A novel Data-driven fuzzy aggregation method for Takagi-Sugeno-Kang fuzzy Neural network system using ensemble learning


Published in:

Document Version:
Peer reviewed version

Queen's University Belfast - Research Portal:
Link to publication record in Queen's University Belfast Research Portal

Publisher rights
Copyright 2021 IEEE. This work is made available online in accordance with the publisher’s policies. Please refer to any applicable terms of use of the publisher.

General rights
Copyright for the publications made accessible via the Queen's University Belfast Research Portal is retained by the author(s) and / or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy
The Research Portal is Queen’s institutional repository that provides access to Queen’s research output. Every effort has been made to ensure that content in the Research Portal does not infringe any person’s rights, or applicable UK laws. If you discover content in the Research Portal that you believe breaches copyright or violates any law, please contact openaccess@qub.ac.uk.
A novel Data-driven fuzzy aggregation method for Takagi-Sugeno-Kang fuzzy Neural network system using ensemble learning

Tao Wang
School of Electronics, Electrical Engineering, and Computer Science
Queen’s University Belfast
Belfast, United Kingdom
twang17@qub.ac.uk

Richard Gault
School of Electronics, Electrical Engineering and Computer Science
Queen’s University Belfast
Belfast, United Kingdom
richard.gault@qub.ac.uk

Des Greer
School of Electronics, Electrical Engineering and Computer Science
Queen’s University Belfast
Belfast, United Kingdom
des.greer@qub.ac.uk

Abstract— Fuzzy aggregation operators commonly rely on expert information to solve multi-attribute decision-making problems. Such expert input may contain human bias or may sometimes be unavailable. This paper proposes a novel data-driven fuzzy aggregation method for Takagi-Sugeno-Kang fuzzy neural networks (TSKFNN) based upon the ensemble learning algorithm, AdaBoost. The objective of this research is to investigate whether ensemble learning is an effective tool for data-driven fuzzy aggregation. Our hypothesis is that ensemble learning would improve model performance and explainability. In this study, AdaBoost is applied to get a weighted combination of fuzzy rules in the TSKFNN and calculate the weighted average of these fuzzy rules to generate model predictions. Existing fuzzy aggregation operators are used as benchmarks to evaluate the proposed model. The results show that the proposed model is capable of yielding higher accuracy and greater interpretability than the existing methods through the identification of the most significant fuzzy rules used in the decision-making process.

Keywords— Fuzzy aggregation method, AdaBoost, TSK Fuzzy-Neural network, multi-attribute decision-making problems, Ensemble Learning

I. INTRODUCTION

Fuzzy-neural networks are effective tools to deal with decision-making problems [1] and modelling uncertainty [2]. Two common types of neuro-fuzzy networks are the Mamdani integrated fuzzy-neuro networks, where the rule consequents are fuzzy sets [3] and Takagi-Sugeno-Kang integrated fuzzy-neural networks (TSKFNN) [4]. Then a crisp result is obtained through the defuzzification of rule consequents in the Mamdani system while this process is implicit in the neural network computation of the TSKFNN [5]. In this paper, the TSKFNN architectures are considered suitable to deal with model regression problems on account of their simplicity and flexibility [6].

Given a large number of rules in fuzzy systems, aggregation operators are proposed to aggregate fuzzy rules in the fuzzy rule base and generate a reliable output [7]. Some fundamental information aggregation operators have already been proposed. For instance, a fuzzy weighted average (FWA) method has been applied to multi-attribute decision making problems [8]. Subsequently, an improved fuzzy weighted average approach was proposed in [9] and [10] to reduce calculation costs. Expert defined aggregation of fuzzy rules may cause problems due to human factors (such as unconscious bias, fatigue and mood). Conversely, a data-driven method might be more useful for developing a fuzzy system. Data-driven aggregation operators need to be developed to aggregate information objectively.

In recent years, Tian et al. [11] defined some hesitant fuzzy linguistic Shapley fuzzy aggregation operators. Every element has 3 membership degrees in picture fuzzy sets: positive membership, neutral membership, and negative membership. In the current investigation we will only consider membership degrees belonging to the closed interval [0,1]. Nie et al. [12] defined the Pythagorean fuzzy partitioned normalized weighted Shapley Bonferroni mean operators. This research is focused on a Pythagorean fuzzy set where the summation of positive membership degree and negative membership degree is allowed to be larger than 1. Unfortunately, these aggregation operators are not data-driven.

Wu et al. [13] propose three novel techniques (DropRule, DropMF, and Drop-Membership) specifically for training Takagi-Sugeno-Kang (TSK) fuzzy systems, which can achieve fast convergence and also superior generalization performance in testing. A data-driven firing-level weighted average aggregation operator is used with the consequent parameters tuned during model training. However, a fuzzy rule trained by the TSK neural network may produce an unsatisfactory test output with high error. This may be due to problems such as overfitting or underfitting. In order to combat this problem, AdaBoost is proposed to aggregate fuzzy rules in a reliable strategy; it has four primary merits:

1. It produces classifiers with high precision by reducing deviations in supervised learning;
2. The AdaBoost algorithm provides a framework where various methods can be utilized to construct the sub-classifier;
3. Strong interpretability; and
4. Strong generalization ability.

AdaBoost [14] will be used to assign each fuzzy rule a weight according to its prediction performance during the training phase indicating the importance of rules. The rules that lead to improved model prediction will be given large weights whilst poorer performance will lead to the assignment of smaller weights. The fuzzy rules with both high firing strengths and high weights will have a large influence on the final prediction, which should result in high model accuracy.

This study investigates whether ensemble learning, in particular AdaBoost, is an effective tool for data-driven fuzzy aggregation. Our hypothesis is that ensemble learning [15] enables TSKFNN to have stronger interpretability and generalization ability than existing research. A novel data-
II. Methodology

This section explains the structure of the proposed TSKFNN with AdaBoost fuzzy aggregation method. The structure of this model is illustrated in Figure 1. TSK fuzzy systems usually have 6 core components: the input layer, the fuzzification layer, the rule antecedent layer, the rule consequent layer, rule strength normalization layer, AdaBoost-fuzzy aggregation layer [16]. In this section, each layer will be briefly introduced.

A. The input layer

The inputs of the TSKFNN are all crisp values. The elements of the input vector are passed to the fuzzification layer.

The fuzzification layer

The fuzzification layer converts the crisp input values to fuzzy values with a membership degree to a particular fuzzy set. In terms of natural language, there are many words with vague concepts. These fuzzy concepts cannot be described by a classic set, because they cannot be expressed by a set that “belongs to” or “does not belong to”. That is, the degree to which the elements on the domain conform to the concept is not 0 or 1, but a real number between 0 and 1. Zadeh’s definition of fuzzy sets [17] is: Given a fuzzy set \( f \) with the universe of discourse \( U \) the membership function of the fuzzy set \( f \) is formulated by Equation (1)

\[
\mu_f: U \rightarrow [0, 1] \quad (1)
\]

Given the input vector \( X = [x_1, x_2, x_3, ..., x_n] \), \( \mu_f(x_i) \) refers to the degree of membership of an element \( x_i \) belonging to a fuzzy subset \( f \), where \( 1 \leq i \leq n \). The membership function is a quantitative description of the fuzzy concept, and the correct determination of the membership function is the basis for solving practical problems using the fuzzy set theory.

In this paper, Gaussian and Trapezoidal membership functions are considered in line with the approach used by other models that will be used as a comparative benchmark [13]. In this work, each feature \( x_i \) has 2 membership functions (representing low and high input values) for each fuzzy set, in keeping with previous approaches [13]. The membership degree of \( x_i \) on a Gaussian membership function \( \mu(x_i) \) is:

\[
\mu(x_i) = \exp \left( -\frac{(x_i-c_i)^2}{\sigma_i^2} \right) \quad (2)
\]

where \( c_i \) is the center of the Gaussian membership function and \( \sigma_i \) is the standard deviation \( (i = 1, 2, 3, ..., n, j = 1, 2, 3, ..., R) \). The fuzzification layer finds the membership degrees of input vectors and passes these values to the rule antecedent layer. The membership degree of \( x_i \) in the \( j \)-th fuzzy rule on a trapezoidal membership function \( \mu(x_i) \) is:

\[
\mu(x_i) = \begin{cases} 
0 & x \leq A_i^j \\
\frac{x - A_i^j}{B_i^j - A_i^j} & A_i^j \leq x \leq B_i^j \\
1 & B_i^j \leq x \leq C_i^j \\
\frac{D_i^j - x}{D_i^j - C_i^j} & C_i^j \leq x \leq D_i^j \\
0 & x \geq D_i^j 
\end{cases} \quad (3)
\]

where \( A_i^j, B_i^j, C_i^j, D_i^j \) are the four vertex parameters of the trapezoidal membership functions.

B. The rule antecedent layer

Consider a TSKFNN with \( n \) inputs \( X = [x_1, x_2, x_3, ..., x_n] \) and \( R \) fuzzy rules where each fuzzy rule \( r \) takes the form

\[
r: \text{if } x_1 \in f_1^r, x_2 \in f_2^r, ..., x_n \in f_n^r \text{ then } y_r = p_{0r} + p_{1r} * x_1 + p_{2r} * x_2 + \cdots + p_{nr} * x_n \quad (3)
\]

where \( f_i^r (1 \leq r \leq R, 1 \leq i \leq n) \) is the fuzzy set of \( r \)-th fuzzy rule for \( i \)-th input \( x_i \) and \( p_{ir} \) is the parameter of the rule consequent layer of \( r \)-th fuzzy rule for \( i \)-th input \( x_i \), \( y_r \) is the output of \( r \)-th fuzzy rule. These parameters are randomly defined at first and refined during training of the TSKFNN.

The rule antecedent layer produces the firing strengths of fuzzy rules. T-norm operations, \( T: [0, 1] \times [0, 1] \rightarrow [0, 1] \) [18] are employed to compute these firing strengths. The firing strength of \( r \)-th fuzzy rule \( w_r \) is defined as:

\[
w_r = \prod_{i=1}^{n} \mu_r(x_i) \quad (4)
\]

The firing strengths of all fuzzy rules are obtained in the rule antecedent layer and then are passed to the rule consequent layer.

C. The rule consequent layer

The second part of a TSK fuzzy rule, \( y_r = p_{0r} + p_{1r} * x_1 + p_{2r} * x_2 + \cdots + p_{nr} * x_n \), is the consequent part of \( r \)-th fuzzy rule. The parameters \( p_{kr} \) (\( k = 0, 1, 2, ..., n \)) are initially random and subsequently learnt during training. The rule consequent
layer produces crisp values using the consequent part of the rules for each rule antecedent.

D. AdaBoost

AdaBoost is an iterative algorithm over a collection of weak learners [19]. At each iteration a new learner is trained and the samples of the training data set are re-weighted. The training samples with larger prediction error are given greater weight, which means a learner’s importance will decrease significantly next iteration if it leads to a poor prediction. Then a voting strength is assigned to each learner. The voting strengths are based on the learners’ prediction accuracy; a larger voting strength indicates that the weak learner has a greater impact on the final decision. Finally, a compound learner, $F$, with increased accuracy is generated by combining all weak regressions.

In this paper, each fuzzy rule in the fuzzy system’s rule base is regarded as a weak learner. AdaBoost will be used to assign every fuzzy rule a weight in the light of their prediction performance that indicates the importance of a fuzzy rule in the overall decision-making process. A larger weight means a greater impact on the final output when all fuzzy rules are aggregated.

Given the training input-output data set $(X_1, Y_1), (X_2, Y_2), \ldots, (X_m, Y_m)$ with $m$ samples, $i$-th sample’s weight $\omega_i$ is initialized as $1/m$ and all of the $R$ fuzzy rules are ordered randomly. The error, $E_r$, generated by the $r$-th fuzzy rule during training is calculated as follows:

$$E_r = \max |Y_i - y_r(X_i)| \quad (1 <= i <= m) \quad (5)$$

The error is the difference between the real value of the $i$-th training sample, $Y_i$ ($i = 1, \ldots, m$), and the predicted value of $i$-th sample in the $r$-th fuzzy rule; denoted $y_r(X_i)$ ($r = 1, \ldots, R$). There are $m$ errors for $m$ training samples. Then the largest error $E_r$ is selected among the $m$ samples for the $r$-th fuzzy rule.

In the second step, the relative error $e_{ri}$ of sample $i$ is calculated as

$$e_{ri} = \frac{|Y_i - y_r(X_i)|}{E_r} \quad (6)$$

The total error rate of the $r$-th fuzzy rule, $e_r$, is the sum of the product of each sample weight $\omega_i$ and the relative error rate $e_{ri}$ is obtained by

$$e_r = \sum_{i=1}^{m} \omega_i \cdot e_{ri} \quad (7)$$

Note that $e_r < 1$. From (7), the voting strength $\alpha_r$ of the $r$-th fuzzy rule is formulated as

$$\alpha_r = \frac{1-e_r}{e_r} \quad (8)$$

A larger fuzzy rule’s voting strength infers that the fuzzy rule has a greater impact on the final decision. Equation 9 shows how the $i$-th training sample’s weight update in the $r$-th iteration:

$$\omega_i' = \frac{\omega_i}{Z_i} \cdot \alpha_r^{1-e_{ri}} \quad (9)$$

where $\omega_i'$ is the new sample weight after updating. The normalization factor of $i$-th training sample, $Z_i$, in $r$-th fuzzy rule is

$$Z_i = \sum_{i=1}^{m} \omega_i \cdot \alpha_r^{1-e_{ri}} \quad (10)$$

After $R$ iterations of training for all fuzzy rules, voting strengths are assigned to $R$ fuzzy rules and samples’ weights have also been updated for $R$ times. These voting strengths are then passed into fuzzy aggregation layer.

E. Fuzzy aggregation operators

The proposed data-driven fuzzy aggregation strategy is defined as

$$f(X) = \sum_{r=1}^{R} w_r \cdot y_r(X) \cdot \alpha_r \quad (11)$$

Herein, $X$ represents a training sample, $f(X)$ is the output of fuzzy aggregation operator, $w_r$ is the firing strength of $r$-th fuzzy rule, $y_r(X)$ denotes the output of the consequent layer, $\alpha_r$ is the voting strength of the $r$-th fuzzy rule. This output is passed to the backpropagation part of the TSKFNN.

F. Backpropagation

TSKFNN’s loss function is defined as

$$E = \frac{1}{2} (Y - f(X))^2 \quad (12)$$

where $Y$ is the the ground truth output value, $f(X)$ is the predicted output, and $E$ represents the error between expected output and actual output. Updating of the rule consequent layer’s parameters is carried out by calculating

$$p_{k}^r = p_{k}^r - Lr \cdot \frac{\partial E}{\partial p_{k}} \quad (13)$$

$p_{k}^r$ is the parameter of $r$-th fuzzy rule consequent for $k$-th input feature $x_k$. $p_{k}^r$ is the updated parameter and $Lr$ is the learning rate. The updated parameters of the Gaussian membership functions are calculated by

$$c_{i}^r = c_{i}^r - Lr \cdot \frac{\partial E}{\partial c_{i}^r} \quad (14)$$

$$\sigma_{i}^r = \sigma_{i}^r - Lr \cdot \frac{\partial E}{\partial \sigma_{i}^r} \quad (15)$$

where $c_{i}^r$ is the center of the Gaussian membership function and $\sigma_{i}^r$ is the standard deviation ($i = 1, \ldots, n$ and $j = 1, \ldots, M_m$) with $c_{i}^r$ and $\sigma_{i}^r$ being the corresponding updated parameters. The pseudo-code of proposed TSKFNN with AdaBoost fuzzy aggregation method is given in Algorithm 1 overleaf.

The updated parameters of the Trapezoidal membership functions are obtained by:

$$A_{i}^r = A_{i}^r - Lr \cdot \frac{\partial E}{\partial A_{i}^r} \quad (16)$$

$$B_{i}^r = B_{i}^r - Lr \cdot \frac{\partial E}{\partial B_{i}^r} \quad (17)$$

$$C_{i}^r = C_{i}^r - Lr \cdot \frac{\partial E}{\partial C_{i}^r} \quad (18)$$

$$D_{i}^r = D_{i}^r - Lr \cdot \frac{\partial E}{\partial D_{i}^r} \quad (19)$$

$A_{i}^r, B_{i}^r, C_{i}^r, D_{i}^r$ are the four vertex parameters of the trapezoid membership functions.

Input: m labeled training samples \( X = \{x_1, x_2, \ldots, x_m\} \)
- \( M_m \), the number of Gaussian/Trapezoidal membership functions in the \( n \)-th input domain;
- \( K_n \), the maximum number of training iterations;
- \( B \), the mini-batch size;
- \( n \), the number of features for one sample;

Output: \( Y_{\text{pred}} \), the predicted output of training samples;

for \( i = 1, \ldots, n \) do

Initialize the center of the \( M_m \) Gaussian/Trapezoidal membership functions randomly.

Initialize the standard deviation of all \( M_m \) Gaussian/Trapezoidal membership functions as the standard deviation;

end

Initialize the consequent parameters of all \( R \) rules as 0, \( R \geq M_m \); for \( k = 1, \ldots, K \) do

Randomly select \( B \) training examples;

for \( i = 1, \ldots, B \) do

Initialize the sample weights \( W \) of \( B \) samples;

The membership degree of \( x \) on a Gaussian/Trapezoidal membership function is computed;

for \( r = 1, \ldots, R \) do

The firing strength of \( r \)-th fuzzy rule \( w_r \) is obtained;

end

The rule consequent of \( r \)-th fuzzy rule \( y_r \) is computed;

for \( r = 1, \ldots, R \) do

The firing strength of \( r \)-th fuzzy rule \( w_r \) is obtained;

end

Compute the maximum error \( E_r \);

Compute the relative error \( f_{x_r} \);

Compute the voting strength \( \sum_{i} (A_{i} < 1) \) of the \( r \)-th fuzzy rule;

The training sample's weight \( W \) is updated;

end

Return \( Y_{\text{pred}} \);

end

Algorithm 1: Pseudo-code of the training process for the proposed TSKFNN with AdaBoost aggregation

III. Experimental Outline

This section outlines the experimental setup to analyze the efficacy of the proposed AdaBoost fuzzy aggregation method. The subsequent results are outlined in Section IV.

A. Datasets

To evaluate the proposed model and compare the model's capabilities against existing models, 3 different datasets from the UCI Machine Learning Repository will be modelled. These datasets are selected as they have previously been used in a TSKFNN investigation [13] that will be used as a benchmark for the proposed model. The datasets are briefly summarized in Table I.

<table>
<thead>
<tr>
<th>Dataset name</th>
<th>Data source</th>
<th>No. of samples</th>
<th>No. of raw features</th>
<th>No. of numerical features</th>
<th>No. of features</th>
<th>No. of Used features</th>
</tr>
</thead>
<tbody>
<tr>
<td>Airfoil(^1)</td>
<td>UCI</td>
<td>1503</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>Powerplant(^2)</td>
<td>UCI</td>
<td>9568</td>
<td>4</td>
<td>4</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Student performance(^3)</td>
<td>UCI</td>
<td>395</td>
<td>32</td>
<td>5</td>
<td>5</td>
<td></td>
</tr>
</tbody>
</table>


The Airfoil dataset consists of different sized NACA 0012 airfoils subjected to various wind tunnel speeds and angles of attack. The 5 numerical inputs represent Frequency (Hz), the angle of attack (in degrees), chord length (m), free-stream velocity (m/s), and suction side displacement thickness (m) while the output parameter for this dataset is scaled sound pressure level (dB). For the Powerplant dataset, hourly measurements of temperature (°C), ambient pressure (millibar), relative humidity (%) and exhaust vacuum (cm Hg) are recorded and used to help predict the net hourly electrical energy output (in megawatt) of a powerplant. Finally, the student performance dataset has 32 input domains. It has not only numerical features but also categorical features such as father's job, mother’s job and sex. Fuzzy systems have difficulty dealing with high dimensional data [13], the maximum input dimensionality is constrained to be no more than five input features. The selected five input features are chosen to be consistent with similar investigations [13] that will be used to benchmark the findings of this paper. From the 15 numerical features, neighborhood component analysis is used to reduce the number of features to five. Neighborhood component analysis identified the 5 numerical features to be the number of past class failures, the free time after school, the current health status, first period grade and second period grade for each student. The aim is to use these features to predict the final grade in the student performance dataset.

B. Models

The proposed TSKFNN with AdaBoost aggregation is benchmarked against two existing models [13]. These models have been chosen as state-of-the-art TSKFNN models and will form the basis of preliminary investigation. Future work will consider alternative fuzzy modelling architectures. Therefore, the models considered in this work are:

1. Mini-Batch Gradient Descent with Regularization, DropRule, AdaBound, and Gaussian membership functions (MBGD -RDAG) introduced in [13].
2. Mini-Batch Gradient Descent with Regularization, DropRule, AdaBound and Trapezoidal membership functions (MBGD -RDAT) introduced in [13].
3. Mini-Batch Gradient Descent TSKFNN with the proposed AdaBoost fuzzy aggregation method and Gaussian membership functions.
4. Mini-Batch Gradient Descent TSKFNN with the proposed AdaBoost fuzzy aggregation method and trapezoidal membership functions.

The models are implemented in MATLAB 2020a, the batch size is 64, the initial learning rate is 0.01 and the number of iterations is 500. We chose 70 percent of the samples at random for training and the remaining 30 percent for testing in each dataset. Every input is normalized to [0,1] to scale the data proportionally and remove the data unit limitation. Each input domain has 2 membership functions (\( M_m = 2 \)) in keeping with [13].

IV. Experimental Results

The performance of the models in replicating the test datasets is quantified using the root mean square errors (RMSE). Figures 2-4 show the performance of each model across the Airfoil, Powerplant, and Student performance datasets respectively on the test dataset after each training iteration.

As shown in Figure 2 the RMSE of the TSKFNN with AdaBoost and Gaussian membership function decreased sharply during the first to 300 iterations. The RMSE fluctuates between 3.8 and 4.8 during the 301st to 500th iteration, then it amounts to 3.882 by the last iteration. As shown in Figure 2 the RMSE of the TSKFNN with AdaBoost and Gaussian membership function decreased sharply during the first to 300 iterations. The RMSE fluctuates between 3.8
and 4.8 during the 301st to 500th iteration, then it amounts to 3.882 by the last iteration. The TSKFNN’s with AdaBoost aggregation generally performed better than the benchmark models in modelling the Airfoil dataset. It is notable that the MBGD RDAG was more stable across iterations than the other approaches despite having poor performance.

It is clear from Figures 3 and 4 that the TSKFNN with AdaBoost aggregation performed at least as well as the existing approaches, if not slightly better in general, and is competitive in performance compared with the existing methods. It is noteworthy that the student performance dataset has only 395 samples, which means the TSKFNN with AdaBoost aggregation might be a possible model to overcome the problem of less data in practice.

In general, the TSKFNN with Gaussian membership functions outperformed their counterparts with Trapezoidal membership functions (Figure 2-4) irrespective of the aggregation approach used. The TSKFNN with AdaBoost aggregation and Trapezoidal membership functions consistently outperformed the MBGD RDAT (Figure 2-4). The experimental results show that AdaBoost aggregation is helpful to increase regression accuracy irrespective of whether Gaussian or Trapezoidal membership functions are used.

In the context of machine learning systems, interpretability is defined as the ability to explain or to present in understandable terms to a human [20]. Linguistic fuzzy rules enable a fuzzy model to become interpretable by humans [21]. The data driven weights of the fuzzy rules established through the AdaBoost algorithm establish which fuzzy rule is most important and which fuzzy rules are less significant providing greater transparency in the decision-making process and thus increasing the model’s explainability. The rule weights are different from the coefficients of the rule consequents. A larger fuzzy rule weight means a larger importance in the decision-making process. The equivalent weight does not exist in the benchmark models. The fuzzy rules have the most impact in the decision-making process for each dataset are shown in Table II.

<table>
<thead>
<tr>
<th>Dataset name</th>
<th>Fuzzy rule</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Airfoil</strong></td>
<td>If the frequency is low and the angle of attack is low and chord length is low and free-stream velocity is high and suction side displacement thickness is low then the sound pressure level = 2.687<em>frequency + 0.403</em> the angle of attack - (0.984)* chord length + (-0.148)* free stream velocity + 2.836*suction side displacement thickness - 0.91</td>
</tr>
<tr>
<td><strong>Powerplant</strong></td>
<td>If the temperature is high and ambient pressure is high and relative humidity is low and Exhaust Vacuum is high then the Net electrical energy = (-7.610) * temperature + (-6.256) * ambient pressure + (3.106) * humidity - 2.520 * Exhaust Vacuum - 1.367</td>
</tr>
<tr>
<td><strong>Student performance</strong></td>
<td>If the number of past class failures is low and free time after school is low and current health status is high and first period grade is low and second period grade is high then the final grade = 0.794 * number of past class failures + (0.595) free time after school + (0.311) * current health status + 0.569* first period grade + 0.182 * second period grade + 2.046</td>
</tr>
</tbody>
</table>

For example, our proposed model can help to extract the top fuzzy rules in student performance evaluation and educators use this information to give assistance and advice to students who are in the risk of failing with help from the fuzzy rules. In this case it is resolvable, and indeed understandable, that final grades are best predicted in cases
with low past failures and high second period grades. Informed interventions could therefore be introduced if necessary by monitoring all noted variables.

V. CONCLUSION AND FUTURE RESEARCH

This paper is addressing the need for data-driven fuzzy aggregation operators with stronger interpretability. This paper explores the possibility of improving the accuracy and enhancing the interpretability of TSKFNN, in particular those developed in [13], by utilizing the ensemble learning algorithm AdaBoost in the aggregation operation. The findings show that the proposed AdaBoost aggregation within a TSKFNN is comparable to and even improves upon existing modelling approaches as shown across multiple datasets. Moreover, the proposed model enhances the interpretablility of TSK fuzzy systems by assigning each fuzzy rule a weight, which indicates the importance of a fuzzy rule in the overall decision-making process. While the existing models cannot provide the information about which fuzzy rule is more valuable and which one is relatively less impact on the final output. Furthermore, one important advantage of the fuzzy model compared to the probability-based approach is that it can deal with the fuzzy concepts effectively. Future work is planned to explore ensemble pruning in the proposed model to eliminate fuzzy rules with poor performance and consequently enhancing the generalization of the model. Another benefit would be that ensemble selection contributes to reduction of storage memory and computational resources. Possible pruning solutions to investigate will include ordering-based, clustering-based and optimization-based.

ACKNOWLEDGEMENTS

The authors would like to thank Prof. Dongrui Wu, HUST for his guidance and code base for the implementation of the benchmark models used in this work. This work supported by a Queen’s University/China Scholarship Council PhD Scholarship.

REFERENCES