery. The method to select the important rules is also investigated. The results show support-based rule selection compared to other methods. The proposed framework can effectively discover the knowledge of incomplete CAD with acceptable number of rules and quality. Small number of rules will be easier to understand by human.

References

Agotnes, T. (1999) Filtering Large Propositional Rule Sets while Retaining Classifier Performance, Department of Computer and Information Science, Norwegian University of Science and Technology.


A knowledge discovery from incomplete coronary artery disease datasets


