Chapter from the book *Fuzzy Inference System - Theory and Applications*
Downloaded from: http://www.intechopen.com/books/fuzzy-inference-system-theory-and-applications

Interested in publishing with IntechOpen?
Contact us at book.department@intechopen.com
A Concise Fuzzy Rule Base to Reason Student Performance Based on Rough-Fuzzy Approach

Norazah Yusof, Nor Bahiah Ahmad, Mohd. Shahizan Othman and Yeap Chun Nyen
Faculty of Computer Science and Information System, Universiti Teknologi Malaysia, Skudai, Johor
Malaysia

1. Introduction

A fuzzy inference system employing fuzzy if then rules able to model the qualitative aspects of human expertise and reasoning processes without employing precise quantitative analyses. This is due to the fact that the problem in acquiring knowledge from human experts is that much of the information is uncertain, inconsistent, vague and incomplete (Khoo and Zhai, 2001; Tsaganou et al., 2002; San Pedro and Burstein, 2003; Yang et al., 2005). The drawbacks of FIS are that a lot of trial and error effort need to be taken into account in order to define the best fitted membership functions (Taylan and Karagözoglu, 2009) and no standard methods exist for transforming human knowledge or experience into the rule base (Jang, 1993).

Evaluation and reasoning of student’s learning achievement is the process of determining the performance levels of individual students in relation to educational objectives (Saleh and Kim, 2009). Although Fuzzy inference system is a potential technique to reason the student’s performance, as well as to present his/her knowledge status (Nedic et al., 2002; Xu et al., 2002; Kosba et al. 2003), it is a challenge when more than one factor involve in determining the student’s performance or knowledge status (Yusof et. al, 2009). Hence, the reasoning of the student’s performance for multiple factors is difficult. This issue is critical considering that the human experts’ knowledge is insufficient to analyze all possible conditions as the information gained is always incomplete, inconsistent, and vague.

Addressing these matters, this work proposes a Neuro-Fuzzy Inference System (ANFIS), which combines fuzzy inference system and neural network, in order to produce a complete fuzzy rule base system (Jang, 1993). The fuzzy system represents knowledge in an interpretable manner, while the neural networks have the learning ability platform to optimize its parameters. Hence, ANFIS has the capability to perform parameter-learning rather than structural learning (Lin and Lu, 1996). ANFIS is expected to recognize other decisions that are previously not complete, in both the antecedents and consequent parts of the fuzzy rules. Unfortunately, too many fuzzy rules will result in a large computation time and space (Jamshidi, 2001). Therefore, reduction of knowledge is possible to be applied to determine the selection of important attributes that can be used to represent the decision system (Chen, 1999) based on the theory of rough sets. Fig. 1 shows the proposed fuzzy inference system.
This chapter is divided into six sections. Section 1 is the introduction and the problem statements. Section 2 discusses about the student modeling and learning criteria. Section 3 presents the Human Expert Fuzzy Inference System model that defines the data representation and the rule base acquired from the human experts. Section 4 describes the ANFIS approach to form a complete fuzzy rule base to solve the problem of incomplete and vague decisions made by human. Section 5 presents the proposed Rough-Fuzzy approach to determine important attributes and refine the fuzzy rule base into a concise fuzzy rule base. Finally, section 6 presents the conclusions of the work.

2. Student modeling and the learning criteria

Student model represents the knowledge about the student’s behavior and learning performance. In this work, student’s performance are classified into three categories, named as Has Mastered (HM), Moderately Mastered (MM), and Not Mastered (NM). The conditions that determine the decision made about the student’s performance is also depend on the criteria set by the human expert. There are four input conditions namely, the score \( S \), time \( T \), attempts \( A \), and helps \( H \) in which each of the input condition is represented by three term sets with values (Norazah, 2005).

a. Score \( S \) is the average scoring, \( x_1 \), which gains from each question of a learning unit and the term sets is represented by low \( (S_1) \), moderate \( (S_2) \), and high \( (S_3) \). It can be found by dividing the total marks for a set of given questions by the total number of questions \( Q \) in the set, as shown in equation (1).

\[
x_1 = \frac{\sum_{i=1}^{Q} m_i}{Q}
\]
Where:

- $m_i$ is marks from each question
- $Q$ is total number the question in the set
- $Q$ is total number of questions
- $T_i$ is the time spent to answer the $i$-th question
- $\bar{X}_i$ is mean score for the time spent distribution
- $\sigma_i$ is the standard deviation for the $i$-th question

b. **Time (T)** is the average duration, $x_2$, taken by a student to answer the each question of a learning unit and with three term sets: *fast* ($T_1$), *average* ($T_2$), and *slow* ($T_3$). The average of time ($x_2$) is obtained by dividing the total time to answer a set of given questions by the total number of questions, see equation (2).

\[
x_2 = \frac{\sum_{i=1}^{Q} T_i}{Q}
\]  (2)

Where:
- $Q$ is total number of questions
- $T_i$ is the time spent to answer the $i$-th question

Measurement of time can be done by using the distribution method. Fig. 2 shows the T-score distribution, in which the mean is 50 and the standard deviation is 10.

![T-score distribution for time taken to answer question](image)

The time taken to answer each question ($T_i$) can be calculated by using the equation (3).

\[
T_i = \frac{10(X_i - \bar{X}_i)}{\sigma_i} + 50
\]  (3)

Where:
- $T_i$ is the time spent to answer the $i$-th question
- $X_i$ is the time spent by the student
- $\bar{X}_i$ is mean score for the time spent distribution
- $\sigma_i$ is the standard deviation for the $i$-th question

The numbered "10" is distance value of standard deviation from mean, while numbered "50" is value of mean. T-score is divided by 100 so that able to get the value in the range 0 to 1.

c. **Attempt (A)** is the average number of tries, $x_3$, for a given learning unit, in which it is counted after student give a wrong answer for the first attempt and the question will repeat again for student to answer until correct. The term sets involve: *a few* ($A_1$), *average* ($A_2$), and *many* ($A_3$). The average of attempt ($x_3$) is calculated as equation (4). Dividing the total number of tries to answer a set of given questions by the total number of questions in the set.

\[
x_3 = \frac{\sum_{i=1}^{Q} x_3}{Q}
\]
\( x_3 = \frac{\sum_{i=1}^{Q} t_i}{Q} \)  \hspace{1cm} (4)

Where:

\( Q \) is total number of questions

The number of attempt \( (t_i) \) is determined by calculating the number of attempts made \( (a_i) \) to answer a given question and dividing it by the maximum number of attempts \( (P_i) \) allowed for the question.

\( t_i = \frac{a_i}{P_i} \)  \hspace{1cm} (5)

Where:

\( a_i \) is the number of attempts made to answer a given question

\( P_i \) is the maximum number of attempts allowed for the question
d. Help \( (H) \) is the average amount of help, \( x_4 \), of a learning unit where it able to help student by giving some hints or notes to answer the question. The term sets involve: little \( (H_1) \), average \( (H_2) \), and needed \( (H_3) \).

The average amount of help \( (x_4) \) is calculated as equation (6), by dividing the total amount of help accessed by a student in answering a set of given questions by the total number of questions in the set.

\( x_4 = \frac{\sum_{i=1}^{Q} h_i}{Q} \) \hspace{1cm} (6)

Where:

\( Q \) is the total number of questions

\( h_i \) is the total amount of help accessed by a student

The amount of help \( (h_i) \) is found by calculating the number of help \( (l_i) \) links that a student accessed while answering a given question and dividing it by the maximum number of help links \( (H_{max}) \) provided for a given question.

\( h_i = \frac{l_i}{H_{max}} \) \hspace{1cm} (7)

The output consequent of the student model is the student’s performance and can be represented as has mastered \( (P_1) \), moderately mastered \( (P_2) \) and not mastered \( (P_3) \) for the output. A student is classified as has mastered when the student earns high scores (i.e. greater than 75%) with below 40% of time spent, not exceeding 25% of number of tries needed and number of helps. Besides that, a student is classified as moderately mastered when the student earns a moderate score, with moderate time spent, tries more than once, and number of help needed. For example, a moderate score would be rated in between 35% and 75%, time spent between 40% and 60%, tries between 25% and 75%, and help between 25% and 75%. Furthermore, a student is classified as not mastered when the student has a low score with a lot of time, many tries and many help needed. However, in acquiring knowledge from the human experts is that, they cannot decide on all
possible students learning performance. Bases on a survey done by Norazah (2005), there are only 18 decisions about the student’s behavior are formed with certainty from seven subject matter experts; and these decisions are considered as the acceptable rules. All other decisions that are not certain and have conflicts are being discarded from the rules.

<table>
<thead>
<tr>
<th>Criteria item</th>
<th>Has Mastered $y &gt; 75$</th>
<th>Moderately Mastered $75 \geq y \geq 25$</th>
<th>Not Mastered $y &lt; 25$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scores (S)</td>
<td>$x_1 \geq 75%$ High</td>
<td>$75% \geq x_1 \geq 35%$ Md</td>
<td>$x_1 &lt; 35%$ Low</td>
</tr>
<tr>
<td>Time (T)</td>
<td>$x_2 &lt; 40$ Fast</td>
<td>$60 \geq x_2 \geq 40$ Avg</td>
<td>$x_2 &gt; 60$ Slow</td>
</tr>
<tr>
<td>Attempt (A)</td>
<td>$x_3 &lt; 25%$ A Few</td>
<td>$75% \geq x_3 \geq 25%$ Avg</td>
<td>$x_3 &gt; 75%$ Many</td>
</tr>
<tr>
<td>Help (H)</td>
<td>$x_4 &lt; 25%$ Little</td>
<td>$75% \geq x_4 \geq 25%$ Avg</td>
<td>$x_4 &gt; 75%$ Needed</td>
</tr>
</tbody>
</table>

Table 1. The criteria for the student’s performance

3. Human expert Fuzzy Inference System

Human expert’s FIS uses a collection of fuzzy membership functions and rules to reason about student’s performance. FIS consists of a fuzzification interface, a rule base, a database, a decision-making unit, and finally a defuzzification interface.

To compute the output of this fuzzy inference system given the inputs, four steps has to be followed (Norazah, 2005):

a. Compare the input variables with the membership functions on the antecedent part to obtain the membership values of each linguistic label. This step is called fuzzification.

b. Combine the membership values on the premise part to get firing strength of each rule.

c. Generate the qualified consequents or each rule depending on the firing strength.

d. Aggregate the qualified consequents to produce a crisp output. This step is called defuzzification.

3.1 Fuzzification

In the fuzzification stage, the input and output of the fuzzy inference system are determined. Table 2 and Table 3 exhibit examples of the four input and one output
variables, in which each of the variables consists of three term values and labels as discussed in Section 2. The fuzzy output follows the zero-order Sugeno style inference, in which the output value of each fuzzy rule is a constant (Sivanandam et al., 2007). Fig. 3 shows the four inputs and one single output for the Human Expert FIS.

<table>
<thead>
<tr>
<th>Fuzzy output variable</th>
<th>Fuzzy linguistic terms</th>
<th>Numerical range(normalized)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Performance (P)</td>
<td>{Not Mastered, Moderately Mastered, Has Mastered}</td>
<td>0.0 0.5 1.0</td>
</tr>
</tbody>
</table>

Table 3. The output variables of the Fuzzy Inference System

Fig. 3. Four inputs and single output for the Human Expert FIS

The membership function of the input is expressed by a Gaussian function specified by two parameters \( \{\sigma, c\} \), and the membership value is derived by the formula in Fig. 4.

\[
\text{gaussian}(x; \sigma, c) = \exp \left( -\left[ \frac{x - c}{2\sigma} \right]^2 \right)
\]

(8)

Where:
- \( c \) represents the membership function’s center
- \( \sigma \) determines the membership function’s width
3.2 Creating fuzzy rules

Fuzzy rules are a collection of linguistic statements that describe how the fuzzy inference system should make a decision regarding classifying an input or controlling an output. Fig. 5 presents the four inputs and one output reasoning of the student’s performance procedure for zero order Sugeno fuzzy model. Each input has its own membership function.

Fig. 5. Fuzzy reasoning procedures for Human Expert FIS model of Student’s Performance

$R_i$ have four input variables and one output variable as shown below:

$R_i$: IF $S$ is $\mu_{i1}$ AND $T$ is $\mu_{i2}$ AND $A$ is $\mu_{i3}$ AND $H$ is $\mu_{i4}$ THEN $P$ is $w_i$

The rule $R_i$ is the $i$-th rule in the fuzzy rule base system, the $\mu_i$ is the membership function of the antecedent part of the $i$-th rule for each input variable and $w_i$ is the weight of the consequent of each rule. For example, for input 1 is score and the membership function can classified as low, moderate or high. If score is high and time is fast and attempt is a few and help is little then student performance is has mastered. This process of taking input such as score and processing it through membership functions to determine the “high” score is called fuzzification. Based on the human experts’ experience and knowledge about the students’ performance, 18 initial rules that are certain have been constructed as shown in Table 4.

3.3 Combining outputs into an output distribution

The outputs of all of the fuzzy rules must now be combined to obtain one fuzzy output distribution. The output membership functions on the right-hand side of the figure are combined using the fuzzy operator AND to obtain the output distribution shown on the lower right corner of the Fig. 5. For a zero-order Sugeno model, the output level $z$ is a constant. The output level $z_i$ of each rule is weighted by the firing strength $w_i$ of the rule (Lin and Lu, 1996). For example, for an $\cap$ rule with input 1 = $x$ and input 2 = $y$, the firing strength is as shown in equation (9).

$$w_i = F_1(x) \cap F_2(y)$$  \hspace{1cm} (9)
Where:
\( F_1 \) and \( F_2 \) are the membership functions for input 1 and 2, respectively

<table>
<thead>
<tr>
<th>Rule</th>
<th>S</th>
<th>T</th>
<th>A</th>
<th>H</th>
<th>P</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>High</td>
<td>Fast</td>
<td>A few</td>
<td>Little</td>
<td>Has Mastered</td>
</tr>
<tr>
<td>2</td>
<td>High</td>
<td>Fast</td>
<td>A few</td>
<td>Average</td>
<td>Moderately Mastered</td>
</tr>
<tr>
<td>3</td>
<td>High</td>
<td>Fast</td>
<td>A few</td>
<td>Needed</td>
<td>Moderately Mastered</td>
</tr>
<tr>
<td>4</td>
<td>High</td>
<td>Fast</td>
<td>Average</td>
<td>Little</td>
<td>Moderately Mastered</td>
</tr>
<tr>
<td>5</td>
<td>High</td>
<td>Fast</td>
<td>Average</td>
<td>Average</td>
<td>Moderately Mastered</td>
</tr>
<tr>
<td>6</td>
<td>High</td>
<td>Fast</td>
<td>Average</td>
<td>Needed</td>
<td>Moderately Mastered</td>
</tr>
<tr>
<td>7</td>
<td>High</td>
<td>Fast</td>
<td>Many</td>
<td>Little</td>
<td>Not Mastered</td>
</tr>
<tr>
<td>8</td>
<td>High</td>
<td>Fast</td>
<td>Many</td>
<td>Average</td>
<td>Not Mastered</td>
</tr>
<tr>
<td>9</td>
<td>High</td>
<td>Fast</td>
<td>Many</td>
<td>Needed</td>
<td>Not Mastered</td>
</tr>
<tr>
<td>10</td>
<td>High</td>
<td>Average</td>
<td>A few</td>
<td>Little</td>
<td>Has Mastered</td>
</tr>
<tr>
<td>11</td>
<td>High</td>
<td>Average</td>
<td>A few</td>
<td>Average</td>
<td>Moderately Mastered</td>
</tr>
<tr>
<td>12</td>
<td>High</td>
<td>Average</td>
<td>Many</td>
<td>Needed</td>
<td>Not Mastered</td>
</tr>
<tr>
<td>13</td>
<td>High</td>
<td>Slow</td>
<td>A few</td>
<td>Little</td>
<td>Has Mastered</td>
</tr>
<tr>
<td>14</td>
<td>High</td>
<td>Slow</td>
<td>Many</td>
<td>Needed</td>
<td>Not Mastered</td>
</tr>
<tr>
<td>15</td>
<td>Moderate</td>
<td>Fast</td>
<td>A few</td>
<td>Little</td>
<td>Moderately Mastered</td>
</tr>
<tr>
<td>16</td>
<td>Moderate</td>
<td>Average</td>
<td>Average</td>
<td>Average</td>
<td>Moderately Mastered</td>
</tr>
<tr>
<td>17</td>
<td>Moderate</td>
<td>Average</td>
<td>Many</td>
<td>Needed</td>
<td>Not Mastered</td>
</tr>
<tr>
<td>18</td>
<td>Low</td>
<td>x</td>
<td>x</td>
<td>x</td>
<td>Not Mastered</td>
</tr>
</tbody>
</table>

Table 4. Initial fuzzy rules determine by human experts

3.4 Defuzzification of output distribution

The input for the defuzzification process is a fuzzy set and the output is a single number crispness recovered from fuzziness. Given a fuzzy set that encompasses a range of output values, we need to return one number, thereby moving from a fuzzy set to a crisp output. The final output of the system is the weighted average of all rule outputs, computed as in equation (10).

\[
\text{Final Output} = \frac{\sum_{i=1}^{N} w_i z_i}{\sum_{i=1}^{N} w_i}
\] (10)

Finally, all the outputs of datasets for reasoning of the student’s performance in the human expert FIS have been recorded.

Next section describes the ANFIS approach to form a complete fuzzy rule base to solve the problem of incomplete and vague decisions made by human.


Basically, fuzzy rules and fuzzy reasoning are the backbone of fuzzy inference systems, which are the most important modeling tools based on fuzzy sets (Jang et al., 1997). Fuzzy reasoning is an inference procedure that derives conclusions from the set of fuzzy If-Then
rules and known facts. The ANFIS model is proposed to form a complete fuzzy rule bases so that all possible input conditions of the fuzzy rules are being generated.

It is necessary to take into consideration the scarcity of data and the style of input space partitions. For example, for a single input problem, usually 10 data points are necessary to come up with a good model (Jang et al., 1997). Details on ANFIS model structure will be described in section 4.1.

4.1 ANFIS model structure

The ANFIS model structure consists of four nodes for input layer, the nodes of hidden layer and one node for output layer as presented in Fig. 6. The input layer represents the antecedent part of the fuzzy rule, which is the student’s learning behavior such as the scores ($S$) earned, the time ($T$) spent, the attempts ($A$), and helps ($H$); the output layer represents the consequent part of the rule, i.e. the student’s performance ($P$). The size of the hidden layer is determined experimentally.

In this work, the ANFIS model is trained with 18 fuzzy rules obtained from the human expert. These rules are considered as the rules that are certain. After that, 81 potential fuzzy rules are used for testing the network that represent the $3 \times 3 \times 3 \times 3$ rule antecedents.

![Fig. 6. ANFIS model structure](www.intechopen.com)

From the Fig. 6, every nodes of the same layer have similar functions. Layer 1 is the input layer and the neurons in this layer simply pass external crisp signals to Layer 2.
\[ y_i^{(1)} = \mu_{S_i}(S) \]
\[ y_i^{(1)} = \mu_{T_i}(T) \]
\[ y_i^{(1)} = \mu_{A_i}(A) \]
\[ y_i^{(1)} = \mu_{H_i}(H) \]

(11)

Where:
\( \mu_{S_i}(S), \mu_{T_i}(T), \mu_{A_i}(A), \mu_{H_i}(H) \) are the input
\( S_i, T_i, A_i, H_i \) are the linguistic value
\( y_i^{(1)} \) is the output of input neuron \( i \) in Layer 1

Layer 2 is the fuzzification layer. Neurons in this layer perform fuzzification. In this student model, fuzzification neurons have a Gaussian function. A Gaussian function, which has a Gaussian shape, is specified as:

\[ y_i^{(2)} = e^{-\frac{(x_i^{(2)} - c_i)^2}{2\sigma^2}} \]

(12)

Where:
\( x_i^{(2)} \) is the input
\( y_i^{(2)} \) is the output of neuron \( i \) in Layer 2
\( c \) represents the membership function’s center
\( \sigma \) determines the membership function’s width

Layer 3 is the rule layer. Each neuron in this layer corresponds to a single to a single Sugeno type fuzzy rule. A rule neuron receives inputs from the respective fuzzification neurons and calculates the firing strength of the rule it represents. In an ANFIS, the conjunction of the rule antecedents is evaluated by the operator product (Negnevitsky, 2005). Each node output represents the firing strength of a rule. Thus, the output of neuron \( i \) in Layer 3 is obtained as,

\[ y_i^{(3)} = \bar{w}_i = \mu_{S_i}(S) \times \mu_{T_i}(T) \times \mu_{A_i}(A) \times \mu_{H_i}(H) \quad i = 1,2 \]

(13)

Layer 4 is the normalization layer. Each neuron in this layer receives inputs from all neurons in the rule layer and calculates the normalized firing strength of a given rule. The normalized firing strength is the ratio of the firing strength of a given rule to the sum of firing strengths of all rules. It represents the contribution of a given rule to the final result.

\[ y_i^{(4)} = \bar{w}_i = \frac{w_i}{w_1 + w_2 + w_3 + w_4} \]

(14)

Layer 5 is the defuzzification layer. Each neuron in this layer is connected to the respective normalization neuron and also receives initial input \( S, T, A, \) and \( H \). A defuzzification neuron calculates the weighted consequent value of a given rule as,

\[ y_i^{(5)} = \bar{w}_i f_i = \bar{w}_i (a_i S + b_i T + c_i NT + d_i NH + e_i) \]

(15)
Where:

- $\bar{w}_i$ is the output of the Layer 4
- $y_i^{(5)}$ is the output of defuzzification neuron $i$ in Layer 5
- $\{a_i, b_i, c_i, d_i, e_i\}$ is a set of consequent parameter of rule $i$

Layer 6 is represented by a single summation neuron. This neuron calculates the sum of outputs of all defuzzification neurons and produces the overall ANFIS output ($y$).

$$y_i^{(6)} = \text{overall output} = \sum_l \bar{w}_i f_i = \frac{\sum_i w_i f_i}{\sum_i w_i}$$ (16)

4.2 Training with different training datasets

The preparation of the input patterns for training the ANFIS involves the conversion of the linguistic terms of the fuzzy rules into numeric values. Initially, there are 44 rules that are the certain and consistent rules, which are obtained from the human experts. The total number of input patterns for the training datasets is 44 rather than 18, because the ‘x’ symbol used in rule-18 in Table 4 should be represented by all possible linguistic terms for the respective antecedents.

The increments of the training datasets are very important until the ANFIS model had provided the best result and reasonably able to classify all of the student performance. Due to insufficient training data problem, the increments of 10 training patterns were proposed. Therefore, besides the 44 input patterns for training, this research also proposes 54, 64 and 69 trained ANFIS model.

In order to determine the best ANFIS model, ten tests had been carried out for each model and calculate their mean square errors (MSE). The error is the difference between the training data output value, and the output of the ANFIS corresponding to the same training data input value. The ANFIS model with the lowest mean square errors is being chosen for the next experiment.

4.3 Results and discussion on ANFIS

This section explains the testing results of the three ANFIS model selected from the trained fuzzy inference system. All the results had been tabulated in a line graphs to compare between ANFIS output based on 44, 54, 64 and 69 training datasets, respectively and the testing data.

In this section, four ANFIS model selected from the previous experiment are selected to test the 81 input data patterns. All the results had been tabulated into a line graphs to compare between the ANFIS output. Fig. 7 shows the comparison between ANFIS outputs based on 44 training datasets and testing data. There are 69.14% of the input patterns which are classified successfully and 30.86% which are misclassified.

Besides that, Fig. 8 shows the comparison between ANFIS outputs based on 54 training datasets and testing data. From the graph, there are 85.19% were classified successfully and 14.81% were misclassified. Therefore, the increment of the training datasets need to be executed, so that able to achieve better result.
After incrementing the training data from 54 to 64, the results seem becomes better. Fig. 9 shows the comparison between outputs of ANFIS model based on 64 training datasets and outputs of the checking data. The outcomes of the trained ANFIS able to achieved up to 96.3% which are classified successfully. However, still have some of outputs are illogical decisions. There are 3.7% of the decisions are illogically.

Thus, another experiment carried out by using the 69 training datasets and finally the all the outputs of the ANFIS are able to classify all the 81 input patterns successfully. We can see it clearly in the Fig. 10. From the graph, both of the outputs are same and the ANFIS model can classify the student performance correctly in all possible conditions.
Moreover, the percentage of successful classification for each input data pattern have been calculated and shown in the Table 4 and Fig. 10. The table below indicates that the human experts’ fuzzy rule base consisting of only 18 rules has the possibility of not giving all classification result. For 81 input datasets have been tested only 62% successfully classified; 1500 random input datasets, 66% successfully give the desired result. Meanwhile ANFIS based on 69 training datasets yield encouraging results than human experts’ fuzzy rule base, they have successfully classified all the given input.

By analyzing and comparing the experimental results for the five fuzzy rule bases, it can be concluded that the human experts’ fuzzy rule base is consistent but incomplete. This is
because the 18 rules in this rule base were carefully selected to give full certainty for decisions. However, we found that not all situations covered by this 18 fuzzy rules and still have some rules are not stated. On the contrary, the complete fuzzy rule base in ANFIS is complete but still got some rules are inconsistent and the decision output is not logically. Although all situations for all four attributes are covered by the set of 81 rules, some of the rules have been found to have unnecessary conditions. Thus, the increment of the training data need to done, so that the ANFIS based on 69 training datasets able to eliminate the unnecessary conditions and the illogical decisions. Finally, the ANFIS model is consistent and complete; all situations for all four attributes are covered by the set of 69 training data, and there are no missing rules.

<table>
<thead>
<tr>
<th>Fuzzy Rule Base</th>
<th>Input data patterns</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>81</td>
</tr>
<tr>
<td>Human Experts</td>
<td>62.00%</td>
</tr>
<tr>
<td>ANFIS (44)</td>
<td>69.14%</td>
</tr>
<tr>
<td>ANFIS (54)</td>
<td>85.19%</td>
</tr>
<tr>
<td>ANFIS (64)</td>
<td>96.30%</td>
</tr>
<tr>
<td>ANFIS (69)</td>
<td>100.00%</td>
</tr>
</tbody>
</table>

Table 5. Percentage of successful classifications correctly

5. Rough-fuzzy approach

ANFIS approach described in Section 4 has successfully formed a complete fuzzy rule that able to solve the problem of incomplete and vague decisions made by human. However, not all rules generated are significant and thus it is important to extract only the most significant rules in order to improve the classification accuracy. In this work, we propose Rough-Fuzzy approach to refine the fuzzy rule base into a concise fuzzy rule base (refer Fig. 11).
5.1 Rough fuzzy phases

The three main phases in the rough-fuzzy approach are data pre-processing, reduct computation and data post-processing as shown in Fig. 11 and described as follows:

**Phase 1.** Data pre-processing.

In this phase, the complete fuzzy rules are converts from linguistic terms into numeric values that correspond to the rough set format.

**Phase 2.** Reduct computation.

The fuzzy rules are mapped into a decision system format, discretisation of data, computation of reducts from data and derivation of rules from reducts.

a. In this problem, the fuzzy rules are mapped as rows; while the antecedents and the consequents of the rules are mapped into columns. In the rough set decision table, the antecedents and consequents of the fuzzy rules are labelled as condition and decision attributes, respectively.

b. Discretisation refers to the process of arranging the attribute values into groups of similar values. It involves the transformation of the fuzzy linguistic descriptions of the conditions and the decision attributes into numerical values. In this study, a conversion scheme is formulated to transform the conditions and decisions of fuzzy linguistic values into numerical representations.

c. Computation of reduct

The reduct computation stage determines the selection of an important attribute that can be used to represent the decision system (Carlin et al., 1998). It is used to reduce the decision system, thus generating more concise rules. The rough set approach employs two important concepts related to reduction: one is related to reduction of rows, and the other one is related to reduction of columns (Chen, 1999). With the notion of an indiscernibility class, the rows with certain properties are grouped together, while with the notion of dispensable attributes, the columns with less important attributes are removed. Another essential concept in reduct computation is the lower and upper approximations, in which the computation involved in the lower approximation produces rules that are certain, while the computation involved in the upper approximation produces possible rules (Øhrn, 2001).

d. Rule Generation. A reduct is converted into a rule by binding the condition attribute values of the object class from which the reduct is originated to the corresponding attribute.

**Phase 3.** Data post-processing

The rules in rough set format are converted into linguistic terms of the concise fuzzy rule base.

5.2 Rough fuzzy experiment

In Section 4, there are 81 datasets that represent every possible value of the fuzzy rules with full certainty. This dataset is used for the development of the ANFIS model. Using Rosetta as rough set tool, the genetic algorithm with object reduct is the method used for computing reducts (Øhrn, 2001). This method implements a genetic algorithm for computing minimal hitting sets as described by Vinterbo and Øhrn (2000). Using rough set, we trained the fuzzy
rules incrementally with different training data set that consist of 44, 54, 64 and 69 input data patterns as described in Section 4. The purpose of the iteration with different input patterns of ANFIS is to ensure that the decision is agreed by human expert.

Table 6 shows the number of reducts, the number of rules and the rule percentage of rough set experiment with different input patterns. The result shows that ANFIS with 69 input patterns generates more concise rule with less number of reducts and less number of rules extracted compared to ANFIS with other pattern.

<table>
<thead>
<tr>
<th>Model</th>
<th>No of Reducts</th>
<th>No of Rules</th>
<th>Percentage of Rules</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Human expert</td>
<td>6</td>
<td>13</td>
<td>16%</td>
</tr>
<tr>
<td>2. ANFIS with 44 input patterns</td>
<td>11</td>
<td>23</td>
<td>28%</td>
</tr>
<tr>
<td>3. ANFIS with 54 input patterns</td>
<td>9</td>
<td>16</td>
<td>20%</td>
</tr>
<tr>
<td>4. ANFIS with 64 input patterns</td>
<td>7</td>
<td>13</td>
<td>16%</td>
</tr>
<tr>
<td>5. ANFIS with 69 input patterns</td>
<td>4</td>
<td>8</td>
<td>10%</td>
</tr>
</tbody>
</table>

Table 6. Number of reducts and rules based on different input patterns

To determine whether the performance of the concise fuzzy rule base is consistent with the performance of the complete fuzzy rule base, each rule bases of input patterns is compared.

Table 7 shows that the decision output given by both the rule bases of each input patterns has very small differences (in terms of its mean square error). This result confirms that the concise fuzzy rule base does not degrade the performance of the complete fuzzy rule base.

It can be seen from Table 7 that ANFIS with 69 input patterns matched exactly as predicted by experts with MSE value equal to zero. The reducts and rules generated by rough set for ANFIS with 69 input patterns is chosen for further discussion.

<table>
<thead>
<tr>
<th>Complete Rule Base (81 Rules)</th>
<th>Concise Rule Base</th>
<th>MSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>ANFIS with 44 input patterns</td>
<td>23 Rules</td>
<td>4.76E07</td>
</tr>
<tr>
<td>ANFIS with 54 input patterns</td>
<td>16 Rules</td>
<td>1.02E07</td>
</tr>
<tr>
<td>ANFIS with 64 input patterns</td>
<td>13 Rules</td>
<td>3.70E10</td>
</tr>
<tr>
<td>ANFIS with 69 input patterns</td>
<td>8 Rules</td>
<td>0.00</td>
</tr>
</tbody>
</table>

Table 7. MSE result of Complete vs Concise Rule Base

Furthermore, Table 8 shows four object-related reduct generated by Rosetta for ANFIS with 69 input patterns. All reducts has 100% support, which mean that all objects are mapped deterministically into a decision class. In other words, the support for the decision rule is the probability of an object to be covered by the description that belongs to the class (Grzymala-Busse, 1991).

<table>
<thead>
<tr>
<th>Class</th>
<th>Reduct</th>
<th>Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>C\textsubscript{1}</td>
<td>{Score}</td>
<td>100</td>
</tr>
<tr>
<td>C\textsubscript{2}</td>
<td>{Attempt}</td>
<td>100</td>
</tr>
<tr>
<td>C\textsubscript{3}</td>
<td>{Score, Attempt}</td>
<td>100</td>
</tr>
<tr>
<td>C\textsubscript{4}</td>
<td>{Score, Attempt, Help}</td>
<td>100</td>
</tr>
</tbody>
</table>

Table 8. Object-related reduct based on ANFIS 69 model
Rules generated from reduct are representative rules extracted from the data set. Since a reduct is not unique, rule sets generated from different reducts contain different sets of rules as shown in Table 9.

<table>
<thead>
<tr>
<th>Rule set</th>
<th>Rules</th>
</tr>
</thead>
<tbody>
<tr>
<td>$R_1$</td>
<td>Score = low =&gt; Performance = not mastered</td>
</tr>
<tr>
<td>$R_2$</td>
<td>Attempt = many =&gt; Performance = not mastered</td>
</tr>
<tr>
<td>$R_3$</td>
<td>Score = moderate AND Attempt = a few =&gt; Performance = moderately mastered</td>
</tr>
<tr>
<td>$R_4$</td>
<td>Score = moderate AND Attempt = average =&gt; Performance = moderately mastered</td>
</tr>
<tr>
<td>$R_5$</td>
<td>Score = high AND Attempt = average =&gt; Performance = moderately mastered</td>
</tr>
<tr>
<td>$R_6$</td>
<td>Score = high AND Attempt = a few AND Help = little =&gt; Performance = has mastered</td>
</tr>
<tr>
<td>$R_7$</td>
<td>Score = high AND Attempt = a few AND Help = average =&gt; Performance = moderately mastered</td>
</tr>
<tr>
<td>$R_8$</td>
<td>Score = high AND Attempt = a few AND Help = needed =&gt; Performance = moderately mastered</td>
</tr>
</tbody>
</table>

Table 9. Rule Generation

For example, the given reduct from Table 8 i.e. reduct {Score, Attempt}, is presented by three rules as shown in Table 9 namely $R_3$, $R_4$, and $R_5$.

$R_3$ : IF Score = moderate AND Attempt = a few THEN Performance = moderately mastered

$R_4$ : IF Score = moderate AND Attempt = average THEN Performance = moderately mastered

$R_5$ : IF Score = high AND Attempt = average THEN Performance = moderately mastered

A unique feature of the rough set method is its generation of rules that played an important role in predicting the output. Table 10 listed the rule generation analysis by Rosetta and provides some statistics for the rules which are support, accuracy, coverage and length. The rule coverage and accuracy are measured to determine the reliability of the rules. Below is the definition of the rule statistics (Bose, 2006).

a. The rule support is defined as the number of records in the training data that fully exhibit property described by the IF condition.

b. The rule accuracy is defined as the number of RHS support divided by the number of LHS support.

c. The conditional coverage is the fraction of the records that satisfied the IF conditions of the rule. It is obtained by dividing the support of the rule by the total number of records in the training sample.

d. The decision coverage is the fraction of the training records that satisfied the THEN conditions. It is obtained by dividing the support of the rule by the number of records in the training that satisfied the THEN condition.

e. The rule length is defined as the number of conditional elements in the IF part.
Coverage gives a measure of how well the objects describe the decision class. The conditional coverage is measured by the ratio of the number of rules that fulfil the conditional part of the rules to the overall number of rules in the sample. Meanwhile, the decision coverage is measured by the ratio of the number of rules that give decision rules to the overall number of rules in the sample. Accuracy gives a measure of how trustworthy the rule is in the condition. It is the probability that an arbitrary object belonging to Class C is covered by the description of the reduct (Grzymala-Busse, 1991). According to Pawlak (1998), an accuracy value of 1 indicates that the classes have been classified into decision classes with full certainty and consistency.

For example, there are 27 objects that fulfil the conditional part of the rule $R_1$, compared with the overall 81 rules in the sample. Therefore, the conditional coverage of this rule is about 0.3333. In addition, the decision for the performance and learning efficiency with the value of not mastered is used once in the fuzzy rule base and it is only given to rule $R_1$. Therefore, the decision coverage for this rule is 1. Finally, the accuracy value of this rule is 1, which means that this rule belongs to Class $C_1$ and is covered. Thus, it is said to have full certainty and is consistent. In conclusion, because all of the rules in Table 10 have accuracy values of 1, the concise fuzzy rules are reliable because they are covered, have full certainty, and are consistent.

### 6. Conclusion

In this study, fuzzy inference models provide an efficient way to reason about a student’s learning achievement in quantitative way. In this work, a complete fuzzy rule base are formed using ANFIS approach, where all possible input conditions of the fuzzy rules are being generated apart from the 18 human experts’ rules that are considered certain. By training the neural network with selected 18 conditions that are certain, the ANFIS is able to recognize other decisions that are previously not complete, in both the antecedents and consequent parts of the fuzzy rules. However, some of the decisions are found misclassified and inconsistent. In addition, it is realized that the number of fuzzy rules formed is directly related to the number of fuzzy term sets defined at the antecedents. As the number of fuzzy term sets increases, the fuzzy rules will also increases and will affect the computation time and space. Besides that, when there are too many rules, some of the rules may be found not
significant. Therefore, this work proposes the Rough-Fuzzy approach that able to reduce the complete fuzzy rule base into a concise fuzzy rule base. This approach able to determine the selection of important attributes that can be used to represent the fuzzy rule base system. Therefore, the condition space is reduced by taking only a few conditions to achieve a reasonable size of the condition subspace. Moreover, the proposed concise fuzzy rule base is said to be reliable, due to the fact that it is covered, consistent and have full certainty.

7. Acknowledgment

The authors are especially grateful to the members of the Soft Computing Research Group (SCRG), Faculty of Computer Science and Information Systems, University of Technology Malaysia, for their encouraging support to this work. The authors would also like to thank Universiti Teknologi Malaysia (UTM) for their financial support under Research University Grant Vot. No. Q.J130000.7128.01H82 and Q.J130000.7128.02J57 as well as the FRGS Grant - Vot No. 4F026(NT:2009957).

8. References


www.intechopen.com


This book is an attempt to accumulate the researches on diverse inter disciplinary field of engineering and management using Fuzzy Inference System (FIS). The book is organized in seven sections with twenty two chapters, covering a wide range of applications. Section I, caters theoretical aspects of FIS in chapter one. Section II, dealing with FIS applications to management related problems and consisting three chapters. Section III, accumulates six chapters to commemorate FIS application to mechanical and industrial engineering problems. Section IV, elaborates FIS application to image processing and cognition problems encompassing four chapters. Section V, describes FIS application to various power system engineering problem in three chapters. Section VI highlights the FIS application to system modeling and control problems and constitutes three chapters. Section VII accommodates two chapters and presents FIS application to civil engineering problem.

How to reference
In order to correctly reference this scholarly work, feel free to copy and paste the following: