Density-based approach for fuzzy rule interpolation

Fei Gao*

School of Aeronautics, Shandong Jiaotong University, Jinan, Shandong, 250357 China

Abstract

Sparse rule base is one of the common problems in fuzzy rule-based systems, and fuzzy rule interpolation (FRI) could derive interpolated results for the input based on the neighbor fuzzy rules when the input is not matched by any of the fuzzy rules. The core idea of FRI is that similar inputs would lead to similar results, and several FRI methods that use a pre-defined number of closest rules to obtain the interpolated results have been presented. However, this could lead to the loss of some information as selecting a given number of rules without considering the exact distance between them and the input could lead to the selection of unwanted rules or the ignoring of similar rules. This paper presents a density-based fuzzy rule interpolation method that uses a density-based approach to search and select the closest rules for unmatched inputs. Instead of selecting a given number of rules, the proposed method adaptively selects the closest rules that are within a certain range of the unmatched inputs, thus assuring the selected rules are with high similarity to the inputs. The performance of the proposed method is verified through fifteen classification benchmarks, showing the effectiveness and efficiency of the proposed method.

Keywords: Fuzzy rule interpolation, Fuzzy rule base, Density-based approach, Rule selection

1. Introduction

As one of the most popular and successful soft computing methods, fuzzy rule-based systems have been widely used in many real-world applications

*Corresponding authors

Email address: gaofei1995@hotmail.com (Fei Gao)

[1–4]. Normally, to ensure the successful application of such a system, a dense rule base that covers the entire input space to ensure the overlap with existing rules for any new inputs is usually required. However, there are cases where the knowledge is incomplete, as such a sparse rule rather than a dense rule base could be available [5–7]. In such cases, an input may fail to match and activate any of the existing rules from the rule base, thereby leading to no conclusion. For such a problem, the concept of interpolating in sparse rule bases, known as fuzzy rule interpolation (FRI), is introduced for fuzzy systems where the rule base is sparse and unable to cover the input space [8–11].

Various FRI approaches and their variations have been proposed in the literature [12–16], and they can be generally divided into two categories. The first one is called α-cuts [17, 18], where the interpolation is conducted by directly manipulating the antecedents of rules that are considered closest to the given input. Hence, the consequent of the interpolated result could be viewed as the combined outcome of all these rules involved. The second one is called scale and move transformation-based FRI (TFRI) [19–21], which is based on analogical reasoning, i.e., similar inputs would result in similar consequents, and an intermediate rule is normally constructed and transformed to obtain the consequent for the unmatched input.

TFRI has been popular for its ability to construct the intermediate rule to accurately model the inputs of the unmatched inputs, thus enabling reliable and reasonable interpolated results [22–24]. However, as noted by many previous studies, the interpolated results depend heavily on how the intermediate rule is constructed, more specifically, as the intermediate rule is normally constructed by combining several closest rules, the search and selection of these closest rules directly affect the interpolated results [25–27]. In many conventional T-FRI methods, the selection of closest rules is achieved by selecting a pre-defined number of closest rules, however, the selection of these rules does not necessarily consider how close these rules are to the input, which may result in some low similarity rules to be selected, or some high similarity rules to be ignored. Regarding these limitations, how to properly search and select rules that are close to the inputs to construct the intermediate rule remains one of the significant challenges for FRI.

The existing methods mainly use the user-defined number to determine how many rules will be selected. However, this could lead to the loss of information as a fixed number of rules is selected regardless of the actual distance between the selected rules and the input [5]. Automatic selection of closest rules considering the distance between the rules and the input regardless of the exact number of selected rules is clearly preferred. Recently, the density-based approach, which adaptively searches for data that are close to the given inputs while only considering the distance, has attracted much attention and been applied to many clustering problems [28–31]. Thus, one can adopt the density-based approach to adaptively select the closest rules to the inputs from the distance perspective.

Motivated by the above, it is interesting to investigate whether selecting the closest rules within a certain range instead of selecting a fixed number of rules for the unmatched inputs could provide better performance. To this end, this paper presents a density-based TFRI (D-TFRI) method, in the proposed method, the search and selection of closest rules are carried out in a density-based fashion, that is, instead of defining the number of rules to be selected, the proposed method selects closest rules simply according to their distance to the inputs, and rules that are within the selection range are selected and used to generate the intermediate rule, where the weights of the selected rules are determined based on their distance to the input. Moreover, the scale and move operation of the conventional TFRI method is retained for the constructed intermediate rule, and the interpolated results could be obtained for unmatched inputs. Experiments are carried out to classical classification benchmarks, and the results show that the proposed method could outperform conventional TFRI methods. Moreover, the proposed method is shown to perform well when there are few rules in the sparse rule base or when there are noisy data in the training datasets.

The remainder of this paper is organized as follows. Section 2 reviews the relevant background of the proposed method. Section 3 details the process of the proposed D-TFRI method. The performance of the D-TFRI method is verified through experiments in Section 4. Section 5 concludes the paper.

2. Preliminaries

This section briefly reviews the basic interpolation process of conventional TFRI. Suppose that a sparse rule base R with a set of fuzzy rules r_k is given for a n-dimensional problem, where the rule r_k is represented as [5]:

$$IF\ x_1\ is\ A_1^k\ and\ \ldots\ and\ x_n\ is\ A_n^k,\ THEN\ y^k \tag{1}$$

where x_i is the i antecedent feature, which is described by the fuzzy value A_i^k, and y^k is the consequent.

2.1. Representative values

Normally, triangular fuzzy membership functions are used to represent fuzzy sets. Let A_i be a tuple representing a triangular fuzzy set $A_i = (a_{i1}, a_{i2}, a_{i3})$, where a_{i1}, a_{i2}, and a_{i3} are the left, normal and right point of the fuzzy set, respectively. Then the representative value $Rep(A_i)$ that describes the overall geometric shape and location of A_i, is defined as [22]:

$$Rep(A_i) = \frac{a_{i1} + a_{i2} + a_{i3}}{3} \tag{2}$$

2.2. Selection of closest rules

Given an input $o^* = (A_1^*, A_2^*, \ldots, A_n^*)$, where A_k^* denotes the kth feature value of the input, then the distance between a rule r^k and the input is calculated as the aggregated distance of all the antecedent features, which can be denoted as follows [25]:

$$d(r^k, o^*) = \sqrt{\sum_{i=1}^{N} d(A_i^k, A_i^*)^2} \tag{3}$$

where $d(A_i^k, A_i^*)$ denotes the normalized distance of the absolute distance measure, such that

$$d(A_i^k, A_i^*) = \frac{|Rep(A_i^k) - A_i^*|}{\max_{A_i} - \min_{A_i}} \tag{4}$$

where $|Rep(A_i^k) - A_i^*|$ is the absolute distance between the observed feature value A_i^* and the representative value of the fuzzy set A_i^k for the corresponding attribute x_i, and $\max_{A_i}$ and $\min_{A_i}$ denote the maximal and minimal value of x_i, respectively. Once the distances between the given input and all the rules in the rule base are calculated, the $l(l \geq 2)$ rules that have the minimal distances to the input are selected as the closest l rules to the input.

2.3. Intermediate rule construction

Let $\omega_{A_i^k}$ denote the weight of the ith antecedent fuzzy set A_i^k of the kth rule r^k such that

$$\omega_{A_i^k} = \frac{\omega'_{A_i^k}}{\sum_{k=1}^{l} \omega'_{A_i^k}} \tag{5}$$

where ${\omega'}_{A_i^k}$ represents the similarity between the antecedent fuzzy set A_i^k and the corresponding fuzzy value of the input, which is defined as:

$$ {\omega'}_{A_i^k} = \frac{1}{d(A_i^k, A_i^*) + 1} \tag{6} $$

The intermediate fuzzy variable A_i'' over i is constructed from the antecedents of the l closest rules, which are moved to A_i' to ensure that they would have the same representative values as A_i^* [24]:

$$ A_i' = A_i'' + \delta_{A_i}(\max_{A_i} - \min_{A_i}) \tag{7} $$

where

$$ A_i'' = \sum_{k=1}^{l} \omega_{A_i^k} A_i^k \tag{8} $$

$$ \delta_{A_i} = d(A_i^*, A_i'') \tag{9} $$

Then, the moved intermediate consequent y' can be computed using the parameters ω_{y^k} and δ_y by aggregating the n corresponding values of A_i', such that

$$ y' = \sum_{k=1}^{l} \omega_{y^k} y^k + \delta_y(\max_{y} - \min_{y}) \tag{10} $$

where ω_{y^k} and δ_y are calculated by

$$ \omega_{y^k} = \frac{1}{n} \sum_{i=1}^{n} \omega_{A_i^k} \tag{11} $$

$$ \delta_y = \frac{1}{n} \sum_{i=1}^{n} \delta_{A_i} \tag{12} $$

Subsequently, the scale and move transformation is carried out on the intermediate rule to obtain the interpolated consequent, whose aim is to ensure that the antecedent feature values of the intermediate rule will coincide with their corresponding fuzzy values in the unmatched input. The transformations are implemented in two stages: scale operation and move operation. For scale operation, A_i' is transformed to $\hat{A}_i'$ as the scale intermediate fuzzy set to determine the scale rate s_A. For move operation, $\hat{A}_i'$ is transformed to A_i^* to obtain a move ratio m_{A_i}.

Let $A_i' = (a_{i1}', a_{i2}', a_{i3}')$ be a triangular intermediate fuzzy set, the scale rate s_{A_i} is calculated by [5]:

$$s_{A_i'} = \frac{a_{i3}^* - a_{i1}^*}{a_{i3}' - a_{i1}'} \tag{13}$$

which modifies the support length of A_i', i.e., $a_{i3} - a_{i1}$ so that it becomes the same as that of the input A_i^*. The scaled intermediate fuzzy set $\hat{A}_i'$ is obtained as [23]:

$$\begin{cases} \hat{a_{i1}'} = \dfrac{(1+2s_{A_i})a_{i1}' + (1-s_{A_i})a + i2' + (1-s_{A_i})a_{i3}'}{3} \\ \hat{a_{i2}'} = \dfrac{(1-s_{A_i})a_{i1}' + (1+2s_{A_i})a + i2' + (1-s_{A_i})a_{i3}'}{3} \\ \hat{a_{i3}'} = \dfrac{(1-s_{A_i})a_{i1}' + (1-s_{A_i})a + i2' + (1+2s_{A_i})a_{i3}'}{3} \end{cases} \tag{14}$$

The move operation then changes the position of $\hat{A}_i'$ to the same as A_i^*, and the move ratio $m_{A_i'}$ is defined as [26]:

$$m_{A_i'} = \begin{cases} \frac{3(a_{i1}^* - \hat{a_{i1}'})}{\hat{a_{i2}'} - \hat{a_{i1}'}}, & \text{if } a_{i1}^* > \hat{a_{i1}'} \\ \frac{3(a_{i1}^* - a_{i1}')}{\hat{a_{i3}'} - \hat{a_{i2}'}}, & \text{otherwise} \end{cases} \tag{15}$$

Once all scale and move parameters are determined, the required factors for analogically modifying the intermediate consequent y' are calculated as [5]:

$$s_{y'} = \frac{1}{n} \sum_{i=1}^{n} s_{A'} \tag{16}$$

$$m_{y'} = \frac{1}{n} \sum_{i=1}^{n} m_{A'} \tag{17}$$

The scaled result $\hat{z'}$ of the intermediate consequent y' is obtained as [5]:

$$\begin{cases} \hat{y_1'} = \dfrac{(1+2s_{y'})y_1' + (1-s_{y'})y_2' + (1-s_{y'})y_3'}{3} \\ \hat{y_2'} = \dfrac{(1-s_{y'})y_1' + (1+2s_{y'})y_2' + (1-s_{y'})y_3'}{3} \\ \hat{y_3'} = \dfrac{(1-s_{y'})y_1' + (1-s_{y'})y_2' + (1+2s_{y'})y_3'}{3} \end{cases} \tag{18}$$

Finally, the interpolated consequent y^* is calculated by using the average move ratio as [5]:

$$\begin{cases} y_1^* = \hat{y_1'} + m_{y'}\gamma \\ y_2^* = \hat{y_2'} - 2m_{y'}\gamma \\ y_3^* = \hat{y_3'} + m_{y'}\gamma \end{cases} \tag{19}$$

$$\gamma = \begin{cases} \frac{\hat{y_2'} - \hat{y_1'}}{3}, & \text{if } m_{y'} > 0 \\ \frac{\hat{y_3'} - \hat{y_2'}}{3}, & \text{otherwise} \end{cases} \tag{20}$$

3. Density-based fuzzy rule interpolation

In this section, the density-based fuzzy sparse rule-based inference approach is introduced. A generic framework is first presented, which is then followed by the description of the algorithm for determining the closest rules from the given sparse rule base, and the process of using such an algorithm for TFRI is introduced.

3.1. Framework for sparse rule base interpolation

Without the loss of generality, let $R = \{r_1, r_2, \ldots, r_K\}$ be the original sparse fuzzy rule base with K fuzzy rules, and o^* be an input expressed as:

$$r_k : \text{if } x_1 \text{ is } A_1^k \text{ and } x_2 \text{ is } A_2^k \text{ and } \cdots \text{ and } x_n \text{ is } A_n^k, \text{ then } y \text{ is } Y^k$$
$$o^* : x_1 \text{ is } A_1^* \text{ and } x_2 \text{ is } A_2^* \text{ and } \cdots \text{ and } x_n \text{ is } A_n^*$$

where x_i $(i = 1, 2, \ldots, n)$ denotes the antecedent feature, y denotes the consequent, A_i^k represents the fuzzy value of x_i in the fuzzy rule r_k, and Y^k is the fuzzy value of the consequent y in r_k.

For a dense fuzzy rule base, given R and o^*, a consequent could be reached by firing the matched fuzzy rules. However, if the fuzzy rule base is sparse, and there is no fuzzy rule that matches the input, the fuzzy interpolative inference would be needed as an alternative method to produce an estimated consequent. To this end, the paper presents a general framework for inference with sparse fuzzy rule base by integrating the conventional rule inference approach and a novel density-based TFRI (denoted as D-TFRI) method that searches and selects close and suitable fuzzy rules to construct the intermediate rule for an unmatched input. By combining the advantages of the conventional rule inference approach for matched inputs and the advantages of D-TFRI method for unmatched ones, this integration could obtain more accurate results. For any input, the fuzzy rule base is searched and checked to determine if there is any rules that matched the input at first, if there is at least one rule that matches the input, the result could be obtained by firing the matched rules. Otherwise, the D-TFRI is employed to construct the intermediate rule and estimate the consequent.

For an unmatched input, its close rules are firstly searched and selected using the density-based approach, as detailed in Section 3.2. Then, given several closest rules, the intermediate rule is constructed while considering the distance from the input to the selected rules, as detailed in Section 3.3. Through the density-based close rule selection and the weighted intermediate rule construction, a matched intermediate rule for the input could be obtained, and the scale and move transformation of the conventional TFRI approach is used to estimate the consequent.

3.2. Close rule search and selection

One of the biggest issues in constructing the intermediate rule for unmatched inputs is to determine which existing rules should be selected to construct the intermediate rule. In theory, the relationship of antecedent features between the fuzzy rule and the unmatched input is generally consistent with that of the consequents between the fuzzy rule and the unmatched input, and the consequent of the unmatched input can be estimated by constructing the intermediate rule using the closest rule. However, as the fuzzy rule base is sparse, the density of the fuzzy rules cannot be guaranteed, that is to say, though the closest rule for the unmatched input is found, simply using it to construct the intermediate rule may not be sufficient enough. Therefore, in practice, when constructing the intermediate rule for an unmatched input, the sparse fuzzy rule base is normally searched and sorted based on the distance between the input and the fuzzy rules, and the top k closest rules are selected to construct the intermediate rule, where k is a pre-defined value based on experience. However, selecting the fixed amount of rules to construct the intermediate rule for any unmatched inputs has its limitations: (1) how to determine the value of k remains challenging, some researchers claimed that two rules are sufficient enough for weighted TFRI [5], however, it still needs further investigation and validation; (2) different inputs have different characteristics, using a universal k value may not be effective for all situations. In fact, due to the randomness and unpredictability of the input, there could be cases where one input is very close to one fuzzy rule but not to any other rules, and one input has the similar distance to several rules. In such cases, using a fixed value of k may not be effective and efficient. This requires the introduction of a method that could properly search and select a certain amount of fuzzy rules, not always the same amount, to any unmatched inputs based on their characteristics and relationship with the fuzzy rules.

The basic idea of the density-based searching approach is to find fuzzy rules that are within a certain range of the unmatched input with regard to the antecedent features. As fuzzy rules that are close to the input with regard to the antecedent features generally would be close to the actual consequent of the input, by limiting the distance from the input to the fuzzy rules with a certain range instead of limiting the number of fuzzy rules to be selected, it is ensured that the fuzzy rules are selected based on their *absolute closeness* to the input rather than the *relative closeness* to the input. Reflecting this idea, the search and selection of close rules to the unmatched input are conducted through the following process.

(1) Calculation of the distance from the unmatched input to each rule in the sparse fuzzy rule base with regard to antecedent features.

(2) Identification of the closest rule in the sparse fuzzy rule base to the unmatched input.

(3) Expansion of the selected rules from the closest rule within the given range to the unmatched input.

By doing so, for each unmatched input there are a certain amount of rules that are within the range to be selected, corresponding to *this* specific input, and for different inputs, not only the selected rules might be different, but also the number of selected rules could be different, reflecting the relationship of the antecedent features between the fuzzy rules and the input. The detailed procedure of this process is presented in Algorithm 1.

In this method, for an unmatched input, the rule that is closest to it in the sparse fuzzy rule base is firstly identified, it should be noted that *close* is not necessarily describing the distance, other similarity measures may also apply. In this paper, for simplicity, we use the conventional distance measure presented in Eq. (3) to calculate the distance between each rule and the input. It is also noted that the weights of different antecedent features are not considered, as they are all set to have equal importance, though the attribute weight calculation method could be utilized in the distance calculation process.

Once the closest rule is identified, it is then used to search the rest sparse

Algorithm 1 Close rule search and selection

Input: Sparse rule base $R = \{r_1, r_2, \ldots, r_K\}$, input o^*, selection range ε.
Output: Selected fuzzy rules for intermediate rule construction R_S, distance from the input to the selected fuzzy rules D.

```
Initialize the auxiliary rule base R_A = R
for r_k ∈ R do
    Calculate the distance d(r_k, o*) between r_k and o*
end for
Determine the closest rule r_k with k = arg min_{k=1,2,...,K} d(r_k, o*)
Add r_k to selected fuzzy rules R_S = {R_S} ∪ r_k
Remove r_k from R_A as R_A = R_A \ {r_k}
for r_i ∈ R_A do
    Calculate the distance d(r_i, r_k) between r_i and r_k
    if d(r_i, r_k) < ε then
        Add r_i to selected fuzzy rules R_S = {R_S} ∪ r_i
    end if
    Remove r_i from R_A as R_A = R_A \ {r_i}
end for
Calculate the distance from the input to the rules in R_S
return Selected fuzzy rules R_S, corresponding distance to the input D
```

rule base for other rules that are within the selection range. Similarly, these rules are searched and selected based on the *closeness*, not to the input, but to the selected rules. In this paper, the distance measure is used as:

$$d(r_m, r_k) = \sqrt{\sum_{n=1}^{N} (d(A_i^m, A_i^k)^2} \tag{21}$$

where $d(A_i^m, A_i^k)$ is calculated using the representative value by

$$d(A_i^m, A_i^k) = \frac{|Rep(A_i^m) - Rep(A_i^k)|}{\max_{A_i} - \min_{A_i}} \tag{22}$$

Finally, the distance between the input and each selected rule is obtained for subsequent calculation. Through this density-based approach, it ensures that only rules which not just *close* to the input, but within a certain range are selected, thus enhancing the consistency in the selected rules. Normally, the selection range ε is set to 0.1.

For instance, consider the case shown in Fig 1. Suppose according to prior knowledge, a sparse rule base with 6 rules, i.e., E_1, E_2, E_3, E_4, E_5, and E_6, is constructed. Let E be an input that does not match any rules, and its closest rules are E_5, E_4 and E_3, in descending order, respectively. Clearly, for conventional TFRI methods, as the number of selected rules k is set to 2, only E_5 and E_4 will be selected to construct the intermediate rule, however, this would lead to the ignore of E_3, which has a similar distance to E compared to E_4 and E_5. Moreover, only using E_4 and E_5 to construct the intermediate rule would also lead to the loss of information as both rules have the same values of U_1, whereas the value of U_1 of E_3 is different. Thus, when adopting conventional TFRI methods, that is, selecting a fixed number of closest rules without considering the actual distance and distribution of the rules, certain information could be lost and the interpolation performance would be impacted. On the other hand, when adopting the proposed density-based closest rule selection approach, by defining the selection range, E_5, E_4 and E_3 could all be selected, thus enabling more reliable and accurate interpolation results.

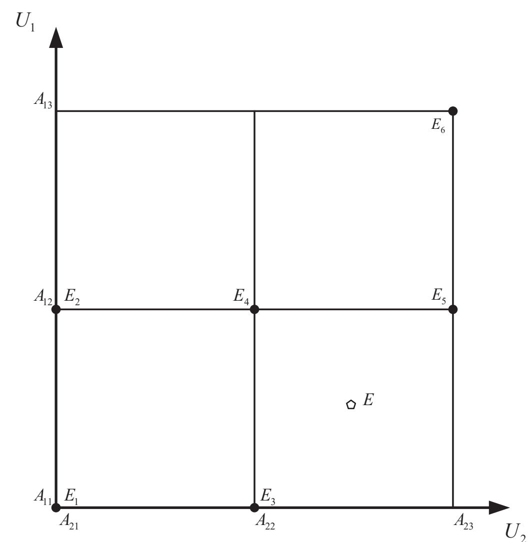


Figure 1: Illustration of close rule selection

3.3. FRI with selected rules

The central idea of TFRI approach is to capture the significance degrees of individual attribute features by the distance and use them to calculate the estimated consequent given an unmatched input. Therefore, it is natural to use rules whose antecedent features are close to the unmatched input for the interpolation. That is to say, the selected rules in the density-based approach are expected to be used to construct the intermediate rule for the unmatched input, which is followed by scale and move transformation to ensure that the antecedent features of the intermediate rule coincide with the corresponding values in the unmatched input. The detailed process of implementing the FRI with selected rules is shown in Algorithm 2.

Algorithm 2 Fuzzy rule interpolation with selected rules

Input: Selected fuzzy rules for intermediate rule construction R_S, distance from the input to the selected fuzzy rules D, input o^*.

Output: Estimated consequent for the unmatched input y^*.

1: Obtain the weight $\omega'_{A_i^k}$ of the ith antecedent feature of the kth selected rule such that

$$\omega'_{A_i^k} = \frac{1}{d(A_i^k, A_i^*) + 1}$$

2: Calculate the intermediate fuzzy term A_i'' over the ith antecedent feature by aggregating the antecedent features of l selected rules using the normalized weight $\omega_{A_i^k} = \omega'_{A_i^k} / \sum_{i=1}^{l} \omega'_{A_i^k}$ as

$$A_i'' = \sum_{k=1}^{l} \omega_{A_i^k} A_i^k$$

3: Compute the antecedent feature of the intermediate rule A_i' based on A_i'' such that $A_i' = A_i^*$ with

$$A_i' = A_i'' + \delta_{A_i} (\max_{A_i} - \min_{A_i})$$

4: Compute the consequent of the intermediate rule by aggregating the consequents of l selected rules with the parameters ω_{y^k} and δ_y

5: Calculate the scale rate $s_{A_i'}$ of A_i' that could transform A_i' to $\hat{A}_i'$ such that it has the same scale as the corresponding antecedent feature in o^* as

$$s_{A_i'} = \frac{a_{i3}^* - a_{i1}^*}{a_{i3}' - a_{i1}'}$$

6: Obtain the move ratio $m_{A_i'}$ that transforms the position of $\hat{A}_i'$ to the same as that of A_i^*

7: Obtain the scale and move parameters of the consequent attribute by aggregating corresponding parameters of all attribute features as

$$s_{y'} = \frac{1}{n}\sum_{i=1}^{n} s_{A'_i}$$

$$m_{y'} = \frac{1}{n}\sum_{i=1}^{n} m_{A'_i}$$

8: Calculate interpolated consequent y^* by applying the scale and move parameters to the intermediate consequent y'

9: **return** y^*

In this process, after obtaining the l close rules to the unmatched input, it is easy to determine the weights of different antecedent features of these rules based on their distance to the input. It is noted that for different antecedent features, different weights could be obtained for the same rule, that is to say, as all the selected rules are not necessarily *the one* to determine the consequent of the input, it is possible and practical to consider these rules with regard to each antecedent feature. As the intermediate rule is constructed by aggregating these rules and through scale and move transformation, having different weights for different antecedent features for computing the antecedent feature of the intermediate would generally have little impact on the consistency.

Once the weights of different rules with regard to the same antecedent feature are obtained and normalized, the corresponding intermediate fuzzy term A''_i could be calculated by aggregating the antecedent features of different rules considering the weights. However, it could be noted that in many cases, A'' does not necessarily have the same representative value as A^*_i as it is merely a weighted combination of all selected rules, as such it cannot be directly used for the intermediate rule, even with scale and move transformation. Thus, A''_i is moved to A'_i such that it has the same representative value as A^*_i. Note that the *move* operation here is conducted through the distance between A''_i and A^*_i, and it is different from the subsequent scale and move transformation as it only deals with the representative value instead of the shape of the fuzzy set. The consequent of the intermediate rule can be computed by aggregating the corresponding consequents of the selected rules using parameters ω_{y^k} and δ_y, such that

$$\begin{aligned}\omega_{y^k} &= \frac{1}{n}\sum_{i=1}^{n}\omega_{A_i^k} \\ \delta_y &= \frac{1}{n}\sum_{i=1}^{n}\delta_{A_i}\end{aligned} \tag{23}$$

Though the constructed intermediate rule has the same representative values of antecedent features as that of the unmatched input, their shape is not necessarily identical. Hence, the scale and move transformations are conducted to the intermediate rule, which aims to ensure that the antecedent features of the intermediate rule coincide with that of the unmatched input. In the proposed method, for each antecedent feature of the intermediate rule, its scale rate $s_{A_i'}$ and move ratio $m_{A'}$ are obtained by using Eqs. (13)-(15). Thus, for the consequent of the intermediate rule, it can also be modified through the scale and move transformations to estimate the consequent of the input, and the scale rate $s_{y'}$ and the move ratio $m_{y'}$ are obtained by averaging the corresponding factors of all antecedent features. On the basis of the scale and move transformations, the interpolated consequent can be obtained by using Eqs. (18)-(20).

4. Experiments

The proposed density-based fuzzy rule interpolation method is applied in this section to fifteen benchmark classification datasets to test its performance. The classification accuracies are compared with conventional TFRI methods.

4.1. Experimental settings

In this case, in order to test the performance of the proposed method, fifteen benchmark classification datasets from UCI machine learning dataset repositories [32] are used, and the details of the datasets are summarized in Table 1.

For the proposed method, triangular membership functions are employed to represent fuzzy values of different antecedent attributes. For simplicity and comparison purposes, a triangular membership function with three partitions is adopted for all datasets after normalizing the antecedent attributes. To

Table 1: Statistics on classification datasets

Dataset	# instances	# attributes	# classes
Banknote	1372	5	2
Bupa	345	6	2
Car	1728	6	4
Diabetes	768	8	2
Ecoli	336	7	8
Glass	214	9	7
Haberman	306	3	2
Iris	150	4	3
Knowledge	403	5	4
Pageblocks	5472	10	5
Seeds	210	7	3
Transfusion	748	4	2
Wdbc	569	30	2
Winequality-red	1599	11	10
Yeast	1484	8	10

conduct the experiments, we run 5 times 10-fold cross-validation for each dataset, that is, 90% data are used as training data to generate the rule base, and 10% remaining data are used as testing data to test the performance of the proposed method. The classical fuzzy rule base generation method [33] is adapted to generate the initial rule base, where 30% of fuzzy rules are purposefully randomly removed from the rule base to obtain a sparse rule base to test the performance of the proposed method. The value of ε is set to 0.1.

For testing data, the fuzzy rules in the rule base are checked first to determine if there exist fuzzy rules that could match the input, if there are matched rules, then the result could be obtained by aggregating the consequents of matched rules. If there are no matched rules for the input in the rule base, the FRI method is employed to search for close fuzzy rules and generate the interpolated result.

4.2. Results

In order to demonstrate the performance of the proposed method, the results of the proposed method are compared with several conventional FRI methods, including α-cuts [34] and TFRI [26], and the results are shown in Table 2 and Fig 2.

Table 2: Average classification accuracy (%) of different FRI methods

Dataset	TFRI		α-cuts		D-TFRI	
	Sparse RB	Full RB	Sparse RB	Full RB	Sparse RB	Full RB
Banknote	69.31	68.93	70.59	71.92	74.55	72.46
Bupa	54.84	52.10	52.63	52.77	58.55	53.89
Car	77.64	78.15	76.62	77.57	80.46	78.93
Diabetes	63.35	67.71	60.09	68.18	72.10	73.51
Ecoli	61.34	63.20	60.84	63.71	66.67	67.19
Glass	57.64	59.21	58.89	60.14	61.90	59.84
Haberman	71.29	72.04	70.32	73.38	71.88	72.51
Iris	92.67	93.33	93.33	94.00	94.00	93.67
Knowledge	73.85	74.47	69.43	70.58	72.33	73.25
Pageblocks	69.77	71.28	71.59	72.68	95.71	99.73
Seeds	75.46	73.50	73.30	75.14	80.48	74.76
Transfusion	62.52	64.45	69.51	66.71	70.27	66.27
Wdbc	92.21	92.91	93.34	93.10	95.77	95.43
Winequality-red	50.28	51.54	52.12	52.03	51.25	53.31
Yeast	39.56	40.28	40.81	41.25	42.46	45.70
Average	67.45	68.21	67.68	68.88	72.56	72.03

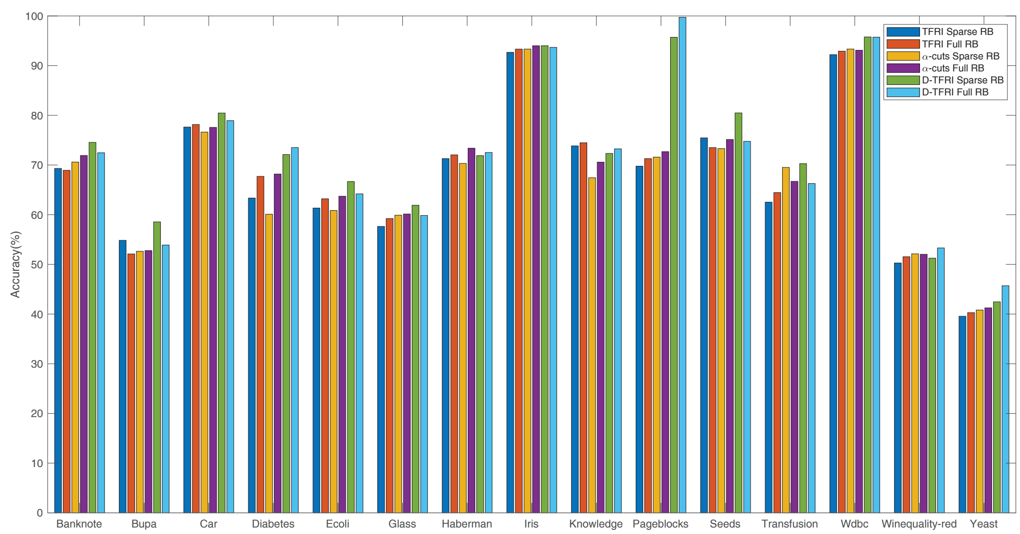


Figure 2: Classification accuracy of different FRI methods

Table 2 illustrates the classification accuracies after averaging the results of the 5 times 10-fold cross-validation of all three methods. For each method, as 30% of fuzzy rules are randomly removed from the original rule base to generate the sparse rule base, in order to thoroughly analyze and compare the performance of the D-TFRI method, the classification accuracies of the interpolated results with sparse rule base are directly compared with those of the full rule base. By comparing the performance of the proposed method on both the sparse rule base and full rule base, it is possible to investigate if the proposed method could achieve better performance with more rules available.

From Table 2 and Fig 2, it can be found that the proposed method could outperform other methods for most datasets, where the classification accuracies of the proposed method for sparse rule base are often about 2%–5% higher than those of other methods. Even for datasets where the performance of the proposed method is not optimal, its classification accuracies are generally satisfactory, and the difference between the performance of the proposed method and that of the optimal method is mostly insignificant. From the comparisons with other FRI methods, it can be said that the proposed method is shown to be an effective way to provide reliable interpolated results for unmatched inputs.

On the other hand, as can be seen from Table 2, by comparing the overall performance of the proposed method with both sparse rule base and full rule base, it is found that though the performance on the full rule base is better than that on the sparse rule base for some datasets, the difference is not very significant. Better performance on the full rule base can be expected as many inputs may be matched by existing rules in the rule base in the first place, thus reducing the burden of interpolation and increasing the classification accuracy by firing the exact fuzzy rule that matches the input. However, though there are more cases that require interpolation on the sparse rule base, by employing the proposed D-TFRI method, the interpolated results are shown to be relatively reliable, as the performance of the proposed method on the sparse rule base is close to that on the full rule base. Moreover, it can be noted from Table 2 that for most datasets, such as Seeds and Wdbc, the performance of the proposed method on the sparse rule base is even better than that on the full rule base, and that can be explained by the fact that more interpolated results are obtained on sparse rule base, which indicates that the proposed method could provide more reliable and accurate results through interpolation than simply firing matched rules.

Table 3 and Fig 3 show the running time of different methods, which can be used to analyze the time complexity of the proposed method. From Table 3 and Fig 3, it can be found that the proposed method has similar or less running time for most datasets, which shows the efficiency of the proposed method. It is also worth noting that for cases where the time complexity of the proposed method is not optimal, the absolute difference is relatively insignificant. Therefore, it can be said that the proposed method could achieve satisfactory time complexity compared with other methods. Combined with the accuracy of the proposed method, it can be concluded that the proposed method could provide an effective and efficient way for fuzzy rule interpolation.

Table 3: Running time (s) of different FRI methods

Dataset	TFRI	α-cuts	D-TFRI
Banknote	0.0071	0.0075	0.0079
Bupa	0.0038	0.0043	0.0039
Car	0.3420	0.3412	0.3390
Diabetes	0.0219	0.0189	0.0228
Ecoli	0.0050	0.0052	0.0049
Glass	0.0023	0.0021	0.0020
Haberman	0.0016	0.0015	0.0017
Iris	0.0011	0.0012	0.0010
Knowledge	0.0082	0.0085	0.0086
Pageblocks	0.2032	0.2548	0.2161
Seeds	0.0029	0.0030	0.0025
Transfusion	0.0036	0.0035	0.0031
Wdbc	0.2370	0.2289	0.2322
Winequality-red	0.1680	0.1687	0.1692
Yeast	0.1481	0.1463	0.1490

Table 4 and Fig 4 show the number of testing data that require interpolation on the sparse rule base and the full rule base, i.e., testing data that are not matched by the fuzzy rules in the rule base. Though there are significantly more interpolated results when using the sparse rule base, the average classification accuracies obtained using the proposed method are higher than those using the full rule base. Thus, with regard to improving classification accuracy, the proposed method is shown to have significant potential.

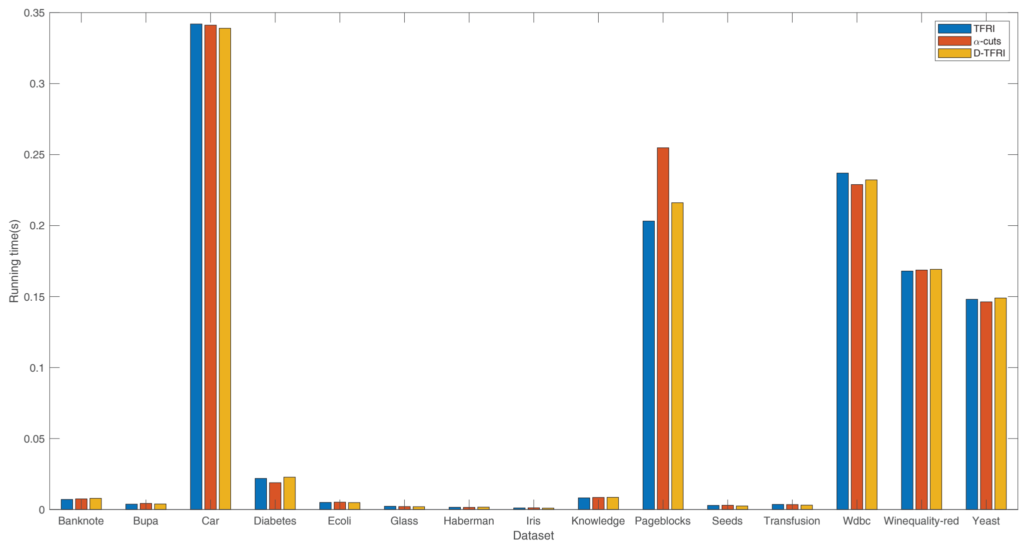


Figure 3: Running time of different FRI methods

Table 4: Average number of interpolated testing data

Dataset	Sparse rule base		Full rule base	
	Interpolated	Total	Interpolated	Total
Banknote	37	137	2	137
Bupa	16	35	3	35
Car	71	173	27	173
Diabetes	33	77	6	77
Ecoli	13	34	3	34
Glass	8	22	3	22
Haberman	12	31	1	31
Iris	7	15	1	15
Knowledge	19	40	7	40
Pageblocks	196	547	5	547
Seeds	10	21	3	21
Transfusion	31	75	2	75
Wdbc	51	57	46	57
Winequality-red	66	160	13	160
Yeast	53	148	10	148

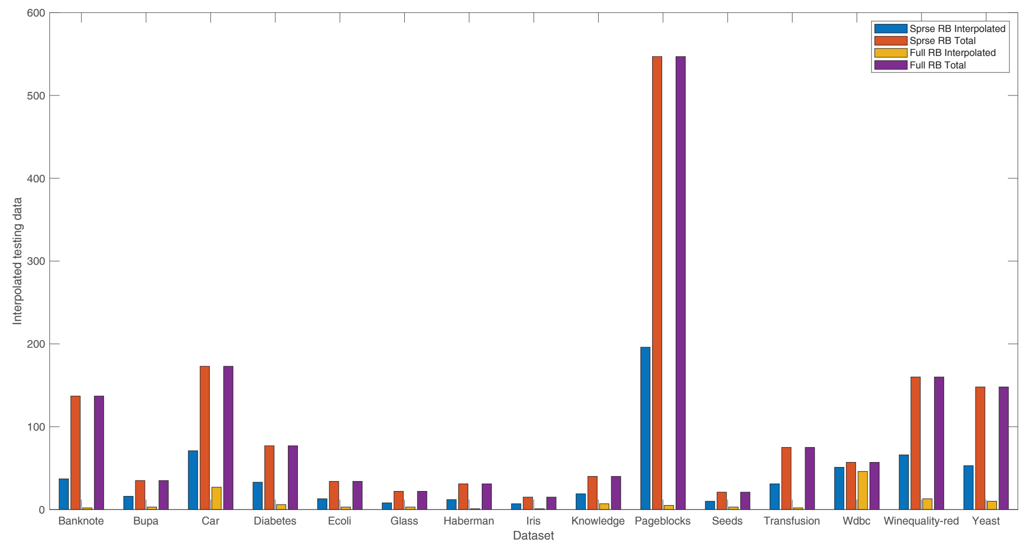


Figure 4: Average number of interpolated testing data

Moreover, the average improvements of the proposed D-TFRI method over TFRI and α-cuts on sparse rule base are computed as 7.58% and 7.21%, respectively, which directly indicates the superior performance of the proposed D-TFRI method. Table 5 shows the statistical pairwise t-test results on the classification accuracies of the proposed D-TFRI method compared with TFRI and α-cuts. From the p-value, it can be found that the differences are significant, thus, it can be further confirmed that the proposed D-TFRI method performs significantly better than other methods. In general, it can be said that by employing the density-based approach, the proposed D-TFRI method could adaptively search and select close rules for interpolation with unmatched inputs, and could provide better results than other FRI methods.

Table 5: Statistical pairwise t-test of classification accuracy for comparison of D-TFRI with other methods

Method	p-value	Hypothesis
T-FRI	0.0076	Rejected
α-cuts	0.0082	Rejected

In order to better demonstrate the effectiveness and efficiency of the proposed method, a nonparametric statistical test is performed, where the proposed method is compared in terms of classification accuracy and running time. The Quade tests are performed in the widely used KEEL [35], and the results are summarized in Tables 6-9.

First, the Quade test is conducted for classification accuracy, and the results are shown in Table 6 and Table 7. For the first step, as listed in Table 6, the ranking of the proposed method is 1.1167, where the Quade statistic is 11.7592 and the p-value is 0.0002, which shows that the proposed method could outperform TFRI and α-cuts. For the post hoc step, the proposed method is compared pairwisely against other methods, where the p-value of α-cuts is 0.0008 and the p-value of TFRI is 0.0024. Therefore, both hypotheses are rejected in each pair-wise comparison, further confirming the proposed method's effectiveness.

Second, the Quade test is conducted for running time, and the results are shown in Table 8 and Table 9. For the first step, the ranking of the proposed method is 1.8750, with the Quade statistic of 0.1312 and the p-value of 0.8775. For the post hoc step, the p-value of TFRI is 0.6227 and the p-value of α-cuts is 0.6806. Hence, both hypotheses are accepted in the pairwise comparison, which indicates that the proposed method has no significant advantages in running time compared with TFRI and α-cuts.

From the statistical test results, it can be concluded that the proposed method could provide a more reliable and effective fuzzy rule interpolation method without increasing time complexity.

Table 6: Average rankings of Quade test of classification accuracy of different methods

Method	Ranking	Quade statistic	P-value
TFRI	2.3750		
α-cuts	2.5083	11.7592	0.0002
D-TFRI	1.1167		

Table 7: Post hoc comparison for $\alpha = 0.05$ of classification accuracy of different methods

i	Method	$z = (R_0 - R_i)/SE$	p
2	α-cuts	3.3534	0.0008
1	TFRI	3.0322	0.0024

Table 8: Average rankings of Quade test of running time of different methods

Method	Ranking	Quade statistic	P-value
TFRI	2.0792		
α-cuts	2.0458	0.1312	0.8775
D-TFRI	1.8750		

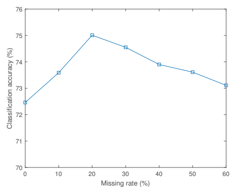
Table 9: Post hoc comparison for $\alpha = 0.05$ of running time of different methods

i	Method	$z = (R_0 - R_i)/SE$	p
2	TFRI	0.4920	0.6227
1	α-cuts	0.4117	0.6806

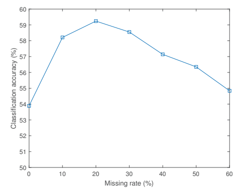
4.3. Sensitivity analysis

To further analyze the effectiveness of the D-TFRI method in interpolation for unmatched inputs, we further conduct some experiments. In Section 4.2, 30% of full rules are randomly removed from the original rule base to generate the sparse rule base, and the classification accuracies of the D-TFRI method on the sparse rule base are compared with those on the full rule base. However, it is worth investigating how would different *level of sparse* of the sparse rule base would affect the performance of the D-TFRI method. In this section, we change the percentage of randomly removed rules of the original rule base from 0% to 60% to further investigate the performance of the proposed D-TFRI method. The results of the sensitivity analysis are shown in Fig 5, where the missing rate of $x\%$ means that we randomly remove $x\%$ of fuzzy rules from the original rule base.

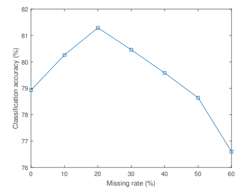
From the results in Fig 5, it can be observed that for most datasets, with the increase of missing rate, i.e., the increase of *level of sparse* of the sparse rule base, the classification accuracy would increase at first, that is, the proposed D-TFRI method performs better on sparse rule base than full rule base. That can be explained by the fact that the interpolated results for the unmatched inputs are normally based on the consequents of several closely related fuzzy rules, thus, by considering several close rules instead of simply firing the one matched rule, it is possible to enhance the consistency and reliability of the results, which may lead to more reliable and accurate results. It is also worth noting that when the missing rate increases to more than 40%, the classification accuracies would decrease, and that is because as more rules are removed from the original rule base, the sparse rule base becomes more



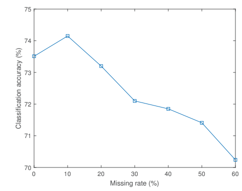
(a) Banknote



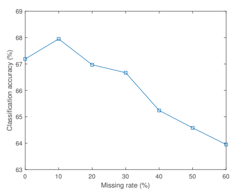
(b) Bupa



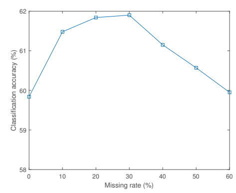
(c) Car



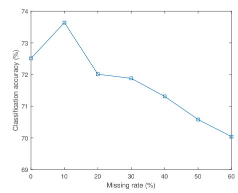
(d) Diabetes



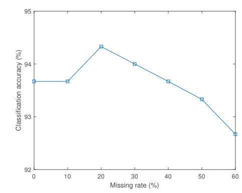
(e) Ecoli



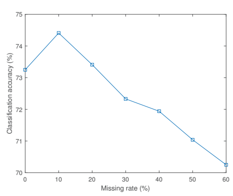
(f) Glass



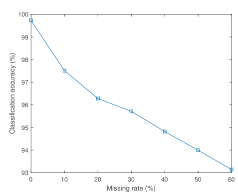
(g) Haberman



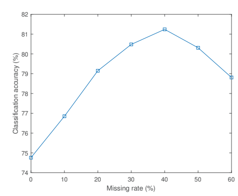
(h) Iris



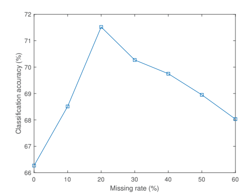
(i) Knowledge



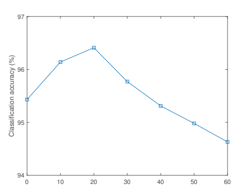
(j) Pageblocks



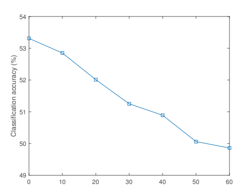
(k) Seeds



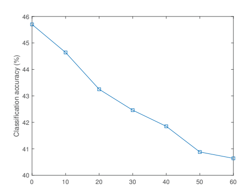
(l) Transfusion



(m) Wdbc



(n) Winequality-red



(o) Yeast

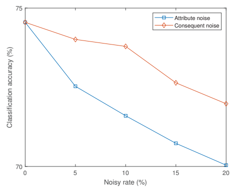
Figure 5: Performance of D-TFRI with different missing rate

and more incomplete, and it may become difficult for some unmatched inputs to find appropriate rules for interpolation as these rules may be removed. However, as shown in Fig 5, for most datasets, the classification accuracies of the proposed D-TFRI method for sparse rule base with 60% missing rate are still satisfactory. Moreover, it can be observed that for some datasets, such as Pageblocks and Yeast, the classification accuracy decreases with the increase of missing rate, which indicates that the original rule base is of high consistency, and removing these rules weakens the performance of the sparse rule base. In general, from the sensitivity analysis, it can be found that the proposed D-TFRI method could provide reliable and satisfactory performance even when there is a high *level of sparse* of the sparse rule base, and is shown to be effective for most classification datasets.

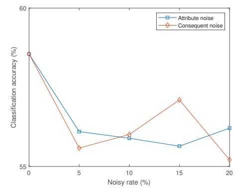
4.4. Robustness analysis

In order to analyze the effectiveness of the proposed D-TFRI method in dealing with noisy data, the robustness analysis is conducted in this section. In this case, we consider two different scenarios of noisy data, attribute noise and consequent noise. Attribute noise means that certain percentages of training data whose attribute values are randomly replaced by values generated between the minimum and maximum values of the corresponding attribute, and consequent noise means that certain percentages of training data whose consequents are replaced by different labels of the dataset. In this section, we change the level of noisy data from 0% to 20%, where the noisy rate of x% means that x% of training data are randomly selected as noisy data. The results of the robustness analysis are shown in Fig 6.

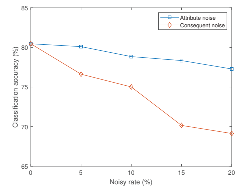
In Fig 6, the blue line denotes the classification accuracy of D-TFRI with attribute noise, and the red line denotes the classification accuracy of D-TFRI with different consequent noise. From Fig 6, it can be found that when there are noisy data in the training dataset, the performance of D-TFRI would generally be affected, as these noisy data could be used to generate the rule base. However, one interesting observation is that for most datasets, the performance of D-TFRI with attribute noise is generally better than those with consequent noise, and that can be explained by the fact that all these datasets have several attributes to be considered, and adding noise to one attribute would have less impact on the performance than adding noise to the consequent. Another interesting observation is that for some datasets, such as Pageblocks, when attribute noise occurs, the performance of the D-TFRI generally remains the same, which could be caused by the inconsistency in the dataset as well as the removal of some noisy rules when obtaining the sparse rule base. In general, from the robustness analysis, it can be found that the proposed method can provide reliable results even when there are noisy data in the training data.



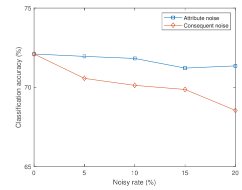
(a) Banknote



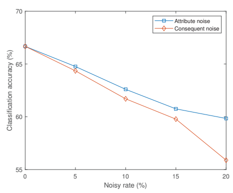
(b) Bupa



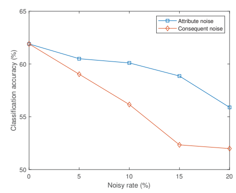
(c) Car



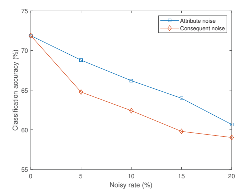
(d) Diabetes



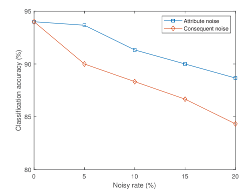
(e) Ecoli



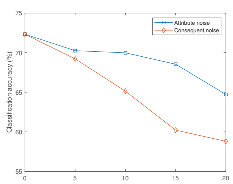
(f) Glass



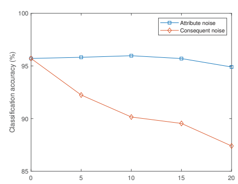
(g) Haberman



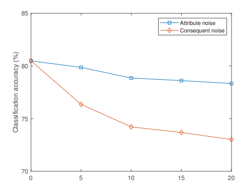
(h) Iris



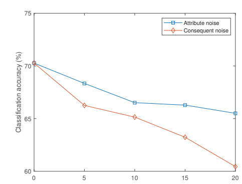
(i) Knowledge



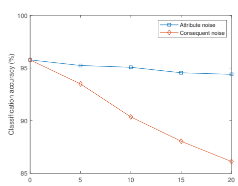
(j) Pageblocks



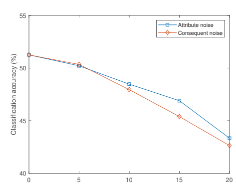
(k) Seeds



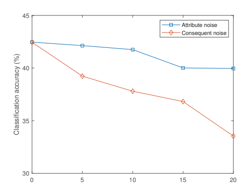
(l) Transfusion



(m) Wdbc



(n) Winequality-red



(o) Yeast

Figure 6: Performance of D-TFRI with different noise rate

4.5. Discussion

By employing the density-based search and selection scheme, the proposed D-TFRI method could benefit in the following aspects. First, by searching fuzzy rules for unmatched inputs using distance measures and only selecting rules that are within a certain radius of the unmatched inputs, it can be guaranteed that the selected rules are rules that are close and similar enough to the unmatched input. Secondly, by weighing different selected rules according to their distance to the unmatched inputs, it can assure that rules closer to the input would play more important roles in determining the intermediate rule.

From the experimental results, it can be noted that, compared with other FRI methods, the proposed method is capable of producing better results, as its classification accuracies are significantly higher than those of other FRI methods. And that is due to the density-based search and selection scheme as the proposed method could ensure the selected rules are rules that are close to the input regardless of the number of rules to be selected, whereas other FRI methods mainly limit the number of rules to be selected, which could lead to the loss of some information. Moreover, both the sensitivity analysis and the robustness analysis further confirm the good performance of the proposed method, as it could provide reliable results when the sparse rule base contains significantly less information or there are a significant amount of noisy data in the training data.

Nevertheless, there are still some limitations to our work. The selection range ε is crucial to the performance of D-TFRI as it determines how many rules would be selected to construct the intermediate rule, thus, how to properly determine the appropriate value of ε remains an important issue. In addition, the search and selection of close rules considers all the attributes in the dataset, which may benefit large-scale and middle-scale datasets, but for some small-scale datasets, the knowledge of the experts might be needed to enhance the performance of the proposed method.

5. Conclusion

In this paper, a density-based fuzzy rule interpolation method is proposed to deal with unmatched inputs for sparse rule base. For unmatched inputs, through the density-based approach, fuzzy rules that are close to the input could be adaptively searched and selected, which could ensure only rules that are within the selection range of the unmatched input could be selected. Then, the selected rules are used to generate the intermediate rule and obtain the interpolated consequent, where the weights of the selected rules are calculated according to their distance to the unmatched input. Fifteen classification benchmarks are tested with 10-fold cross-validations to validate the effectiveness and efficiency of the proposed method.

Experimental results show that the proposed method could provide satisfactory classification accuracies for sparse rule base, and could outperform other FRI methods. Furthermore, through sensitivity analysis, it can be found that the proposed method could provide stable performance when there are a large amount rules missing in the sparse rule base, which could further illustrate the effectiveness of the proposed method. The robustness analysis shows that when noisy data is encountered, the performance of the proposed method is still satisfactory. In general, it can be concluded that the proposed method could effectively provide reliable interpolation results for unmatched inputs with sparse rule base.

In the proposed method, the value of the selection range is determined subjectively, in the future, we will further investigate combining optimization methods with the proposed method to reliably determine its value. Furthermore, we will apply the proposed method to other large-scale problems to expand the application of the proposed method.

References

[1] F. Gao, A. Zhang, W. Bi, J. Ma, A greedy belief rule base generation and learning method for classification problem, Applied Soft Computing 98 (2021) 106856. doi:10.1016/j.asoc.2020.106856.

[2] A. Zhang, F. Gao, M. Yang, W. Bi, Belief rule-based dependence assessment method under interval uncertainty, Quality and Reliability Engineering International 36 (2020) 2459–2477. doi:10.1002/qre.2708.

[3] R. G. G. Caiado, L. F. Scavarda, L. O. Gaviao, P. Ivson, D. L. de Mattos Nascimento, J. A. Garza-Reyes, A fuzzy rule-based industry 4.0 maturity model for operations and supply chain management, International Journal of Production Economics 231 (2021) 107883. doi:10.1016/j.ijpe.2020.107883.

[4] M. Eftekhari, A. Mehrpooya, F. Saberi-Movahed, V. Torra, How fuzzy concepts contribute to machine learning, Springer, 2022.

[5] F. Li, C. Shang, Y. Li, J. Yang, Q. Shen, Interpolation with just two nearest neighboring weighted fuzzy rules, IEEE Transactions on Fuzzy Systems 28 (2020) 2255–2262. doi:10.1109/tfuzz.2019.2928496.

[6] Y.-C. Chang, S.-M. Chen, C.-J. Liau, Fuzzy interpolative reasoning for sparse fuzzy-rule-based systems based on the areas of fuzzy

sets, IEEE Transactions on Fuzzy Systems 16 (2008) 1285–1301. doi:10.1109/TFUZZ.2008.924340.

[7] S.-M. Chen, Y.-C. Chang, J.-S. Pan, Fuzzy rules interpolation for sparse fuzzy rule-based systems based on interval type-2 gaussian fuzzy sets and genetic algorithms, IEEE Transactions on Fuzzy Systems 21 (2013) 412–425. doi:10.1109/TFUZZ.2012.2226942.

[8] Y. Tan, J. Li, M. Wonders, F. Chao, H. P. H. Shum, L. Yang, Towards sparse rule base generation for fuzzy rule interpolation, in: 2016 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), 2016, pp. 110–117. doi:10.1109/FUZZ-IEEE.2016.7737675.

[9] S.-M. Chen, L.-W. Lee, Fuzzy interpolative reasoning for sparse fuzzy rule-based systems based on interval type-2 fuzzy sets, Expert Systems with Applications 38 (2011) 9947–9957. doi:10.1016/j.eswa.2011.02.035.

[10] S.-M. Chen, Z.-J. Chen, Weighted fuzzy interpolative reasoning for sparse fuzzy rule-based systems based on piecewise fuzzy entropies of fuzzy sets, Information Sciences 329 (2016) 503–523. doi:10.1016/j.ins.2015.09.035, special issue on Discovery Science.

[11] S.-M. Chen, Y.-C. Chang, Weighted fuzzy interpolative reasoning for sparse fuzzy rule-based systems, Expert Systems with Applications 38 (2011) 9564–9572. doi:10.1016/j.eswa.2011.01.138.

[12] P. Baranyi, L. Koczy, T. Gedeon, A generalized concept for fuzzy rule interpolation, IEEE Transactions on Fuzzy Systems 12 (2004) 820–837. doi:10.1109/TFUZZ.2004.836085.

[13] Y.-C. Chang, S.-M. Chen, C.-J. Liau, A new fuzzy interpolative reasoning method based on the areas of fuzzy sets, in: 2007 IEEE International Conference on Systems, Man and Cybernetics, 2007, pp. 320–325. doi:10.1109/ICSMC.2007.4413606.

[14] S.-M. Chen, W.-C. Hsin, S.-W. Yang, Y.-C. Chang, Fuzzy interpolative reasoning for sparse fuzzy rule-based systems based on the slopes of fuzzy sets, Expert Systems with Applications 39 (2012) 11961–11969. doi:j.eswa.2012.03.065.

[15] S.-M. Chen, W.-C. Hsin, Weighted fuzzy interpolative reasoning based on the slopes of fuzzy sets and particle swarm optimization techniques, IEEE Transactions on Cybernetics 45 (2015) 1250–1261. doi:10.1109/TCYB.2014.2347956.

[16] Y. Yam, M. L. Wong, P. Baranyi, Interpolation with function space representation of membership functions, IEEE Transactions on Fuzzy Systems 14 (2006) 398–411. doi:10.1109/TFUZZ.2006.876332.

[17] S.-M. Chen, Y.-K. Ko, Fuzzy interpolative reasoning for sparse fuzzy rule-based systems based on α-cuts and transformations techniques, IEEE Transactions on Fuzzy Systems 16 (2008) 1626–1648. doi:10.1109/TFUZZ.2008.2008412.

[18] D. Tikk, P. Baranyi, Comprehensive analysis of a new fuzzy rule interpolation method, IEEE Transactions on Fuzzy Systems 8 (2000) 281–296. doi:10.1109/91.855917.

[19] Z. Huang, Q. Shen, Fuzzy interpolative reasoning via scale and move transformations, IEEE Transactions on Fuzzy Systems 14 (2006) 340–359. doi:10.1109/TFUZZ.2005.859324.

[20] L. Yang, F. Chao, Q. Shen, Generalized adaptive fuzzy rule interpolation, IEEE Transactions on Fuzzy Systems 25 (2017) 839–853. doi:10.1109/TFUZZ.2016.2582526.

[21] S. Jin, R. Diao, C. Quek, Q. Shen, Backward fuzzy rule interpolation, IEEE Transactions on Fuzzy Systems 22 (2014) 1682–1698. doi:10.1109/TFUZZ.2014.2303474.

[22] F. Li, C. Shang, Y. Li, J. Yang, Q. Shen, Approximate reasoning with fuzzy rule interpolation: background and recent advances, Artificial Intelligence Review 54 (2021) 4543–4590. doi:10.1007/s10462-021-10005-3.

[23] F. Li, C. Shang, Y. Li, Q. Shen, Interpretable mammographic mass classification with fuzzy interpolative reasoning, Knowledge-Based Systems 191 (2020) 105279. doi:10.1016/j.knosys.2019.105279.

[24] F. Li, Y. Li, C. Shang, Q. Shen, Fuzzy knowledge-based prediction through weighted rule interpolation, IEEE Transactions on Cybernetics 50 (2020) 4508–4517. doi:10.1109/TCYB.2018.2887340.

[25] F. Li, C. Shang, Y. Li, J. Yang, Q. Shen, Interpolation with just two nearest neighboring weighted fuzzy rules, IEEE Transactions on Fuzzy Systems 28 (2020) 2255–2262. doi:10.1109/TFUZZ.2019.2928496.

[26] F. Li, C. Shang, Y. Li, J. Yang, Q. Shen, Fuzzy rule based interpolative reasoning supported by attribute ranking, IEEE Transactions on Fuzzy Systems 26 (2018) 2758–2773. doi:10.1109/TFUZZ.2018.2812182.

[27] J. Yang, C. Shang, Y. Li, F. Li, Q. Shen, Anfis construction with sparse data via group rule interpolation, IEEE Transactions on Cybernetics 51 (2021) 2773–2786. doi:10.1109/TCYB.2019.2952267.

[28] A. Zhang, F. Gao, M. Yang, W. H. Bi, A new rule reduction and training method for extended belief rule base based on dbscan algorithm, International Journal of Approximate Reasoning 119 (2020) 20–39. doi:10.1016/j.ijar.2019.12.016.

[29] H.-P. Kriegel, P. Krger, J. Sander, A. Zimek, Density-based clustering, WIREs Data Mining and Knowledge Discovery 1 (2011) 231–240. doi:10.1002/widm.30.

[30] H. Li, X. Liu, T. Li, R. Gan, A novel density-based clustering algorithm using nearest neighbor graph, Pattern Recognition 102 (2020) 107206. doi:10.1016/j.patcog.2020.107206.

[31] Y. Ren, N. Wang, M. Li, Z. Xu, Deep density-based image clustering, Knowledge-Based Systems 197 (2020) 105841. doi:10.1016/j.knosys.2020.105841.

[32] D. Dua, C. Graff, UCI machine learning repository, 2017. URL: http://archive.ics.uci.edu/ml.

[33] L.-X. Wang, J. Mendel, Generating fuzzy rules by learning from examples, IEEE Transactions on Systems, Man, and Cybernetics 22 (1992) 1414–1427. doi:10.1109/21.199466.

[34] L. Koczy, K. Hirota, Approximate reasoning by linear rule interpolation and general approximation, International Journal of Approximate Reasoning 9 (1993) 197–225. doi:10.1016/0888-613X(93)90010-B.

[35] I. Triguero, S. Gonzlez, J. M. Moyano, S. Garca, J. Alcal-Fdez, J. Luengo, A. Fernndez, M. J. del Jess, L. Snchez, F. Herrera, Keel 3.0: An open source software for multi-stage analysis in data mining, International Journal of Computational Intelligence Systems 10 (2017) 1238–1249. URL: https://doi.org/10.2991/ijcis.10.1.82. doi:10.2991/ijcis.10.1.82.