Diagnosis of Hypoglycemic Episodes using a Neural Network Based Rule Discovery System

K.Y. Chan[1], S.H. Ling[2], T.S. Dillion[1], and H.T. Nguyen[2]

[1] Digital Ecosystems and Business Intelligence Institute,
Curtin University of Technology,
WA 6102
AUSTRALIA
[2]Centre for Health Technologies, Faculty of Engineering and Information Technology,
University of Technology, Sydney, NSW, Australia

Abstract

Hypoglycemia or low blood glucose is dangerous and can result in unconsciousness, seizures and even death for Type 1 diabetes mellitus (T1DM) patients. Based on the T1DM patients' physiological parameters, corrected QT interval of the electrocardiogram (ECG) signal, change of heart rate, and the change of corrected QT interval, we have developed a neural network based rule discovery system with hybridizing the approaches of neural networks and genetic algorithm to identify the presences of hypoglycemic episodes for TIDM patients. The proposed neural network based rule discovery system is built and is validated by using the real T1DM patients' data sets collected from Department of Health, Government of Western Australia. Experimental results show that the proposed neural network based rule discovery system can achieve more accurate results on both trained and unseen T1DM patients' data sets compared with those developed based on the commonly used classification methods for medical diagnosis, statistical regression, fuzzy regression and genetic programming. Apart from the achievement of these better results, the proposed neural network based rule discovery system can provide explicit information in the form of production rules which

compensate for the deficiency of traditional neural network method which do not provide a clear understanding of how they work in prediction as they are in an implicit black-box structure. This explicit information provided by the product rules can convince medical doctors to use the neural networks to perform diagnosis of hypoglycemia on T1DM patients.

Keywords: neural networks, genetic algorithm, hypoglycemic episodes, medical diagnosis, Type 1 diabetes mellitus

1. Introduction

Episodes of hypoglycemia for Type 1 diabetes mellitus (T1DM) patients, which can result in unconsciousness, seizures or even death, are common and have serious side effect in insulin therapy (DCCT 1993). Especially at night, at least 50% of all severe episodes of hypoglycemia occur during that time (Pickup 2000) because usual insulin preparations do not adequately mimic the normal patterns of endogenous insulin secretion (Yale 2004). However, it is impossible in practice to monitor the episodes of hypoglycemia by measuring the blood glucose levels around the clock. Based on the physiological parameters including heart rate and QT interval of ECG signal, the glucose levels of the T1DM can be determined and thus the episodes of hypoglycemia can be diagnosed (Harris et al. 1996).

The statistical regression method (Seber 2003) is a common empirical approach to develop such classification models for various medical diagnoses such as diabetic nephropathy (Cho et al. 2008), acute gastrointestinal bleeding (Chu et al. 2008), pancreatic cancer (Chang & Hsu 2009). However, statistical regression models are

accurate over the range of the patients' data in which they are developed. Therefore it can only be applied if the patients' data is distributed according to the developed regression model, and the correlation between dependent and independent variables does not exist. If the patients' data is irregular, the developed regression models have unnaturally too wide possibility range. Genetic programming (Gray et al. 1998) is another commonly used method to generate classification models for diagnosis purposes for examples in heart disease (Winkler et al. 2009), and Parkinson's disease (Subashini et al. 2009). In this approach, genetic operations are used to generate structures of classification models with nonlinear terms in polynomial forms, and then the least squares algorithm is used to determine the contribution of each nonlinear term of the model classification. However, it is unavoidable that patients' data involves uncertainty, due to fuzziness of measures. Therefore the genetic programming together with the least square algorithm may not yield the best classification models for diagnosis purposes, since it does not consider the fuzziness of uncertainty in measures. Neural networks (Reggia & Sutton 1998) have been used to develop classification models for medical diagnosis purposes. The advantages of using neural network approaches in diagnosis are their generalization ability in addressing both the nonlinear and fuzzy nature of the patients' data. Although neural networks have been applied in building diagnosis models for various diagnoses like gastrointestinal disorders (Aruna et al. 2007), abdominal pain (Mantzaris et al. 2008), urological dysfunctinos (Gil et al. 2009), dermatologic disease (Chang & Chen 2009), breast cancer (Tanaka & Watada 1998), heart disease (Das et al. 2009), these diagnosis models only have the capability to transform the nonlinear or fuzzy patients' data into simplified black-box structures in which no explicit knowledge or information can

observed. Because of the black-box nature of the neural networks, some medical doctors may feel uncomfortable to use neural networks for diagnosis purposes even through the approaches may achieve better accuracy diagnosis than the other explicit modeling methods like classical statistical methods or genetic programming. This could also pose serious issues of one has to provide justification for one's decision based on the implicit output of the neural network. Therefore this is essential to extract explicit information from the neural networks, so that the decision's basis is explicit.

Recently neural fuzzy networks have been applied on modeling and classification based on patients' data for medical diagnosis purposes in breast cancer (Oentaryo et al. 2008, Sim et al. 2006), prostate cancer (Keles et al. 2007), heart disease (Kannathal et al. 2006). The fuzzy neural networks model is constructed by distributing input and output relationships to the weights connecting neurons. The error value is limited to a reasonable level via sample training and used for modification of each weight value to acquire the final weight value for connection between neurons. The model of the system is constructed with these weight values. This approach is especially suitable for the construction of highly nonlinear models, and also fuzzy rules which contain certain information of the developed models can be generated (Lin et al. 2008). However, comparing with traditional network networks, more parameters need to be determined from the fuzzy neural networks, because not only parameters of the weights connecting neurons need to be determined but also the parameters inside the fuzzy rules needed to be determined. Therefore larger memory, more computational time and learning data are required than the ones required to develop neural networks. Even fuzzy rules which represent curtain information from the models can be generated, the domains of inputs

and outputs represented by the fuzzy rules are all fuzzy. Medical doctors may find it difficult to make diagnosis decisions based on those fuzzy rules.

In this paper, a neural network based rule discovery system, which consists of a neural network based classification unit and a rule based extraction unit, is proposed to perform diagnosis of hypoglycemic episodes in T1DM patients. The neural network based classification unit is used for determining hypoglycemic episodes in T1DM patients using the specified physiological parameters, and a set of rules, which describe the domains of physiological parameters for which hypoglycemic episodes occur, is extracted from the neural network classification unit by a rule based extraction unit. Based on a set of training data collected from TIDM patients, a neural network based classification unit was developed by a genetic algorithm, which has a multi-objective fitness function with two goals. It maximizes the number of TIDM patients with hypoglycemic episodes diagnosed correctly with hypoglycemic episodes, and the number of TIDM patients in normal conditions diagnosed correctly in normal conditions. It also minimizes the number of TIDM patients with hypoglycemic episodes diagnosed wrongly under normal conditions and the number of TIDM patients under normal conditions wrongly diagnosed with hypoglycemic episodes. The neural network classification unit was validated by a set of testing data, and satisfactory results can be found. After the development of the neural network based classification unit, explicit rules were extracted by a rule discovery unit in which the explicit rules were generated based on a data set generated by the neural network classification unit. The explicit were validated by a set of testing data, and satisfactory results can be also found. The neural network based rule discovery system compensates for the limitation of neural network not providing explicit

information.

This paper is organized as follows: Section 2 presents the proposed neural network based rule discovery system, which consists of a neural network based classification unit and a rule based extraction unit. Section 3 describes the nature of the data collected from the TIDM patients, and the presents the evaluation and validation results of the proposed neural network based rule discovery system. Finally a conclusion is given in Section 4.

2. Diagnosis of Hypoglycemic Episodes

Diagnosis of hypoglycemic episodes is essential for T1DM patients especially at night, largely because episodes of hypoglycemia are common while usual insulin preparations do not adequately mimic the normal patterns of endogenous insulin secretion (Yale 2004). Based on the medical doctors' experiences (Harris et al. 1996), the blood glucose levels of T1DM, y, which indicate whether the patients are hypoglycemia, are significantly related to several physiological parameters of which the three most significant ones have been identified as follows:

- rates of changes of heart rates, x_1
- corrected QT interval of electrocardiogram signal, x_2
- rates of changes of corrected QT interval, x_3

Hypoglycemic episodes are suggested as those in which the patient's blood glucose level, $y < 3.33$ mmol/l (60mg/dl) (DCCT 1993, DCCT 1995). The neural network based rule discovery system consists of two main units: a) a neural network based classification unit which is trained by a genetic algorithm based on a set of patients' data.

It is used to determine whether the patient is hypoglycemia based on the three physiological parameters; b) a rule discovery unit which is trained by a genetic algorithm based on a set of data generated by the neural network based classification unit. It is used to discover informative rules from the neural network which is black-box in nature. The proposed neural network based rule discovery system is illustrated in Figure 1, and the two main units are discussed in Section 2.1 and Section 2.2 respectively.

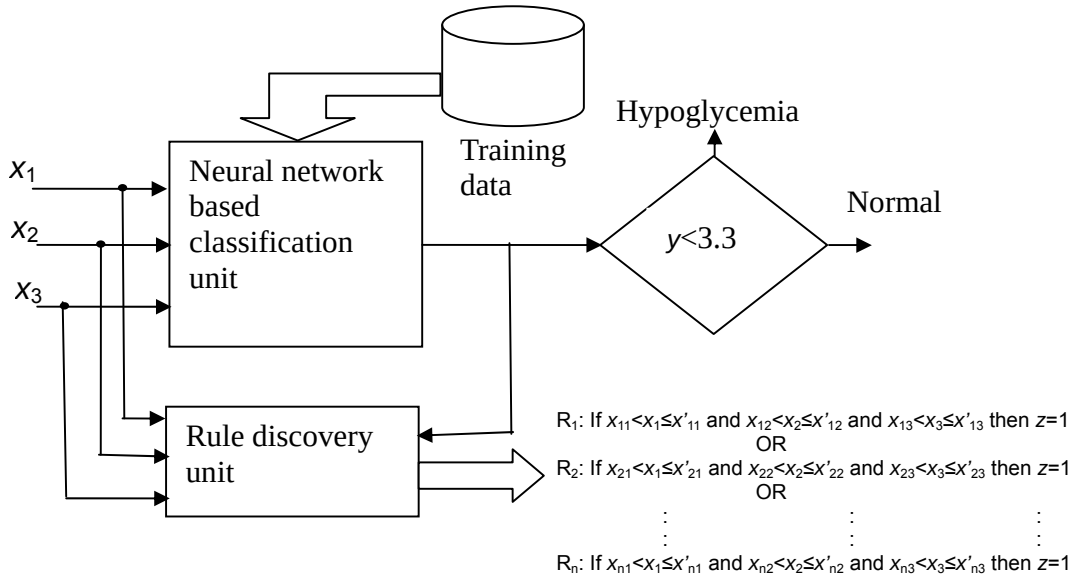

Figure 1 Neural network based rule discovery system

2.1 Neural network based classification unit

A three-layer feed-forward neural network was used to indicate whether the patient is hypoglycemia or not in relation to the three patient's physiological parameters, rate of change of heart rate, x_1, corrected QT interval of electrocardiogram signal, x_2 and rate of change of corrected QT interval, x_3. Its structure is shown in Figure 2. It consists of an input layer including the three physiological parameters x_1, x_2 and x_3 which are fed in, and the output layer which produces the indication of hypoglycemia z = 0 or 1. The

patient is in normal condition with z=0 if the glucose level y of the patient is higher than 3.3 mmol/l. Otherwise, they are in hypoglycemia with z=1 if the glucose level y of the patient is lower than 3.3 mmol/l. The hidden layer links the physiological parameters and the indication of hypoglycemia together and allows for complex, nonlinear interactions between the three physiological parameters to produce the indication of hypoglycemia.

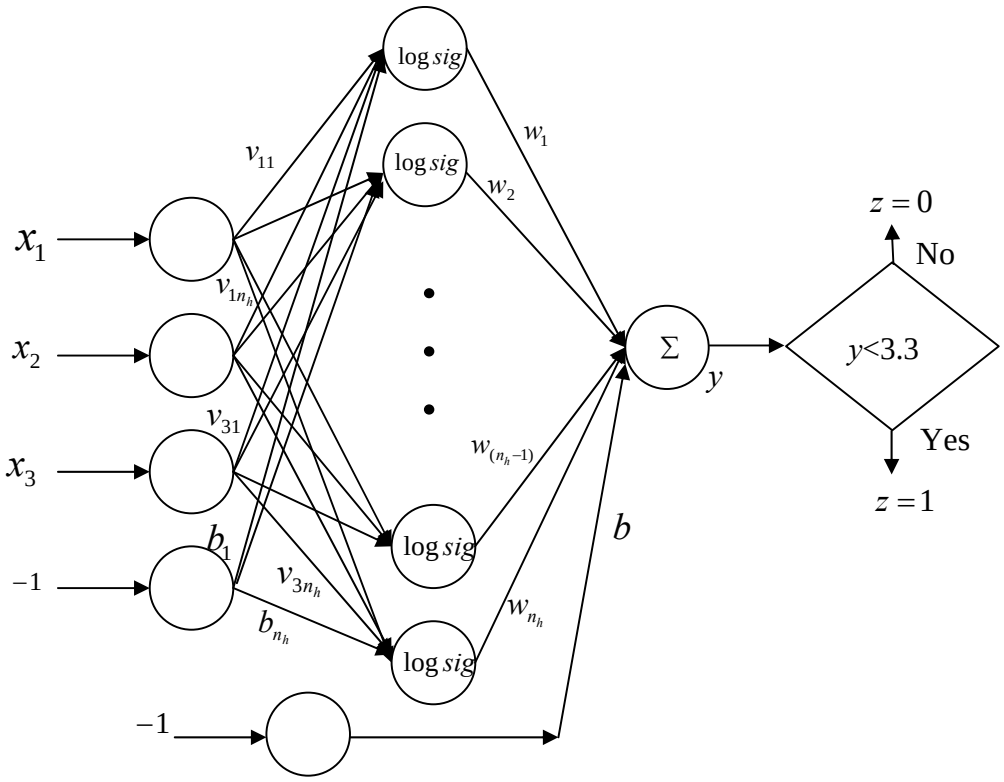

Figure 2 Structure of the neural network based classification unit

Referring to Figure 2, the input-output relationship of the proposed three-layer neural networks for the glucose level y can be written as follows:

$$y = \sum_{j=1}^{n_h} w_j \log sig\left[\sum_{i=1}^{3}\left(v_{ij}x_i - b_j\right)\right] - b \qquad (1)$$

where n_h denotes the number of the hidden nodes; w_j , j=1, 2, …, n_h , denotes the

weight of the link between the *j*-th hidden node and the output; v_{ij} , *i*=1,2,3 and *j*=1, 2, …, n_h, denotes the weight between the *i*-th input and the *j*-th hidden node; b_j and *b*, denote the biases for the *j*-th hidden nodes and output nodes respectively; log*sig*(.) denotes the logarithmic sigmoid function:

$$\log sig(\alpha) = \frac{1}{1+e^{-\alpha}}, \qquad \alpha \in \Re. \tag{2}$$

To develop the neural network for the estimation of hypoglycemia, values of the neural network parameters (i.e. w_j, v_{ij}, b_j and *b* with *i*=1, 2, 3 and *j*=1, 2, …, n_h) and the number of hidden-nodes (i.e. n_h) used in the hidden layer need to be determined. These two settings not only affect the convergence of neural networks, but also affect the accuracy of the estimation of the neural network. Here a genetic algorithm is introduced to determine the neural network parameters. The neural network architecture are 3-n_h-1, denoted as NNA_{n_k}, where n_h is the number of hidden nodes. First the neural network architecture, NNA_{n_k}, is selected and then a neural network is constructed according to the neural network architecture. After that, the neural network parameters (i.e.: w_j, v_{ij}, b_j and *b* with *i*=1, 2, 3 and *j*=1, 2, …, n_h) are searched by a genetic algorithm, and finally the trained error of the developed neural network is calculated.

The genetic algorithm first generates a population of strings, represented by the parameters of the neural network with an architecture NNA_{n_k} randomly. The strings are expressed as $[w_j, v_{ij}, b_j, b]$ where $-1 \geq w_j, v_{ij}, b_j, b \geq 1$, *i*=1,2,3 and *j*=1,2,…, n_h. The length of the strings is equal to the total number of neural network parameters, which is $n_h + 3n_h + n_h + 1 = 5n_h + 1$. Then the fitness of each string is evaluated by a fitness

function which is defined as:

$$fitness = \lambda\varsigma + (1-\lambda)\kappa \tag{3}$$

where ς and κ are the sensitivity and the specificity of the T1DM problem represented by the string respectively, and $\lambda \in [0\ 1]$ is a constant value to control the importance of the sensitivity and specificity. ς and κ are defined as follows:

$$\varsigma = \frac{N_{TP}}{N_{TP} + N_{FN}}, \tag{4}$$

$$\kappa = \frac{N_{TN}}{N_{TN} + N_{FP}}, \tag{5}$$

where N_{TP} is number of true positives which implies the sick people correctly diagnosed as sick; N_{FN} is number of false negatives which implies the sick people wrongly diagnosed as healthy; N_{FP} is number of false positives which implies healthy people wrongly diagnosed as sick; and N_{TN} is number of true negatives which implies healthy people correctly diagnosed as healthy (Carvalho & Freitas 2000). The objective of the genetic algorithm is to maximize both ς and κ. The population of strings is evolved and improved iteratively by the evolution operation, crossover and mutation, until a termination condition is met. In genetic algorithms, there can be many possible termination conditions. Here the termination condition is met when a string with $\varsigma > 0.75$ and $\kappa > 0.5$, which satisfies the requirement of the diagnosis, is found.

2.2 Rule discovery unit

Although the hypoglycemia z of a T1DM can be determined based on the three physiological parameters, x_1, x_2 and x_3 by using the neural network based classification unit discussed in Section 2.3, it is difficult to extract explicit information from the neural

network based classification unit solely based on (1) which is in an implicit structure. A methodological based on a rule discovery unit developed by a genetic algorithm is discussed to extract explicit rules from the neural network based classification unit.

Based on the neural network based classification unit as shown in Figure 3, hypoglycemia z of a T1DM can be determined based on the three physiological parameters, x_1, x_2 and x_3, where $\Re_1$, $\Re_2$, …. and $\Re_n$ are the domains regarding to the three physiological parameters, x_1, x_2 and x_3 that hypoglycemia of a T1DM occurs. Based on those domains, n rules in the following form can be extracted. In this section, a rule discovery unit which is used to extract rules from the neural network based classification unit is proposed. First, a set of data regarding the relationship between the three physiological parameters and hypoglycemia is generated based on the neural network based classification unit. Then the rules consist of a conjunction of hypoglycemia and true recommended domains of the three physiological parameters and these are developed by the genetic algorithm. Based on the rule discovery system, informative rules involving a domain of three physiological parameters with respect to the hypoglycemia can be extracted from the data sets. The rules generated can be represented as follows:

$$R_1 : X_{11} < x_1 \le X_{11}' \text{ and } X_{12} < x_2 \le X_{12}' \text{ and } X_{13} < x_1 \le X_{13}' \text{ then } z = 1$$

$$R_2 : X_{21} < x_1 \le X_{21}' \text{ and } X_{22} < x_2 \le X_{22}' \text{ and } X_{23} < x_1 \le X_{23}' \text{ then } z = 1$$

$$\vdots \qquad \vdots \qquad \vdots$$

$$\vdots \qquad \vdots \qquad \vdots$$

$$R_n : X_{n1} < x_1 \le X_{n1}' \text{ and } X_{n2} < x_2 \le X_{n2}' \text{ and } X_{n3} < x_1 \le X_{n3}' \text{ then } z = 1$$

where the rules R_1, R_2, …. and R_n represent the domains $\Re_1$, $\Re_2$, …. and $\Re_n$ of the

three physiological parameters in Figure 3 respectively.

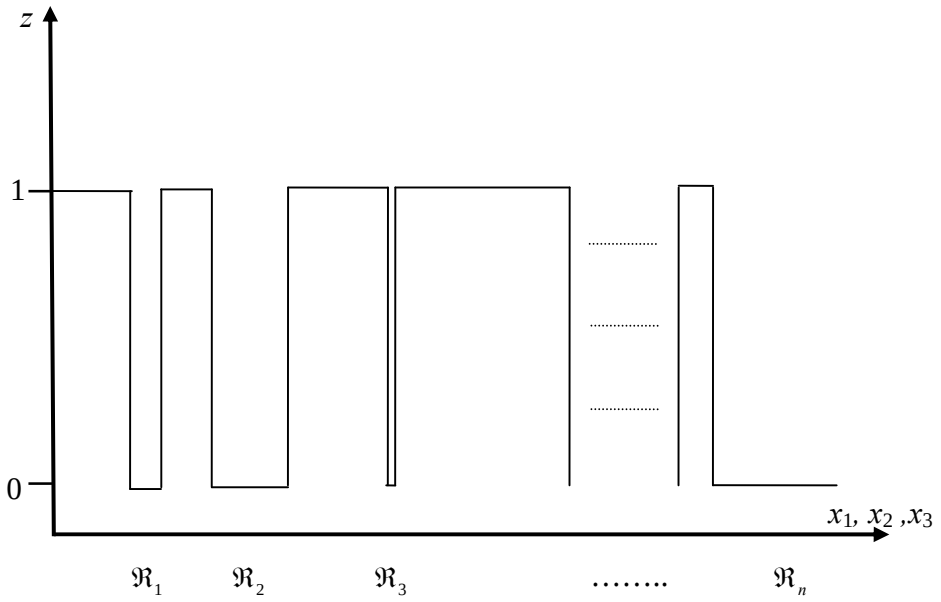

Figure 3 An illustration of the extract rules represent the domain of the physiological parameters

The pseudocode of the genetic algorithm, which is used to search the parameters (i.e. $X_{11}, X_{11}', X_{12}, X_{12}', X_{13}, X_{13}', X_{21}, X_{21}', X_{22}, X_{22}', X_{23}, X_{23}', ... X_{n1}, X_{n1}', X_{n2}, X_{n2}', X_{n3}, X_{n3}'$) on the *n* rules (i.e. R_1, R_2, …. and R_n), is shown as follows:

n=1; /* *where n is the number of rules*
{
$n = n + 1$
Initialize $\Omega(t)=[\alpha_1(t), \alpha_2(t),\ldots \alpha_{POP}(t)]$
Evaluate all $\alpha_k(t)$ according to the fitness function defined in equation (3)
while ($t<T_{gen}$) **do /*** T_{gen} *is the predefined number of generations*
{
Parent Selection $\Omega(t+1) =[\alpha_1(t), \alpha_2(t), \ldots \alpha_{POP}(t)]$
Crossover $\Omega(t+1)$
Mutation $\Omega(t+1)$
Evaluate all $\alpha_k(t+1)$ according to the fitness function defined in equation (3)
Reinsert $\Omega(t)$ by $\Omega(t+1)$
$t=t+1$
}
Select the best $\alpha_k(t+1)$ among all strings to be the rules extracted from the neural network based classification unit
} **While (**the accuracy of the rules is satisfactory**)**
***/** *The algorithm terminates if a string which* $\varsigma>0.75$ *and* $\kappa>0.5$
**/ can achieve and is found.*

In the genetic algorithm, n is the number of rules extracted from the neural network. The larger n is, the more likely the data generated by the neural network is to be covered by the rules, and better accuracy is more likely to be achieved. However, a longer computational time is required to search for the optimal parameters of the rules. Therefore the value of n is initially set at 1. If no satisfied rules can be found by the genetic algorithm, n is incremented by 1 before the next genetic algorithm run. With an increased n, a better accuracy rate is more likely to be achieved.

The genetic algorithm first creates a random initial population $\Omega(t)$ of strings $[\alpha_1(t), \alpha_2(t),\ldots \alpha_{POP}(t)]$ with t=0, while POP is the number of strings of the population. The string $\alpha_i(t)$ is represented as the parameters of the n rules and can be denoted by:

$$\alpha_i(t) = \{ X_{11}^{\,i}(t), X_{11}^{\,i}(t)', X_{12}^{\,i}(t), X_{12}^{\,i}(t)', X_{13}^{\,i}(t), X_{13}^{\,i}(t)', \ldots.., X_{n1}^{\,i}(t),$$

$X_{n1}{}^{i}(t)'$, $X_{n2}{}^{i}(t), X_{n2}{}^{i}(t)'$, $X_{n3}{}^{i}(t), X_{n3}{}^{i}(t)'$ }

where all $X_{jk}{}^{i}(t)$ and $X_{jk}{}^{i}(t)'$ with j=1,2,…,n; k=1,2,3 and $X_{jk}{}^{i}(t) < X_{jk}{}^{i}(t)'$ are randomly generated within the operation domains of three physiological parameters.

Then each string is evaluated by the fitness function based on (3) that indicates the accuracy of the rules discovered. The objective of the genetic algorithm is to maximize the accuracy of the rules discovered by searching the values of the parameters of the rules. After assigning the score of accuracy to each string, the string selection process uses the score of each string to determine the selection of potential strings for the next generation. The approach of roulette-wheel, which is one of the most common selection methods, is proposed to select the strings, $\alpha_1(t)$, $\alpha_2(t)$, … $\alpha_{POP}(t)$, for the next evolutionary population $\Omega(t+1)$. Then evolution of the strings is performed by crossover and mutation.

The fitness function of the rule discovery system is used to evaluate how good a rule fits the data samples generated by the neural network based classification unit. Rules need to be evaluated during the training process in order to establish points of reference for the rule discovery system. The fitness function defined in (3) which considers the data sets as correctly classified, lift to be classified, and the wrongly classified ones are discovered. With the higher numbers of N_{TP} and N_{TN}, and the lower number of N_{FP} and N_{FN}, a better rule is generated. For a comprehensive discussion about rule-quality measures, the reader can refer to (Hand 2001). An illustration of a rule generated by the rule discovery system is shown as follows:

$$\text{if } 1.10 \le x_1 \le 1.40 \text{ and } 0.20 \le x_2 \le 0.60 \text{ and } -0.30 \le x_3 \le -0.10 \text{ then } z = 1 \quad (6)$$

where $X_{11}(t)=1.10,\quad X_{11}'(t)=1.40,\quad X_{12}(t)=0.20,\quad X_{12}'(t)=0.60,\ X_{13}(t)=-0.30,$ and $X_{13}'(t)=$-0.10, are the values from the string of the GA based knowledge discovery system. To evaluate the fitness of the rule, 4 training data sets as shown in Table 1 are used. Classifications of the training data sets are shown in the last column of Table 1.

- The 1-st data set is classified as N_{TP} class, since z=1, and also all $x_1=1.20$, $x_2=0.5$ and $x_3=-0.2$ are within the ranges $1.10\le x_1\le 1.40$, $0.2\le x_2\le 0.6$ and $-0.3\le x_3\le -0.1$ respectively. Therefore the data set is covered by the rule and is correctly classified.
- The 2-nd data set is classified as N_{TN} class, since z=0, and both $x_2=0.1$ and $x_3=-0.4$ are not within the ranges $0.2\le x_2\le 0.6$ and $-0.3\le x_3\le -0.1$ respectively. This means the data set is not covered by the rule but differs from the target class.
- The 3-rd data set is classified as N_{FN} class, since z=0, and also all $x_1=1.30$, $x_2=0.4$ and $x_3=-0.25$ are within the ranges $1.10\le x_1\le 1.40$, $0.2\le x_2\le 0.6$ and $-0.3\le x_3\le -0.1$ respectively. This means the sample is not covered by the rule but matches the rule.
- The 4-th data set is classified as N_{FP} class, as z=1, but all $x_1=1.05$, $x_2=0.1$ and $x_3=-0.2$ are within the ranges, $1.10\le x_1\le 1.4$, $0.2\le x_2\le 0.6$ and $-0.3\le x_3\le -0.1$ respectively. Therefore the data set is not covered by the rule but is wrongly classified as belonging to the target class.

In this example, the number of data sets in all N_{FN}, N_{FP}, N_{TP} and N_{TN} classes is 1. Thus based on the fitness function (3), the score of accuracy of rule (6) is calculated as:

$$fitness = \lambda\varsigma + (1-\lambda)\kappa$$
$$= 0.55 \times 0.5 + 0.45 \times 0.5 = 0.5$$

Table 1 Training data for rule (6)

Data sets	x_1	x_2	x_3	z	*Class*
1st	1.20	0.5	-0.2	1	N_{tp}
2nd	1.15	0.1	-0.4	0	N_{tn}
3rd	1.30	0.4	-0.25	0	N_{fn}
4th	1.05	0.1	-0.2	1	N_{fp}

After assigning the score of accuracy to each string, the string selection process uses the score of each string to determine the selection of potential strings for the next generation. The approach of roulette-wheel, which is one of the most common selection methods, is proposed to select the strings, $\alpha_1(t)$, $\alpha_2(t)$, $\ldots\alpha_{POP}(t)$, to the population $\Omega(t+1)$. Then evolution of the strings is performed by crossover and mutation.

Intermediate crossover (Muhlenbein & Voosen 1993), a common crossover operation for real encoding representation, is used in the genetic algorithm. It produces a new string in which the values of the parameters of the new rule are between the values of the parameters of the two rules represented by the two selected parent strings. A new string $[x_{11}, x_{11}', x_{12}, ..., x_{n3}, x_{n3}']$ is produced based on equation (7):

$$[x_{11}, x_{11}', x_{12}, ..., x_{n3}, x_{n3}'] = [y_{11}, y_{11}', y_{12}, ..., y_{n3}, y_{n3}'] + \alpha\{[y_{11}, y_{11}', y_{12}, ..., y_{n3}, y_{n3}'] - [z_{11}, z_{11}', z_{12}, ..., z_{n3}, z_{n3}']\} \quad (7)$$

where α is a scaling factor chosen uniformly at random over some interval typically $[-0.25, 1.25]$, and $[y_{11}, y_{11}', y_{12}, ..., y_{n3}, y_{n3}']$ and $[z_{11}, z_{11}', z_{12}, ..., z_{n3}, z_{n3}']$ are the two

selected parent strings. The parameter values of the new rule represented by the new string are the result of combining the parameter values of the rules represented by the parent strings according to (7) with a scaling factor α chosen for each parameter. In geometric terms, intermediate crossover is capable of producing new parameter values within a slightly larger hypercube than that defined by the parent strings but constrained by the range of the scaling factor α. Then mutation operation is carried out by randomly changing one or more parameter values in the selected string. For example, the parameter τ_j is selected to be mutated. After performing mutation operation, its value becomes:

$$\tau_j' = \tau_j + \text{MutMx} \times R_j \times \delta \tag{8}$$

where MutMx = +1 or –1 with equal probability; R_j = 0.5 × upper and lower bounds of the parameter τ_j; δ = a value in the range [0,1] for shrinking the mutation range based on Gaussian perturbation. Updated population $\Omega(t+1)$ is produced by reinserting the new reproduced strings into the old population $\Omega(t)$ based on random reinserting approach to avoid pre-mature convergence. The evolutionary process continues until the pre-defined number of generations is reached.

If the best string can fulfill the accuracy specified by the medical doctors, the genetic algorithm stops. Otherwise, the number of rules represented by the strings is incremented by 1, and the genetic algorithm is restarted with a new population strings in which the number of rules represented is incremented. The process continues until the accuracy of a set of rules can fulfill the accuracy specified by the medical doctors.

It is necessary to mention that an enough amount of data generated by the neural network based classification unit is required for the rule discovery unit to create rules for representing the characteristics of the neural network based classification unit. For

example, Figure 4 shows the characteristic of the neural network based classification unit, and ‘o’s on the figure are used for representing the data generated by it to create the rule. If this small amount of data is used for creating the rule by the rule discovery unit, the misleading rule as illustration on Figure 4, which can only cover some of the characteristics of the neural network classification unit, is likely to be created. The rule created is under-fit the characteristic of the neural network based classification unit. Therefore large error between the characteristics of neural network based classification and the rule created by rule discovery unit is produced. As shown in Figure 5, a satisfactory amount of data is required to be generated by the neural network based classification unit in order to provide for the rule discovery unit to create rule that fits the characteristic of the neural network based classification. Small error between the characteristics of neural network based classification and the rule created by rule discovery unit is produced. To ensure the quality of the rules generated by the rule discovery unit is high enough, another set of real data is used for the validation. The rules created are considered to be satisfactory if the validation is satisfactory. If unsatisfactory validation is found on the rule set, the performance of the rule set could be improved by creating with larger amount of data generated by the neural network based classification unit.

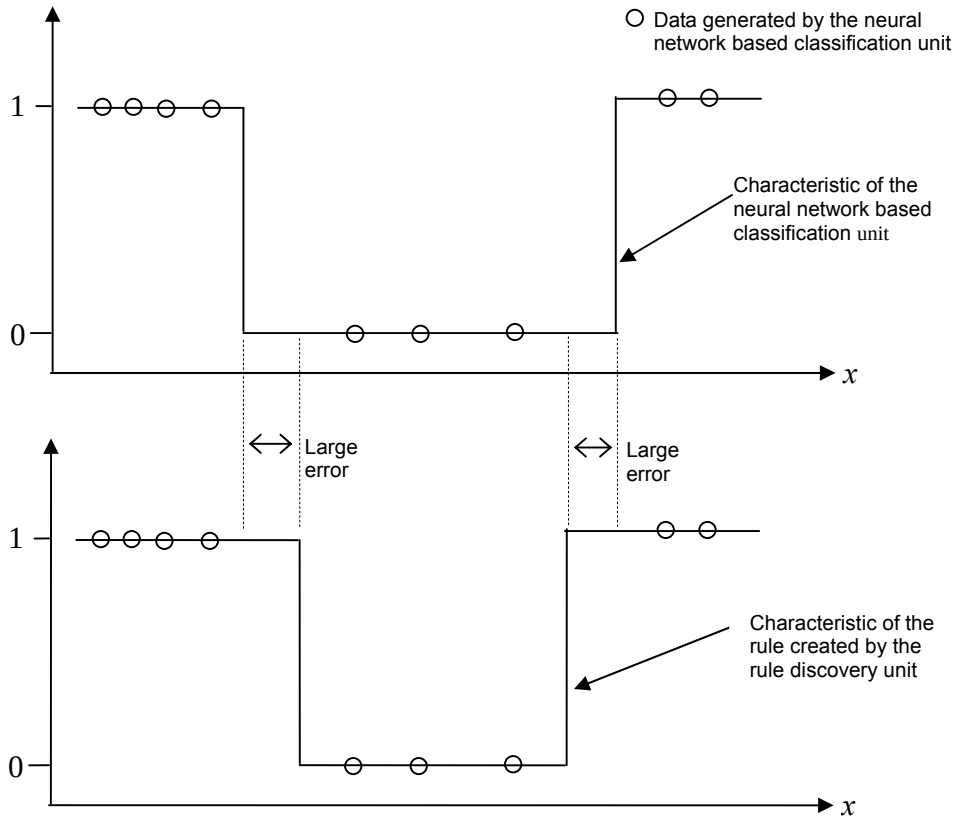

Figure 4 rule created based on a small amount of data generated by the neural network classification unit

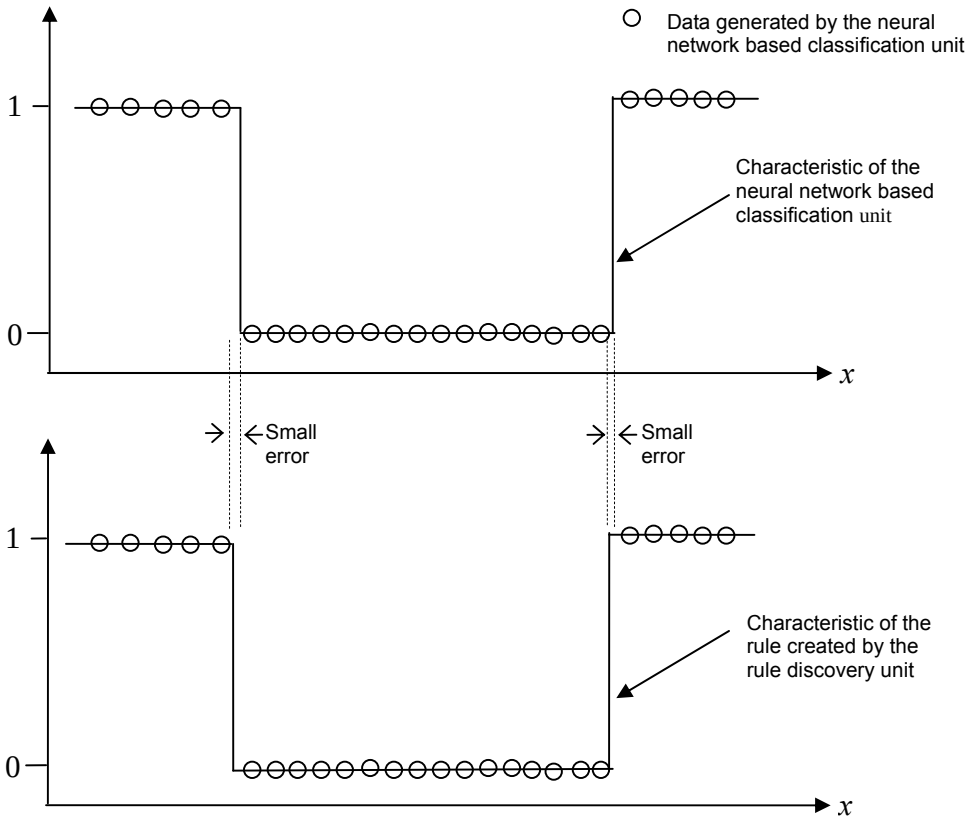

Figure 5 rule created based on a satisfactory amount of data generated by the neural network classification unit

3. Results and Discussion

3.1 T1DM's data

The data is collected from 16 T1DM patients at 14.6±1.5 years of age who volunteered to carry out the 10-hour overnight hypoglycemia study at the Princess Margaret Hospital for Children in Perth, Western Australia, Australia. Each T1DM patient was monitored overnight for the natural occurrence of nocturnal hypoglycemia. We measure the required physiological parameters, while the actual blood glucose levels were measured by a Yellow Spring Instrument. The actual blood glucose profiles for 16 T1DM children are shown in Figure 6.

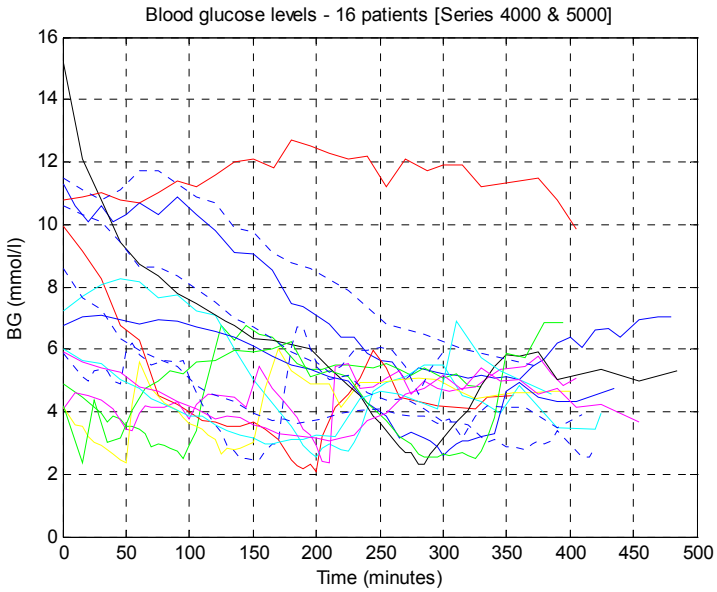

Figure 6 Actual blood glucose level profiles in the 16 T1DM patients

The responses of the 16 T1DM patients exhibited significant changes during the hypoglycemia phase against the non-hypoglycemia phase. The sampling period is around 5 minutes and 35-40 data was collected from each patient. 320 training data sets regarding to the blood glucose level y, rate of change of heart rate x_1, QT interval x_2 and rate of change of QT interval x_3 are used for developing the proposed neural network based rule discovery system, and 100 testing data sets were used for validating the developed proposed neural network based rule discovery system.

3.2 Classification

The following parameter settings suggested by (Schaffer et al. 1989) were implemented on the genetic algorithm to train the neural network based classification unit discussed in Section 2.2:

Crossover rate = 0.8;

Mutation rate $=1/(5n_h+1)$, where $5n_h+1$ is the number of variables of a string

and n_h is the number of hidden nodes on the neural network;

Number of generations = 1000;

Population size = 500.

The work described above was implemented using Matlab programming software. As a comparison, the neural network based classification unit has also been trained with a Levenberg Marquardt (LM) algorithm on the Matlab toolbox, which is a popular training algorithm for feed-forward neural networks (Hagan & Menhaj 1994, Lera & Pinzolas 2002). Table 3 shows the training errors of the trained neural networks corresponding to the six different neural network architectures which are 3-5-1, 3-8-1, 3-10-1, 3-12-1, and 3-15-1. As the genetic algorithm is a stochastic algorithm, different neural networks could be found with different runs. Also the neural networks searched by the Levenbery Marquardt algorithm are different with different initial searching points set in the algorithm. Therefore both the genetic algorithm and the Levenbery Marquardt algorithms were run for 30 times, and their average results among the 30 runs were recorded in Table 2. It can be found from Table 2 that the neural network with 10 hidden nodes is the best among the ones found by both the genetic algorithm and the Levenbery Marquardt algorithm, which can achieve the largest fitness. The sensitivity and the specificity of the optimized neural network architecture 3-10-1 with the best mean fitness found by the genetic algorithm are 0.7930 and 0.6053 respectively which is considered to be a satisfactory result in which the sensitivity is larger than 0.75 and the specificity is larger than 0.5. However, the ones found by the Levenbery Marquardt algorithm are 0.1614 and 0.9913 which is not satisfactory, as the sensitivity is smaller than 0.75 and the specificity is smaller than 0.5.

Table 2 Neural network configuration and training error produced by the genetic algorithm and the Levenberg Marquardt algorithm

Configuration of the neural networks	Mean sensitivity		Mean specificity		Mean fitness	
	NN-GA	NN-LM	NN-GA	NN-LM	NN-GA	NN-LM
3-5-1	0.6837	0.1111	0.6063	0.9948	0.6450	0.6032
3-8-1	0.6940	0.1466	0.6016	0.9917	0.6478	0.5692
3-10-1	0.7930	0.1614	0.6053	0.9913	0.6991	0.5763
3-12-1	0.8178	0.1450	0.5654	0.9846	0.6916	0.5298
3-15-1	0.8183	0.1984	0.5605	0.9834	0.6894	0.6113

Apart from using the neural network approaches, four commonly used approaches for generating explicit models are used:

a) statistical regression (SR) (Seber 2003), which is a common modeling approach to develop empirical models in linear polynomial forms.

b) fuzzy regression (FR) (Tanaka & Watada 1998), which can generate empirical models in fuzzy linear polynomial forms and the fuzziness of experimental data can be addressed.

c) genetic programming (GP) (Koza 1994), which can generate empirical models in nonlinear polynomial forms. In the GP, the structures of the polynomials are generated by the evolutionary operations, and the coefficients of the polynomials are determined by the least square method. The following parameters with reference to (Madar et al. 2005) were used in the GP: population size =100; probability of crossover = 0.5; probability of mutation= 0.5.

d) genetic programming based fuzzy regression (GP-FR) (Chan et al. 2009), which can generate empirical models in fuzzy nonlinear polynomial forms and can address fuzziness of experimental data. In the GP-FR, the structures of the

fuzzy polynomials are generated by the evolutionary operations, and the fuzzy coefficients of the polynomials are determined by the fuzzy regression method. The parameters used in the GP-FR were identified to the ones used in the GP.

The four algorithms were implemented with Matlab programming software. As both GP and GP-FR are stochastic algorithms, different results can be found with different runs. Therefore both GP and GP-FR were run for 30 times, and their average results among the 30 runs were recorded. The mean sensitivities, the mean specificities, the maximum sensitivities among the 30 runs and the maximum specificities among the 30 runs of the six algorithms, SR, FR, GP, GP-FR, NN-GA (with the neural network configuration 3-10-1) and NN-LM (with the neural network configuration 3-10-1) among the 30 runs are summarized in Table 3.

It can be found from Table 3 that the four approaches, SR, FR, GP and GP-FR achieved good specificities but poor sensitivities. NN-BP is better than the four approaches. Only the proposed NN-GA can achieve satisfactory results in both sensitivities and specificities in which the sensitivities are larger than 0.75 and the specificities are larger than 0.5. Also validation tests were carried out to evaluate the six approaches based on a set of testing data. Table 4 shows that the NN-GA can achieve satisfactory validation results in which the mean sensitivity is larger than 0.75 and the mean specificity is larger than 0.5, while the other five methods cannot achieve satisfactory results. The overall performance regarding to the maximum sensitive and the maximum specificity of the NN-GA are also better than the other five methods.

Table 3 Training results of the six methods, LR, FR, GP, FR-GP, NN-BP and NN-GA

	Mean sensitivity and	Maximum sensitive	Maximum specificity

	specificity					
	Sensitivity	Specificity	Sensitivity	Specificity	Sensitivity	Specificity
SR	0	1	0	1	0	1
FR	0	1	0	1	0	1
GP	0.0990	0.9978	0.1005	1.0398	0.1322	1.1537
GP-FR	0.0833	1.0990	0.1377	1.0774	0.1006	1.0177
NN-LM (10)	0.1984	0.9834	0.4921	0.9922	0.2381	1.0000
NN-GA (10)	0.7930	0.6053	0.9048	0.5175	0.7302	0.7549

Table 4 Validation results of the six methods, LR, FR, GP, FR-GP, NN-BP and NN-GA

	Mean sensitivity and specificity		Maximum sensitive		Maximum specificity	
	Sensitivity	Specificity	Sensitivity	Specificity	Sensitivity	Specificity
SR	0	1	0	1	0	1
FR	0	1	0	1	0	1
GP	0.0893	0.8996	0.0275	0.9374	0.0290	1.0401
GP-FR	0.0744	0.9816	0.0337	0.9623	0.0005	0.9090
NN- LM (10)	0.1443	0.7646	0.3826	0.7715	0.1851	0.7775
NN–GA (10)	0.7557	0.5768	0.8623	0.4932	0.6959	0.7194

The results of the four approaches, SR, FR, GP and FR-GP are shown in Figures 4 and 5 in which the patients' actual measured glucose levels and their estimated glucose levels based on the four approaches are shown. Figure 7 shows the results of SR and GP in which the least square method is used in both methods. It shows that SR intended to fit in the middle of the glucose levels of all patients' data. Even GP is slightly better than SR, but the results are far not satisfactory in diagnosis of hypoglycemia. Figure 8 shows the similar results of FR and FR-GP in which fuzzy regression are involved in both methods. It shows that both FR and FR-GP provided the estimates of glucose levels of all patients in the middle of all actual data. Both FR and FR-GP are unsatisfactory in diagnosis of hypoglycemia. The results of NN-LM and NN-GA are shown in Figure 6 in which the determinations of hypoglycemia of patients are shown. Comparing with NN-

LM and the other four approaches, Figure 9 shows clearly that NN-GA can achieve much more satisfactory results in which much more patients can be diagnosed correctly if they are hypoglycemia.

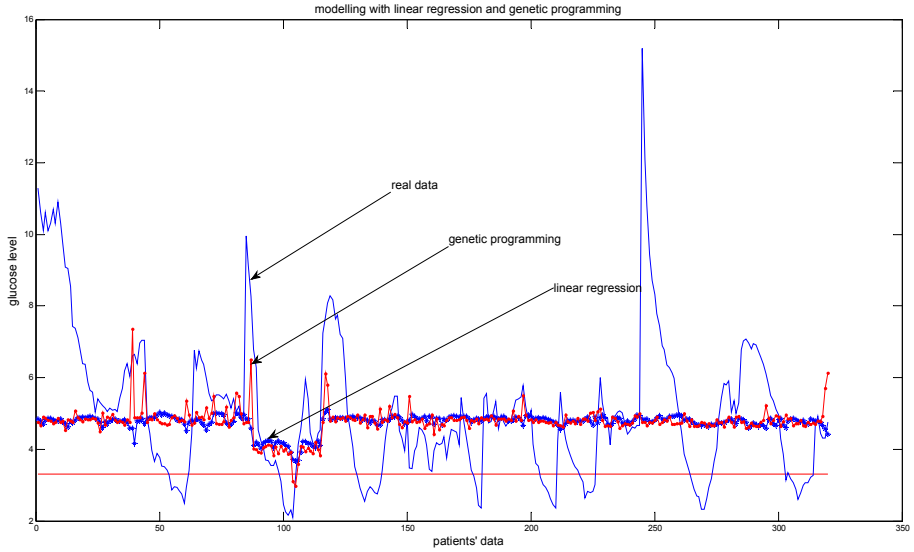

Figure 7 Results achieved by SR and GP

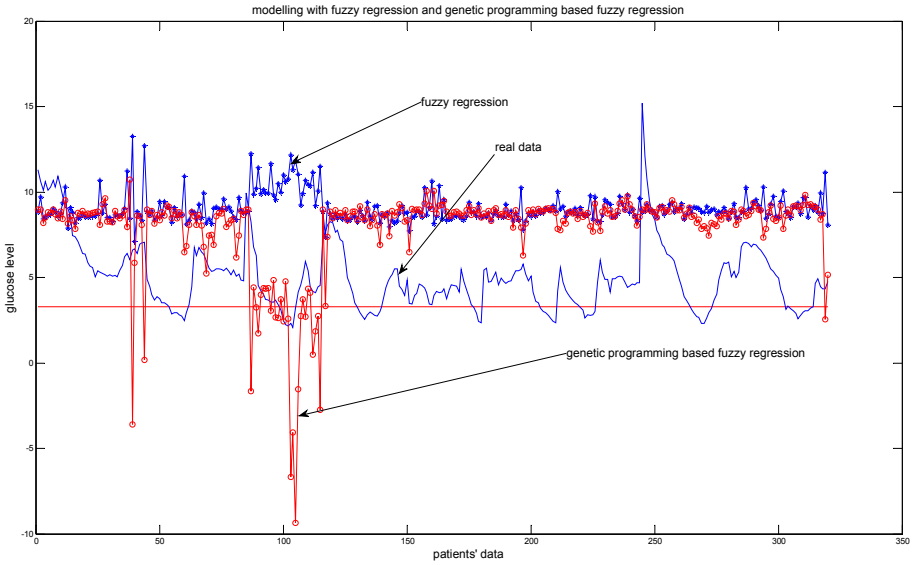

Figure 8 Results achieved by FR and FR-GP

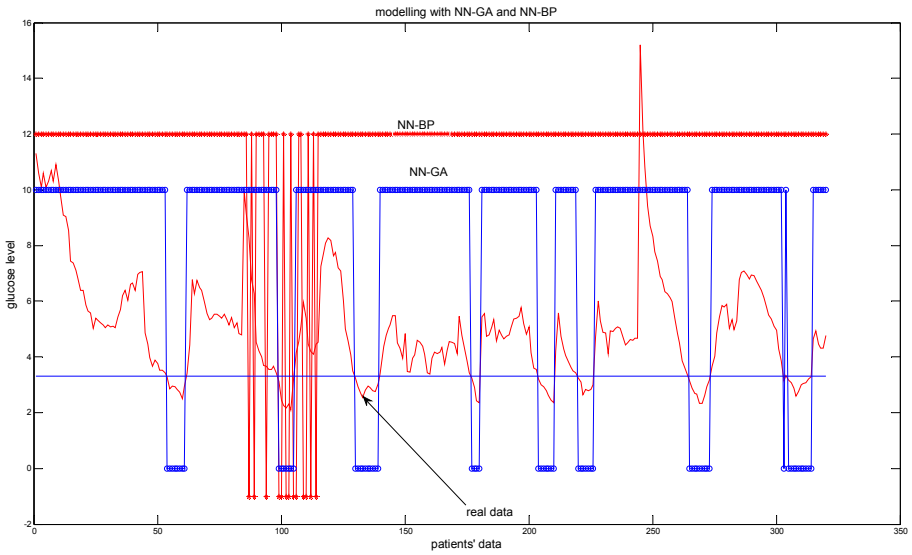

Figure 9 Results achieved by NN-GA and NN-LM

3.3 Rule extraction

The rule discovery system was implemented using Matlab programming software. The parameters setting, crossover rate =0.8, mutation rate = $1/(6 \times N_R)$, where the number of variables of the string is $6 \times N_R$, and N_R is the number of rules extracted by the rule discovery system. Based on the 500 data sets regarding to the three physiological parameters and the hypoglycemia generated by the neural network based classification unit, the genetic algorithm has been run for 30 times. The best rule set among the 30 runs were extracted as follows:

If 1.0021 <x1<= 1.2534 & -0.6504 <x2<= 0.3041 & 0.0362 <x3<= 0.1112 => 0
Or If 1.0370 <x1<= 1.0826 & -0.1322 <x2<= 0.3351 & -0.1066 <x3<= 0.0244 => 0
Or If 0.9416 <x1<=1.0768 & -0.0727 <x2<= 0.2059 & -0.0083 <x3<= -0.0818 => 0
Or If 1.0153 <x1<= 1.1753 & -0.1374 <x2<= 0.5327 & 0.0390 <x3<= 0.0740 => 0
Or If 1.0740 <x1<= 1.1627 & 0.0717<x2<= 0.2226 & -0.0219 <x3<= -0.2675 => 0
Or If 0.9846 <x1<= 1.0341 & -0.5341 <x2<= -0.7819 & 0.0164 <x3<= 0.0546 => 0
Or If 1.0679 <x1<= 1.1259 & -0.5876 <x2<= 0.3072 & -0.1534 <x3<= 0.0051 => 0
Or If 1.0807 <x1<= 1.0862 & -0.0942 <x2<= -0.1308 & -0.1377 <x3<= 0.0360 => 0
Or If 1.0226 <x1<= 1.1891 & -0.2558 <x2<= 0.2251 & -0.1860 <x3<= 0.0138 => 0
Or If 1.1694 <x1<= 1.1731 & -0.1051 <x2<= -0.0349 &-0.0468 <x3<= -0.0085 => 0
Or If 1.1229 <x1<= 1.2217 & 0.0821 <x2<= 0.2235 & -0.0714 <x3<= 0.1527 => 0
Or If 1.1555 <x1<= 1.1565 & 0.072 <x2<= 0.0770 & -0.1939 <x3<= 0.0817 => 0
Or If 0.9406 <x1<= 1.3289 & -0.0839 <x2<= 0.2479 & -0.0752 <x3<= 0.0763 => 0

The sensitivity and the specificity for the rule set were found as 0.7047 and 0.5427 respectively with respect to the data generated by the neural network based classification unit which is considered to be satisfactory. Based on the 100 testing data which were the actual measured data of the T1DM patients, validation of the rule set can be performed. It was found that the sensitivity and the specificity are 0.7911 and 0.5201 respectively which is larger than the specified ones 0.75 and 0.5. Therefore the rule set created by the rule discovery system is considered to be satisfactory. Based on the rule set, information can be extracted from the neural network.

Conclusion

In this paper, a neural network based rule discovery system is developed to determine the presence of hypoglycemic episodes based on the TIDM patients' physiological parameters, rate of change of heart rate, corrected QT interval of electrocardiogram signal and rate of change of corrected QT interval. It was developed based on T1DM patients'

420 data sets which were collected from 16 T1DM patients by using the genetic algorithm. 320 data sets were used to develop the neural network based rule discovery system and 100 data sets were used to validate its performance. It was found that the sensitivity and specificity were found as 79.30% and 60.53% respectively which are considered to be reasonable and are better than the ones found by the commonly used methods, statistical regression, genetic programming and fuzzy regression. Explicit rules can be extracted from the neural network unit based on the proposed genetic algorithm. It is found that the extracted rules can achieve sensitivity with 79.11% and specificity with 52.01% which are considered to be satisfactory. The neural network based rule discovery system compensates the limitation of the traditional neural network approaches that no implicit information can be generated.

References

Aruna P., Puviarasan N., & Palaniappan B. (2007). Diagnosis of gastrointestinal disorders using DIAGNET. *Expert Systems with Applications*, 32, 329-335, 2007.

Carvalho D.R., & Freitas A.A. (2000). A hybrid decision tree/genetic algorithm for coping with the problem of small disjoints in data mining. *Proceedings of Conference of Genetic and Evolutionary Computation*, 1061-1068, 2000.

Chan K.Y., Kwong C.K., & Tsim Y.C. (2009). A genetic programming based fuzzy regression approach to modeling manufacturing processes. *International Journal of Production Research*, 2009.

Chang C.L., & Chen C.H. (2009). Applying decision tree and neural network to increase quality of dermatologic diagnosis. *Expert Systems with Applications*, 36, 4035-4041, 2009.

Chang C.L., & Hsu M.Y. (2009). The study that applies artificial intelligence and logistic regression for assistance in differential diagnostic of pancreatic cancer. *Expert Systems with Applications*, 36, 10663-10672, 2009.

Cho B.H., Yu H., Kim K.W., Kim T.H., Kim I.Y. &. Kim S.I. (2008). Application of

irregular and unbalanced data to predict diabetic nephropathy using visualization and feature selection methods. *Artificial Intelligence in Medicine*, 42(1), 37-53.

Chu A., Ahn H., Halwan B., Kalmin B., Artifon E.L.V., Barkun A., Lagoudakis M.G. & Kumar A. (2008). A decision support system to facilitate management of patients with acute gastrointestinal bleeding. *Artificial Intelligence in Medicine*, 42(3), 247-259.

Das R., Turkoglu I. & Sengur A. (2009). Effective diagnosis of heart disease through neural networks ensembles. *Expert Systems with Applications*, 36, 7675-7680.

DCCT Research Group. (1993). The effect of intensive treatment of diabetes on the development and progression of long-term complications in IDDM. *The New England Journal of Medicine*, 329, 977-986.

DCCT Research Group. (1995). Adverse events and their association with treatment regimens in the Diabetes Control and Complications Trial. *Diabetes Care*, 18, 1415-1427.

Gil D., Johnson M., Chamizo J.M.G.., Paya A.S. & Fernandez D.R. (2009). Application of artificial neural networks in diagnosis of urological dysfunctions. *Expert Systems with Applications*, 36, 5754-5760.

Gray H.F., Maxwell R.J., Martinez-Perez I., Arus C., & Cerdan S. (1998). Genetic programming for classification and feature selection: analysis of ^{1}H nuclear magnetic resonance spectra from human brain tumour biopsies. *NMR in Biomedicine*, 11, 217-224.

Hand D.J., (2001). Principles of Data Mining, MIT Press.

Hagan M.T., & Menhaj M., (1994). Training feedforward networks with the Marquardt algorithm. *IEEE Transactions on Neural Networks*, 5(6), 989-993.

Harris N.D., Baykouchev S.B., Marques L.B., Cochrane T., George E., Heller S.R., & Ward J.D., (1996). A portable system for monitoring physiological responses to hypoglycaemia. *Journal of Medical Engineering and Technology*, 20(6), 196-202.

Koza J. (1994). Genetic Programming II: automatic discovery of reusable programs, MIT Press.

Keles A., Hasiloglu S.A., Ali K., & Aksoy Y., (2007). Neuro-fuzzy classification of prostate cancer using NEFCLASS-J. *Computers in Biology and Medicine*, 37(11), 1617-1628.

Kannathal N., Lim C.M., Acharya R., & Sadasivan P.K., (2006). Cardiac state diagnosis using adaptive neuro-fuzy technique. *Medical Engineering and Physics*, 28, 809-805.

Lera G., & Pinzolas M., (2002). *Neighborhood based Levenberg-Marquardt algorithm for neural network training*. *IEEE Transactions on Neural Network*, 13(5), 1200-1203.

Lin W.B., (2008). The effect of knowledge sharing model. *Expert Systems with Application*s, 34, 1508-1521.

Madar J., Abonyi J., & Szeifert F., (2005). Genetic programming for the identification of nonlinear input – output models. *Industrial and Engineering Chemistry Research*, 44, 3178 – 3186.

Mantzaris D., Anastassopoulos G..A. & Gardikis S., (2008). A non-symbolic implementation of abdominal pain estimation in childhood. *Information Sciences*, 178, 3860-3866.

Muhlenbein H. & Voosen D.S. (1993). Predictive models for the breeder genetic algorithm: I. Continuous parameter optimization. *Evolutionary Computation*, 1(1), 25-49.

Oentaryo R.J., Pasquier M., & Quek C. (2008). GenSoFNN-Yager: A novel brain-inspired generic self-organizing neuro-fuzzy system realizing Yager inference. *Expert Systems with Application*, 35, 1825-1840.

Pickup J.C. (2000). Sensitivity glucose sensing in diabetes. *Lancet*, 355, 426-427.

Reggia J.A., & Sutton S.S. (1988). Self-processing networks and their biomedical implications. *Proceedings of the IEEE*, 76(6), 580-592.

Schaffer J., Caruana R., Eshelman L., & Das R. (1989). A study of control parameters affecting online performance of genetic algorithms for function optimization. *Proceedings of the 3rd International Conference on Genetic Algorithms*, 51-60.

Seber G..A.F. (2003). Linear regression analysis, Wiley.

Sim J., Tung W.L., & Quek C. (2006). FCMAC-Yager: a novel yager-inference-scheme-based fuzzy CMAC. *IEEE Transactions on Neural Networks*, 17(16), 1394-1410.

Smith S.L., Gaughan P., Halliday D.M., Ju Q., Aly N.M. & Playfer J.R. (2007). Diagnosis of Parkinson's disease using evolutionary algorithms. *Genetic Programming and Evolvable Machines*, 8, 433-447.

Subashini T.S., Ramalingam V., & Palanivel S. (2009). Breast mass classification based on cytological patterms using RBFNN and SVM. *Expert Systems with Applications*, 3, 5284-5290.

Tanaka H. & Watada J. (1998). Possibilistic linear systems and their application to the linear regression model. *Fuzzy Sets and Systems*, 272, 275-289.

Winkler S.M., Affenzeller M. & Wagner S. (2009). Using enhanced genetic programming techniques for evolving classifiers in the context of medical diagnosis. *Genetic Programming and Evolvable Machines,* 10, 111-140.

Yale J.F. (2004). Nocturnal hypoglycemia in patients with insulin-treated diabetes. Diabetes Research and Clinical Practice, 65, 41-46.