Distributed Probabilistic Fuzzy Rule Mining for Clinical Decision Making

Samane Sharif and Mohammad-R. Akbarzadeh-T

Department of Computer Engineering, Center of Excellence on Soft Computing and Intelligent Information Processing, Ferdowsi University of Mashhad, Mashhad, Iran

ABSTRACT

INTRODUCTION: With the growing size, complexity, and distributivity of databases, efficiency and scalability have become highly desirable attributes of data mining algorithms in decision support systems.

OBJECTIVES: This study aims for a computational framework for clinical decision support systems that can handle inconsistent dataset while also being interpretable and scalable.

METHODS: This paper proposes a Distributed Probabilistic Fuzzy Rule Mining (DPFRM) algorithm that extracts probabilistic fuzzy rules from numerical data using a self-organizing multi-agent approach. This agent-based method provides better scalability and fewer rules through agent interactions and rule-sharing.

RESULTS: The performance of the proposed approach is investigated on several UCI medical datasets. The DPFRM is also used for predicting the mortality rate of burn patients. Statistical analysis confirms that the DPFRM significantly improves burn mortality prediction by at least 3%. Also, the training time is improved by 17% if implemented by a parallel computer. However, this speedup decreases with increased distributivity, due to the added communication overhead.

CONCLUSION: The proposed approach can improve the accuracy of decision making by better handling of inconsistencies within the datasets. Furthermore, noise sensitivity analysis demonstrates that the DPFRM deteriorates more robustly as the noise levels increase.

1. Introduction

Today’s datasets are growing rapidly in terms of size and geographical spread. Many modern problems are now categorized as ‘big data’, where the dataset size and distribution are enormous. Traditional approaches to mining are particularly challenged with such datasets in terms of conflicting objectives such as interpretability, accuracy, and the cost of computation vs. communication [1–5]. In other words, a solution that is accurate and computationally efficient may not be desirable as it may also be complex and incomprehensible, and vice versa. When facing conflicting objectives, different application domains may have different preferences. For instance, interpretability gains more importance in medical
domains since the transparency and understanding of information are vital factors for both physicians and patients [6–10]. In this regard, fuzzy systems have the advantage of interpretability due to their linguistic rule structure and proximity to human knowledge, but they cannot learn when standing alone. A common choice to remedy this aspect by combining fuzzy systems with neural networks, such as in [11–21] for mining fuzzy rules from databases. However, these combinations use centralized learning processes that is often inadequate for bigger databases and lead to higher computational complexity and more rules.

In the current literature, there are three main approaches to deal with this rising complexity in large databases [1,22,23]. The first is using a priori knowledge to search small subspaces instead of the whole of space; however, such knowledge is seldom available. The second is using data reduction strategies, which is a common technique in this context. But data reduction only reduces the size of the problem; it does not solve it. The third is using scalable algorithms, but these are algorithmically the most challenging. Distributed architectures and computing platforms often address scalability. The main challenge is creating new distributed algorithms that take the architecture into account and have parallel execution ability [3,15,24–28]. In this process, efficient and more generalizable representation of information is an important aspect that has not received the attention it deserves. For instance, multiple genetic algorithms (GAs) are introduced in [1] that extract classification rules in a parallel and distributed architecture. These GAs, each running on a given node, optimize their respective local populations independently from other nodes. Therefore, they are highly parallelizable. However, architectures for crisp rule extraction are generally inefficient and entail higher communication costs. They also produce large numbers of rules for reaching reasonable accuracy, leading to higher communication costs and less transparency in distributed architectures.

In contrast to the schemes mentioned above, we propose integrating a multi-agent self-organizing architecture with a probabilistic fuzzy logic-based representation for extracting efficient rules from datasets. The probabilistic fuzzy logic-based structure considers both deterministic and stochastic uncertainty in data samples. This structure hence leads to better interpretability and higher accuracy in the classification and decision-making process. The multi-agent design further offers a parallel and fast execution by distributing ‘intelligence and decision making’ among multiple localities and computational agencies [29–36]. Such an approach aims for better scalability in decision-making and faster learning through efficient knowledge representation and distributed information sharing/computing. As demonstrated in this study, the most notable characteristic of this framework is its efficiency and performance in larger datasets.

The rest of this paper is organized as follows. Section 2 provides a more detailed literature survey. The proposed framework is described in Section 3. Experimental results are presented in Section 4. Finally, conclusions are offered in Section 5.

2. A Literature Review on Distributed Fuzzy Rule Mining

Data mining is generally applied to large datasets, often by classifying data into known categories. This process reduces the large datasets into manageable and more transparent pieces of information. Classifier algorithms are hence a valuable component of most data
mining algorithms. Their level of transparency, accuracy, and computational speed determine the performance of the overall data mining approach. Among the existing classifier methods, rule-based classification techniques such as neuro-fuzzy and associative fuzzy classifiers are distinguished for their interpretability. But only a few instances of classifier algorithms address parallel and distributed processing. In [29], interpretability is considered as a major concern for fuzzy rule-based classifiers and a new distributed learning algorithm is proposed for designing accurate and compact fuzzy rule-based classification systems for big data. In [30], an efficient distributed fuzzy associative classifier is designed for big data based on the MapReduce paradigm. In [31], a distributed approach is developed for multi-objective evolutionary generation of fuzzy rule-based classifiers from big data. In [32], a distributed fuzzy classification scheme is proposed to reduce communication cost using knowledge-based partitioning of features. The performance of this method highly depends on the available expert knowledge, especially in high-dimensional problems. In [33], a fuzzy classifier is introduced based on a multi-agent approach. In that work, data is collected from different sources by agents and sent to a central module, where fuzzy rules and a classifier are constructed. But that method suffers from communication overhead. In [34], a parallel framework is presented for CBA (Classification Based on Associations) problem based on associative fuzzy rules. In that research, each processor transfers its local support counts to other processors via broadcast communication. In contrast to the central mode, the computational cost of the parallel mode is higher because of the communication overhead among processors. In [35], an associative fuzzy classifier is designed for distributed environments that uses broadcast communication among processors. Similar to the previous case, however, this method suffers from high communication cost.

In general, most fuzzy rule-based classification methods, either neuro-fuzzy or associative-fuzzy classifiers, have similar behavior in dealing with inconsistent rules, such that among the available inconsistent rules, only one is kept and the others are discarded. For this purpose, neuro-fuzzy classifiers assign a weight to each rule based on the number of references to that rule, keeping only the rule with the maximum weight among the inconsistent rules. Similarly, in associative-fuzzy classifiers, the rules with the maximum support and confidence are selected. It should be noted that in many of these cases, some of the removed rules may have had a significant weight (or support and confidence) and could have been suitable for covering some patterns. Eliminating such rules often reduces the classification accuracy, particularly in problem domains with inherently inconsistent, overlapping, and random patterns.

Inconsistent rules often emerge due to the presence of probabilistic uncertainty in the data. Accordingly, here we invoke probability theory, in addition to fuzzy theory for dealing with such rules. In this regard, the earlier works in [37,38] introduced a fuzzy probabilistic multi-agent system for medical decision-making. In their cooperative system, an interface agent accesses the database and distributes the whole information into different packages and sends the data attributes among agents. Each agent then creates Low and High membership functions for the attributes assigned to it.

In contrast to the above, the dataset is divided among agents in a horizontal manner. In other words, a portion of the dataset with all of the data attributes is assigned to each agent. Each agent then extracts knowledge from its local database and creates membership functions for all of the data attributes. The number of these MFs that are created by agents can hence be different among agents. Data segmentation could be due to the large volumes of
centralized dataset or different geographical locations of distributed datasets. A main point of departure from the work in [37,38] is also the use of probabilistic fuzzy reasoning. In other words, the probabilistic fuzzy rules with probability values in their consequent are suitable for dealing with inconsistent patterns. The concept of probabilistic fuzzy systems is introduced in the literature and various applications of this notion are considered [39–43]. In contrast to the existing distributed classification methods in the literature, the new scheme does not require any prior knowledge for feature space partitioning; instead, agents obtain this knowledge from local data through self-organized learning. The important difference between the present method and the former distributed classifiers is the use of probabilistic fuzzy rules in reaching a transparent ruleset and dealing with the available inconsistency in massive and distributed datasets.

3. The Proposed Distributed Probabilistic Fuzzy Rule Miner (DPFRM)

Multi-agent systems are one of the most significant paradigms in distributed artificial intelligence. A multi-agent system consists of several autonomous agents that can cooperate and sometimes compete to achieve specific goals. This autonomy, reactivity, and social ability of agents can lead to efficient decision-making. In this section, a new multi-agent approach is proposed for extracting probabilistic fuzzy rules from distributed data, which can be widely used in classification and decision-making problems. As a case study, this paper employs the proposed method for designing a clinical decision support system. In this approach, each agent has access to a part of the whole data and extracts its local knowledge from this local data through a self-organized process. Here, data are randomly and evenly distributed among agents. The structure of the proposed system is shown in Figure 1. All agents have a similar internal structure. Each agent uses a simple learning procedure to partition the input space according to its local data. During the learning process, each agent creates a local rule base from its local data. The confidence level of an agent represents the correctness of its acquired knowledge.

Each agent can communicate and exchange information with its neighbors. Figure 2 depicts a system with six agents and few neighborhood relations. For example, since $a_3$ is neighbors with $a_1$ and $a_6$, it benefits from the local rule bases of its neighbors (LFRB1 and LFRB6) and own local rule base (LFRB3). Communication topology is allowed to change during the learning process. There is no coordinator and leader in the proposed system. Rule generation and structure learning are performed by a self-organized process that uses local knowledge and local interactions. At the end of the learning process, each agent will have a local rule base. The local rule bases of good agents are used for decision making, where good agents are those that could earn acceptable confidence levels in the validation phase.

3.1. Agent Learning (Input Space Partitioning and Probabilistic Fuzzy Rule Generation)

In [44], a method called Categorical Learning Induced Partitioning (CLIP) is introduced for learning concepts from training patterns and creating Gaussian membership functions (MFs) based on the available information in the given data. In this paper, agents use CLIP algorithm as a simple learning method. When an agent observes a training pattern with
Figure 1. The structure of the proposed distributed decision-making system based on local probabilistic fuzzy rule base.

The input vector $X = (x_1, \ldots, x_p, \ldots, x_P)$ and output vector $Y = (y_1, \ldots, y_q, \ldots, y_Q)$, it checks all input and output dimensions, where $P$ and $Q$ are the dimensions of the input and output spaces and $Y$ represents the class label of the input pattern. If there is no cluster over any input or output dimension, then a new cluster is created. $A_{jp}$ is the $j$th cluster that is formed over the $p$th axis in the input space. This cluster can be described as

$$A_{jp} = \exp \left( \frac{-(x - x_p(i))^2}{\sigma^2} \right), \quad (1)$$

where $\sigma$ is the width of Gaussian membership function $A_{jp}$, and $x_p(i)$ is its center which is the $p$th entry of the $i$th data record. If there are already-created fuzzy clusters over the $p$th input dimension, the agent computes the similarity value between $x_p(i)$ and the existing clusters. If the maximum similarity value is higher than a predefined threshold $\beta$, it is not required to create any new cluster on dimension $p$. Otherwise, a new cluster is formed with center $x_p(i)$ and width $\sigma$. The same partitioning algorithm is performed over all of the input and
output dimensions. Each agent considers the clusters created by itself and its neighboring agents as

\[ Pt_p(a_i) = \bigcup \{ L Pt_p(a_i), \bigcup_{j \in Nei(a_i)} L Pt_p(a_j) \} \]  

(2)

where \( Pt_p(a_i) \) is a set of MFs that can be seen by agent \( a_i \), \( L Pt_p(a_i) \) is a set of MFs that are locally created by \( a_i \), and \( Nei(a_i) \) represents the neighbors of \( a_i \). If the input data is not covered by any of the existing MFs in \( Pt_p(a_i) \), the agent creates \( A_{jp} \) and updates the left and right clusters of \( A_{jp} \). The width update procedure is described in detail in [44].

For example, Figure 3 depicts a dataset with one feature and three classes. It should be noted that the number of samples in this example is low (about 30), while the ability of DPFRM is better described in massive datasets. This simple example is used for a better illustration of agent interactions. First, the dataset is randomly partitioned between three agents. Each agent creates few MFs over the input space according to its local data through learning. Figure 4 shows the resulting MFs at the 10th step of the learning process. Regarding Figure 3, the created MFs by agents are appropriate, since they can cover the whole decision space. It is noteworthy that each agent could see only 1/3 of the whole data, while the interactions between agents play a crucial role in reaching this result. The communication structure of this step (Step 10) is displayed in Figure 5. When \( a_1 \) received a new sample from the second class \((x = 0.63, y = 2)\), the matching degree between this new data \((x = 0.63)\) and the existing MFs \((A_1 \text{ to } A_5)\) is calculated. It should be noted that, as indicated in Figure 5, \( a_1 \) can use the created MFs by its neighbor \((a_2)\). The maximum matching degree belongs to \( A_5 \), therefore \( a_1 \) uses this MF to form the antecedent part of a rule for inserting it in its LPFRB. With receiving each training tuple \([X, Y]\) by an agent, that agent finds the fuzzy MF with maximum matching degree over all input and output dimensions.
The best matched fuzzy MF in the $p^{th}$ input dimension ($q^{th}$ output dimension) is denoted as $A_{j_p}^i (B^j_l)$. As a result of this competition, a fuzzy rule $R^g_k$ is formulated by:

$$\text{If } x_1 \text{ is } A^i_1 \text{ and } \ldots \text{ and } x_P \text{ is } A^i_P \text{ then } y_1 \text{ is } B^j_1 \text{ and } \ldots \text{ and } y_Q \text{ is } B^j_Q \text{ with probability } P^g_k \text{ and } FC_k$$

where $P^g_k$ is the probability of selecting the $k$th rule in the $g$th group from inconsistent rules.

The methods of calculating probabilities and grouping inconsistent rules are explained later. $FC_k$ is the weight of the $k$th rule which determines the certainty of rule. This weight is different from the probability value. The concept of the certainty factor is described in [45–49]. $FC_k$ is computed locally using:

$$FC_k = \frac{\sum_{x \in d_{s_{ai}}} \min(\mu_{A^i_1}(x_1), \mu_{A^i_2}(x_2), \ldots, \mu_{A^i_P}(x_P), \mu_{B^j_1}(y_1), \ldots, \mu_{B^j_Q}(y_Q))}{\sum_{x \in d_{s_{ai}}} \min(\mu_{A^i_1}(x_1), \mu_{A^i_2}(x_2), \ldots, \mu_{A^i_P}(x_P))}$$

where $d_{s_{ai}}$ is a set of training samples that can be seen by agent $a_i$. Thus, $FC_k$ shows the degree of certainty according to its local knowledge. The created rule is not immediately inserted in an agent’s local rule base since the agent must check for redundancy with the existing rules in its local rule bases and those of its neighbors. In the case of any duplication, the insertion of this new rule is canceled. This means that there exists a fuzzy rule $R^*_{k^*}$ with the same antecedent and consequent parts in LPFRB of an agent or its neighbors. So, rule $R^*_{k^*}$ has already been discovered by the agent $a_i$ or its neighbors. Subsequently, the number of references to $R^*_{k^*}$, denoted by $RF(R_k)$, increases as

$$RF(R^*_{k^*}) \equiv RF(R^*_{k^*}) + 1$$
Figure 4. The MFs created by Agents (A) $a_1$ (B) $a_2$ (C) $a_3$.

Figure 5. A neighborhood relationship in the learning process.

$R_f(R_k)$ is updated during the learning process. The agent $a_i$ can see a set of rules according to:

$$FRB(a_i) = \bigcup \{LFRB(a_i), \bigcup_{j \in \text{Nei}(a_i)} LFRB(a_j)\}$$  \hspace{1cm} (6)

where $FRB(a_i)$ is the union of two sets of rule bases, including the available rules in the local rule base of $a_i$ (i.e. $LFRB(a_i)$) and the existing rules in the local rule bases of $a_i$’s neighbors, (i.e. $\bigcup_{j \in \text{Nei}(a_i)} LFRB(a_j)$).

There are two common types of uncertainty: deterministic and nondeterministic. In complex environments, both of them are available. The combination of fuzzy and probability theories is a powerful tool to simultaneously deal with these two types of uncertainty. As already explained, each fuzzy rule is created by a number of samples. If there are few samples from different classes that overlap in the feature space, they lead to inconsistent rules, i.e. rules with the same antecedents but different consequents. As a novel method for dealing with inconsistent rules, we propose a probabilistic fuzzy approach in this paper. In the proposed technique, each agent checks all of its reachable rules according to (6) and categorizes them based on inconsistency. In other words, the rules with the same antecedent and different consequent parts are placed in the same inconsistent group. For more clarity, consider the previous example. The local probabilistic fuzzy rule bases of $a_1$ and its neighbor ($a_2$) are shown in Table 1.

Before receiving a new pattern ($x = 0.63, y = 2$), there are three rules that can be seen by each of $a_1$ and $a_2$. These rules may be consistent or inconsistent. In the proposed system, inconsistent rules are not removed since they may be suitable for covering some new
Table 1. The rule bases of $\alpha_1$ and $\alpha_2$ in a learning step of the example with one input and one output.

| LPFRB1 | LPFRB2 |
|--------|--------|
| If $x$ is $A_1$ then $y$ is $C_1$ with probability $P^1_1 = 0.55$ | If $x$ is $A_2$ then $y$ is $C_3$ with probability $P^1_1 = 1$ |
| If $x$ is $A_1$ then $y$ is $C_2$ with probability $P^1_2 = 0.45$ | If $x$ is $A_4$ then $y$ is $C_1$ with probability $P^2_2 = 0.8$ |
| If $x$ is $A_5$ then $y$ is $C_3$ with probability $P^2_3 = 1$ | If $x$ is $A_4$ then $y$ is $C_2$ with probability $P^2_3 = 0.2$ |

Data samples. As indicated in Table 1, there are three rules, in $\alpha_1$’s local rule base, that are categorized in two inconsistent groups; because the first and second rules have the same antecedents and different consequents. These two inconsistent rules are placed in the same group with different probability values. Since the third rule is not inconsistent with other rules, it is placed in another group. So, the second group has only one rule. The probability of the third rule is set to one.

To calculate the probability of a rule’s consequent, two factors are used: the total number of references to that rule and the sum of firing strength of that rule during the learning process. Accordingly, the probability value of the $k$th rule in the $g$th group is described by (7):

$$p_{g_k} = \frac{Rf(R_k) + \sum_{x \in d_{s_{i_1}}} \mu_{A_1^k}(x_{1}) \ast \mu_{A_2^k}(x_{2}) \ast \ldots \ast \mu_{A_p^k}(x_{p})}{\sum_{r=1}^{i_g} (Rf(R_r) + \sum_{x \in d_{s_{i_1}}} \mu_{A_1^r}(x_{1}) \ast \mu_{A_2^r}(x_{2}) \ast \ldots \ast \mu_{A_p^r}(x_{p}))},$$

where $R_{g_k}$ is the probability of $k$th rule in the $g$th group. $i_1, \ldots, i_{g}$ determine the indexes of rules, and each of them is a member of $\{1, 2, \ldots, M_{a_i}\}$, where $M_{a_i}$ is the number of rules in the local rule base of $a_i$, and $r_g$ is the number of rules in the $g$th group. Therefore, the probability vector is created by (7). As a representative for the decision-making and inferencing process, one of the rules in each group is selected according to the probability vector of that group. It should be noted that the probability vector only specifies the priority of selecting rules in a group. Finally, which rule is selected in each group depends on the selection mechanism. One representative rule, denoted by $R_{g_k}^*$, is selected for each group which affects the final decision.

In the previous example, with receiving a new data, $A_5$ is selected and then $\alpha_1$ forms a rule such as

$$\text{If } x \text{ is } A_5 \text{ then } y \text{ is } C_2 \text{ with probability } P_4 = 1$$

This rule is inserted in LPFRB$_1$, because there is not any same rule in LPFRB$_1$ and LPFRB$_2$. Since this rule is inconsistent with the third rule, it is placed in the second group. According to (6), the probability vector of the second group changed from $[1, 1]$ to $[0.75, 0.25]$. The reference number of the new rule, which has been recently inserted, is set to 1. After inserting the new rule and updating the probability vector of the second group, LPFRB$_1$ is modified, as depicted in Table 2.
Table 2. The rule base of agent $a_1$ in the example after receiving a new sample.

| LPFRB1                                      |
|---------------------------------------------|
| If $x$ is $A_1$ then $y$ is $C_1$ with probability $P_{11}^1 = 0.55$ |
| If $x$ is $A_1$ then $y$ is $C_2$ with probability $P_{12}^1 = 0.45$ |
| If $x$ is $A_5$ then $y$ is $C_3$ with probability $P_{23}^2 = 0.75$ |
| If $x$ is $A_5$ then $y$ is $C_2$ with probability $P_{24}^2 = 0.25$ |

3.2. The Neighborhood and Communication Structure

Communication and interactions among neighbors play an essential role in the proposed system. Each agent refers to its local rule base and the local rule bases of its neighbors to make decisions. Thus, for each agent, it is essential to determine its neighbors and their number. Selecting a correct metric for determining the neighborhood is very influential on the system output. In this paper, the similarity between the created MFs over the feature space is considered as neighborhood metric. To detect if there is a neighborhood relation between $a_i$ and $a_j$, all of the MFs created by these two agents are checked based on their centers. If the difference between the centers of two MFs is less than a pre-defined threshold, these MFs are similar and one of them is created by $a_i$ and another by $a_j$. Finally, the number of similar MFs can be determined by the percentage of closeness between $a_i$ and $a_j$. In the early stages of the learning process, the neighborhood threshold is small because there is no MF in the feature space. As a result, an agent can use the local knowledge of another agent even if there are small number of similar MFs which have been created by them in the early stages. This threshold is increased during the learning process to reach a pre-determined fixed value. Since agents gradually get more training samples during the learning process, their knowledge about the dataset increases, and accordingly, they need less help from other agents.

In this paper, the neighborhood threshold changes from 0.1 to 0.8. At the beginning of the learning process, the neighborhood threshold is small, and each agent has more neighbors to benefit from their knowledge. But this threshold increases gradually; and consequently, the number of neighbors of each agent decreases. For example, we consider two membership functions MF1 and MF2 according to Figure 6 with centers of 2 and 2.5, respectively. The domain of this feature ranges between 1 and 10. In Figure 7, MF1 and MF2 are similar because their centers are close to each other. If the center of MF2 changes to 4.5, MF1 and MF2 will no longer be similar as shown in Figure 7.

It is important to note that the neighborhood among agents changes in the learning process. Therefore, we require a process to identify each agent’s neighbors. This process can be performed in different periods of time and even in a parallel mode to reduce the computational cost. These changes can lead to the restructuring of the system. In Figure 8, two different structures of a sample system with three agents are shown in two stages of the learning process.
Figure 6. Two similar MFs based on center closeness.

Figure 7. Two different MFs based on center closeness.

3.3. Decision-Making Process

As described in the previous sections, each agent extracts probabilistic fuzzy rules from its local dataset and gets help from its neighbors in this process. Finally, each agent creates a local rule base, but the knowledge of some agents may not be enough. Therefore, it is necessary to distinguish between weak and strong agents. Weak agents cannot participate in the decision-making process. As indicated in Figure 1, each agent has a confidence level that shows the correctness of its local knowledge. At the end of the learning process, a number of samples that have not been already observed by agents are given to them. For this set of samples, the classification error rate is computed for all agents. Each agent with a smaller error gets a higher confidence level. The confidence level is the reverse of classification error.
A good agent is defined as an agent with a confidence level above a pre-determined threshold (9). The LPFRBs of good agents are sent to a central part. The union of these LPFRBs, denoted as PFRB and defined by (10), can be used for decision making and classification problems.

$$\text{Good Agents} = \{i \in \{1, \ldots, n\} \mid CL(a_i) \geq \phi\}$$  \hspace{1cm} (9)

$$PFRB = \bigcup_{i \in \text{Good Agents}} LPFRB_i$$  \hspace{1cm} (10)

The proposed classifier employs probabilistic fuzzy rules, while most of the previous fuzzy classification methods, either centralized or distributed, have been designed based on conventional fuzzy rules.

In (9), the performance of each agent is determined by threshold $\phi$. This value was set to 0.9 in our experiments. $M$ is the total number of rules. These rules are grouped in the global probabilistic fuzzy rule base because of the possible presence of inconsistent rules from good agents. Since some groups may be inconsistent with other groups in GPFRB, it is necessary to merge them and create a probability vector for the newly-merged group. One rule in each group is selected as a representative for inference and decision making. This selection can be done by two mechanisms: DPFRM_S1 and DPFRM_S2. The first selection mechanism is based on maximum probability and selects a rule with the highest probability. This mechanism is recommended for datasets that are collected from one source and usually have fewer inconsistent rules. The second selection mechanism is random selection. In this paper, the roulette wheel is used for this purpose. This mechanism is recommended for datasets that are collected from different sources and usually have more inconsistent rules. Finally, the entire of the selected rules take effect on the final decision. The number of these selected rules is denoted by $M_{opt}$. A fuzzy inference engine with Mamdani minimum implication is applied for decision making, and the output is calculated by the below center
average defuzzifier:

\[ y_q = \frac{\sum_{l=1}^{M_{opt}} w^l \cdot c_q^l}{\sum_{l=1}^{M_{opt}} w^l} \]  

where \( w^l \) is the firing strength of the \( l \)th rule in GPFRB described by (12), \( c_q^l \) is the center of the output MF in the \( l \)th rule, \( y_q \) is the \( q \)th output dimension, \( M_{opt} \) is the number of effective rules (the number of groups) in decision making, \( A_p^l \) is the \( p \)th MF in the antecedent part of the \( l \)th rule, \( x_p \) is the \( p \)th input dimension, and \( FC^l \) determines the certainty degree of the \( l \)th rule.

4. Experimental Results

In this section, the proposed framework is evaluated by five medical decision-making problems as well as two other classification problems. A medical decision-making system can offer diagnosis and treatment recommendations based on the data items that are received from a patient. The purpose of this system is to assist physicians in the decision-making process. In this section, the performance of the proposed system is evaluated by six datasets from UCI [50] and Ahwaz Burn Data (ABD) collected from Iranian hospitals. The classification accuracy on testing data, the size of the final rule base, and the computational time are considered as the performance criteria.

4.1. UCI Datasets

Table 3 shows six selected UCI datasets. 60% of the dataset is used for training and 40% for testing. The training samples are partitioned equally but randomly among the agents. For each dataset, Table 4 indicates the average accuracy and the number of rules during 20 runs with different training and testing sets (selected randomly from that dataset). The number of agents is set to 5 in our experiments; however, this number can be more or less than this value based on the dataset size. Both types of the proposed system (DPFRM_S1 and DPFRM_S2) have the same results on Heart and WBC, which have fewer inconsistencies. But they have different results on PIMA and Thyroid that have more inconsistencies. As demonstrated in Table 4, the average accuracy of DPFRM_S2 is better (about 1%) than DPFRM_S1 for PIMA and Thyroid. It should be noted that the performance of the selection mechanism depends on the given problem. In general, the maximum probability selection method may provide more reliable results in medical diagnosis.

Table 5 displays the computational time during 20 different runs. Figure 9 shows the speedup, i.e. the average ratio of the execution time of distributed mode to central mode, during 20 runs. Table 6 compares the proposed method with a classification method based on fuzzy rules called SaFIN [44]. Furthermore, Table 7 shows the comparison among the results of the proposed method and three classification methods based on association rules, including CBA [51], ARC-AC [52], and OAC [53] on some medical datasets. In contrast to the proposed approach, these tables demonstrate that in the other mentioned methods, if the number of samples in a dataset is increased, the size of the final rule base is increased...
Table 3. Description of datasets.

| Dataset       | Number of samples | Number of classes | Number of features |
|---------------|-------------------|-------------------|--------------------|
| WBC           | 699               | 2                 | 10                 |
| Thyroid       | 215               | 3                 | 5                  |
| Pima Diabetes | 768               | 2                 | 8                  |
| Heart         | 270               | 3                 | 13                 |
| Tic-Tac       | 958               | 2                 | 9                  |
| Iris          | 150               | 3                 | 4                  |

Note: Datasets are selected from the UCI library [50].

Table 4. The average accuracy of the DPFRM over twenty runs on medical datasets.

| Method | Accuracy (mean) | \(M\) (mean) | \(M_{\text{opt}}\) (mean) |
|--------|-----------------|---------------|---------------------------|
| WBC    | 95.95           | 14.25         | 14.25                     |
| Thyroid| 86.09           | 11.45         | 10.50                     |
| Pima   | 73.73           | 26.60         | 25.70                     |
| Heart  | 69.16           | 17.60         | 17.55                     |

\(M\): Number of Rules.

Table 5. The average computational time of the DPFRM over twenty runs on medical datasets.

| Method | Training time (mean ± std) | Testing time (mean) | Communication time (mean ± std) |
|--------|-----------------------------|---------------------|---------------------------------|
| WBC    | 0.13 ± 0.28                 | 0.001               | 0.017 ± 0.004                   |
| Thyroid| 0.034 ± 0.006               | 0.0013              | 0.009 ± 0.0                      |
| Pima   | 0.12 ± 0.012                | 0.002               | 0.019 ± 0.002                   |
| Heart  | 0.035 ± 0.008               | 0.0013              | 0.0088 ± 0                      |

Figure 9. Comparing the DPFRM with its non-probabilistic version: the average speed up over twenty runs on the medical datasets.

majorly. This issue can be obviously observed in PIMA and WBC due to their notable inconsistency level. But DPFRM benefits from the infractions among agents to extract fewer fuzzy rules (i.e. smaller size of final rule base) with high ability in decision making. This means that the proposed scheme can reach a compact fuzzy rule base even on large datasets. Finally, the proposed method is compared with few distributed fuzzy classification methods in Table 8. This table depicts that DPFRM has better average performance in contrast
Table 6. Comparing the DPFRM with SaFIN based on fuzzy rule extraction from data.

| Dataset       | SaFIN       | DPFRM [Proposed] |
|---------------|-------------|-------------------|
|               | Accuracy (mean%) | Number of rules (mean) | Accuracy (mean%) | Number of rules (mean) |
| WBC           | 94.23       | 119               | 95.95           | 14.25               |
| Thyroid       | 88.31       | 22.15             | 87.46           | 10.85               |
| Pima Diabetes | 70.21       | 139               | 74.42           | 29.30               |
| Heart         | 61.42       | 113               | 69.16           | 17.55               |

Notes: The grey cells indicate the better performance. SaFIN [44]: Self-adaptive Fuzzy Inference Network.

Table 7. Comparing the DPFRM with some centralized classification methods based on fuzzy rule extraction from data.

| Method   | Dataset | Accuracy | Number of rules | Accuracy | Number of rules | Accuracy | Number of rules | Accuracy | Number of rules | Accuracy | Number of rules |
|----------|---------|----------|-----------------|----------|-----------------|----------|-----------------|----------|-----------------|----------|-----------------|
| CBA      | WBC     | 95.8     | 49              | 93.4     | 135             | 95.1     | 83              | 95.95    | 14.25           |
|          | Pima    | 72.4     | 45              | 78.0     | 190             | 72.4     | 112             | 74.42    | 29.30           |
|          | Heart   | 81.5     | 52              | 77.2     | 410             | 81.1     | 157             | 69.16    | 17.75           |

Notes: The grey cells indicate the better performance. CBA [51]: Classification Based on Associations. ARC-AC [52]: Association Rule-based Classifier with All Categories. OAC [53]: Optimal Association Classifiers.

Table 8. Comparing the proposed method with a number of distributed classification methods.

| Dataset    | DPFRM-S1 | Distributed Methods |
|------------|----------|---------------------|
|            | Accuracy | Number of Rules     | Accuracy | Number of Rules | Refs.  |
| Thyroid    | 90.00    | 12                  | 90.00    | 15              | [Rod, 11] |
| WBC        | 96.41    | 23                  | 96.27    | 5.40            | [Noj, 9] |
| Pima       | 72.11    | 39.30               | 77.2     | > 30            | [Rag, 12] |
| Iris       | 90.17    | 13.01               | 96.00    | 14              | [Rod, 11] |
| Heart      | 81.62    | 26                  | 81.07    | not reported    | [Rag, 12] |
| Tic-Tac    | 71.75    | 55                  | 66.58    | not reported    | [Van, 8] |
| Average    | 83.68    | 28.05               | 83.47    | 94.8            |         |

Note: The grey cells indicate the better performance.

to the other distributed methods in terms of accuracy and the number of rules for all datasets. Figure 10 illustrates the relation between speed up and problem size. According to this figure, speed up is usually increased by growing the problem size. It reveals that the efficiency of the proposed method is more understandable in large datasets.

It is noteworthy that the proposed method can be employed with different number of agents. The number of agents depends on the problem size such that more agents are selected for larger problems. Figures 11 and 12 display the average results of the proposed method with 1–10 agents in Tic-Tac dataset. Increasing the number of agents up to a specific number improves the performance. After this point, the higher number of agents leads to increased communication overhead and decreased speed up, as shown in Figure 11.
4.2. Estimating the Survival Rate of Burn Patients

Burn plays a vital role in a noteworthy percent of hazardous events. Statistics indicate that nearly 3000 people die each year due to burn-related injuries [54]. Unlike many other medical treatments, treating burns is a long-term process. Here, the ability to correctly predict the treatment outcome for a given patient can be instrumental in helping elevate the rate of survival and the quality of the patient’s life. In other words, the future is not inevitable for a burn patient if the physician can accurately predict the outcome of each treatment.

Since 1949 when it was first addressed, a number of methods have been developed to predict mortality rate in burn patients [55–61]. These models have been applied to different
Table 9. The average results of competing classification methods in ABD during twenty runs.

| Method      | Accuracy | Precision | Recall | Sensitivity | Specificity | Number of Rules |
|-------------|----------|-----------|--------|-------------|-------------|-----------------|
| SaFIN       | 91.0820  | 0.8958    | 0.8188 | 0.8188      | 0.9481      | 78.50           |
| CPFRM_S1    | 86.7385  | 0.9744    | 0.5537 | 0.5537      | 0.9931      | 103.95          |
| CPFRM_S2    | 88.3539  | 0.8689    | 0.7253 | 0.7253      | 0.9471      | 103.95          |
| DPFRM_S1    | 94.4261  | 0.9474    | 0.8538 | 0.8538      | 0.9807      | 66.50           |
| DPFRM_S2    | 95.4087  | 0.9760    | 0.8629 | 0.8629      | 0.9913      | 58.55           |

Note: SaFIN [44]: Self-adaptive Fuzzy Inference Network.

Figure 13. The classification MSE for ABD during ten runs.

populations of indigenous patients. These models have been applied to different populations of indigenous patients. The database under study here contains 4,464 medical records of burn patients admitted to Taleghani Hospital in Ahwaz and few hospitals around Ahwaz in the southwest of Iran [58]. There are notably higher burn statistics in southwest Iran, particularly Ahwaz, due to the hot climate and the occupation of its population. The database includes features such as 1: age (A), 2: sex (S), 3: percent body area (PBA), 4: degree of burn (DB), 5: duration of bed rest in the hospital (DBR); and two outcome classes including 1: live (L), and 2: die (D). This dataset has been collected from different sources while allowing for inconsistent samples, which lead to inconsistent rules. It is hence expected that DPFRM_S2 works better than DPFRM_S1. The following results confirm this claim.

Table 9 illustrates the results of the proposed method, with 50% of data randomly selected for training and 50% for testing. First, the proposed system is implemented in a central mode, i.e. by one agent only, with maximum probability selection (CPFRM_S1) and random selection (CPFRM_S2) mechanisms. In the distributed mode, training samples are distributed equally and randomly among 20 agents. Similar to the central mode, two selection mechanisms based on maximum probability (for DPFRM_S1) and random selection (for DPFRM_S2) are considered. The average error rate (in MSE) and the number of rules of SaFIN, CPFRM_S1, CPFRM_S2, DPFRM_S1, and DPFRM_S2 are compared in Figures 13 and 14. Although SaFIN has high accuracy in predicting the survival of burn patients, the size of the final rule base is larger. It is important to note that the accuracy of the proposed system
Figure 14. The optimal number of rules (i.e. the number of groups) for ABD during ten runs.

Figure 15. The MFs created by one of the agents from the decision-making team on ABD.

is improved, and fewer rules are generated in distributed mode. This could be due to the
efficient interactions among agents. As illustrated here, the capabilities of the distributed
approach DPFRM could be better understood in large datasets.

SaFIN reaches an average of 78.50 fuzzy rules and an accuracy of 91.08%, whereas the
proposed system in central mode, i.e. only one agent, generates an average of 103.95 proba-
bilistic fuzzy rules with an accuracy of 86.74% by CPFRM_S1 and 88.35% by CPFRM_S2. Both
of these methods reach lower accuracy and a higher number of rules in comparison with
SaFIN. But if the proposed system extracts rules in the distributed mode (i.e. with more than
one agent), it reaches a more compact fuzzy rule base. DPFRM_S1 could obtain an average
Table 10. Several extracted probabilistic fuzzy rules that are grouped based on their inconsistency in ABD database.

| Probabilistic fuzzy rule | Group Number |
|--------------------------|-------------|
| If A is Young and S is Male and PBA is High and DB is Medium and DBR is Vshort then D with $P = 0.66$ | 1 |
| If A is Young and S is Male and PBA is High and DB is Medium and DBR is Vshort then L with $P = 0.34$ | 1 |
| If A is Young and S is Female and PBA is Vhigh and DB is Medium and DBR is Vlong then D with $P = 0.89$ | 3 |
| If A is Old and S is Male and PBA is Vlow and DB is Medium and DBR is Vshort then D with $P = 0.11$ | 3 |
| If A is Middle − Aged and S is Male and PBA is Vhigh and DB is Medium and DBR is Vshort then L with $P = 1$ | 4 |
| If A is Old and S is Female and PBA is Vhigh and DB is Medium and DBR is Vshort then L with $P = 1$ | 5 |

Figure 16. The number of inconsistent rules extracted from ABD in the presence of noise.

of 66.50 probabilistic fuzzy rules with an accuracy of 94.43%. Also, DPFRM_S2 generates 58.55 probabilistic fuzzy rules with 95.41% accuracy, on average. Figure 15 shows the MFs that are created by one of the agents in decision-making team over five input features. For instance, few rules from the final rule base are illustrated in Table 10. As expected, these rules are probabilistic fuzzy rules with the probability value in their consequent part.

The presence of noise in data may lead to inconsistency in the rule base. To study this aspect, we add noise to the input data as

$$x_p^{noisy} = x_p (1 + \lambda r), p = 1 \ldots P$$

(13)

where $r$ is a uniformly random number between $−1$ and $1$, and $\lambda$ is a positive real number between $0$ and $1$ which determines the percent of noise. Figure 16 depicts that increasing the percent of noise increases the number of inconsistent rules in the rule base. The accuracy of different classification methods in the presence of noise for ABD is displayed in Figure 17. Figure 18 shows the relation between speed up and problem size. According to this figure, speed up is high for different problem sizes, which demonstrates that the proposed method can have a better performance in comparison with the central mode.

Figures 19 and 20 represent the average results of the proposed method with different number of agents in ABD dataset. As mentioned earlier, increasing the number of agents up to a specific number improves the performance. But a higher number of agents increases the communication overhead, which reduces speed up, as is evident in Figure 19. Hence, the number of agents should be selected properly according to the properties of the given problem.
**Figure 17.** The accuracy of competing methods in the presence of noise for ABD dataset.

**Figure 18.** The speedup vs. problem size in ABD (over twenty runs).

**Figure 19.** The average speedup over twenty runs with 1–10 agents for ABD dataset.

**Figure 20.** The average accuracy over twenty runs with 1–10 agents for ABD dataset.
5. Conclusion

In this paper, a distributed framework is introduced for classification and decision-making that aim for higher transparency and scalability. More specifically, a probabilistic fuzzy rule mining approach is proposed that takes benefit from the interpretability properties of the fuzzy logic paradigm and the distributivity of agent-based designs. The resulting architecture also aims for higher accuracy by better handling rule inconsistencies that stem in deterministic and nondeterministic sources of data uncertainties. More specifically, the proposed approach employs a number of interactive intelligent agents with the same learning process to discover knowledge from a distributed dataset by generating probabilistic fuzzy rules. This probabilistic fuzzy rule structure properly deals with inconsistent data samples since inconsistent rules usually emerge due to probabilistic data uncertainty. This, in turn, leads to improved efficiency and higher accuracy. Additionally, the final linguistic rule base has high interpretability due to its closeness to human knowledge. Finally, scalability is another major characteristic of the proposed system due to its agent-based architecture.

The proposed DPFRM approach is evaluated on six benchmark datasets from the UCI library as well as a larger clinical decision-making problem, i.e. the mortality rate prediction for ABD database. The results clearly show that DPFRM leads to interpretable and accurate classifiers with compact rule bases. More specifically, the number of generated rules by DPFRM is fewer by an average 70.4% in contrast to the other competing distributed methods. As for the ABD clinical decision-making problem, the DPFRM shows a more compact rule base than other classification methods. More specifically, the proposed system reduces the number of rules by about 25.4%, while also increasing the accuracy by about 3%, in comparison with central methods. Comparing the effectiveness of the algorithm on the databases from UCI library and the ABD reveals a strikingly different pattern. In other words, the increased distributivity decreases accuracy in UCI databases, while it has the inverse effect in ABD database. This pattern is attributed to the database size. Since the ABD database is larger, it provides a sufficient number of learning instances for each of its agents, even as they distribute the database amongst themselves. Hence, it is concluded that the distributivity feature of the present work will become more valuable for larger and more redundant databases.

Disclosure statement

No potential conflict of interest was reported by the author(s).

Funding

The author(s) reported there is no funding associated with the work featured in this article.

Notes on contributors

S. Sharif is currently a PhD Candidate of Software Engineering at Ferdowsi University of Mashhad. She received her B.S. and M.S. Degrees in Software Engineering in 2009 and 2013, respectively. Her research interests are in the areas of artificial intelligence, multi agent systems, fuzzy systems, Internet of things and medical informatics.

Mohammad-R. Akbarzadeh-T (Ph.D. University of New Mexico (UNM), 1998) is currently a professor and director of the Center of Excellence on Soft Computing and Intelligent Information Processing
FUZZY INFORMATION AND ENGINEERING

(SCIIP), Electrical Engineering Department, Ferdowsi University of Mashhad. From 1996 to 2003, he was also affiliated with the NASA Center for Autonomous Control Engineering at UNM. In 2006 and 2017, he was also with the Berkeley Initiative on Soft Computing (BISC), UC Berkeley as a visiting scholar. In 2007, he also served as a consulting faculty at the Department of Aerospace and Aeronautic Engineering, Purdue University. Prof. Akbarzadeh is the founding president of the Intelligent Systems Scientific Society of Iran, the founding councilor representing the Iranian Coalition on Soft Computing in IFSA, an IEEE senior member and the founding faculty councilor of the IEEE student branch until 2008. He has received several awards including the National Outstanding Faculty Award in 2019 and the IDB Excellent Leadership Award in 2010. His research interests are in the areas of bio-inspired computing/optimization, fuzzy logic and control, soft computing, multi-agent systems, complex systems, robotics, cognitive sciences, and medical informatics. He has published over 450 peer-reviewed articles in these and related research fields.

References

[1] Rodríguez M, Escalante DM, Peregrín A. Efficient distributed genetic algorithm for rule extraction. Appl Soft Comput. 2011 Jan 1;11(1):733–743.
[2] Zhang Z, Pedrycz W, Huang J. Efficient mining product-based fuzzy association rules through central limit theorem. Appl Soft Comput. 2018 Feb 1;63:235–248.
[3] Park B, Kargupta H. Distributed data mining: algorithms, systems, and applications. The Handbook of Data Mining. Hillsdale, NJ: Lawrence Erlbaum Associates. 2002; p. 341–358.
[4] Li B, Vorobeychik Y, Li M, et al. Scalable iterative classification for sanitizing large-scale datasets. IEEE Trans Knowl Data Eng. 2016 Nov 11;29(3):698–711.
[5] Yang Q, Chen WN, Gu T, et al. A distributed swarm optimizer with adaptive communication for large-scale optimization. IEEE Trans Cybern. 2019 Apr 9;50(7):3393–3408.
[6] Gorzalczany MB, Rudziński F. Interpretable and accurate medical data classification—a multi-objective genetic-fuzzy optimization approach. Expert Syst Appl. 2017 Apr 1;71:26–39.
[7] Gadaras I, Mikhailov L. An interpretable fuzzy rule-based classification methodology for medical diagnosis. Artif Intell Med. 2009 Sep 1;47(1):25–41.
[8] Pach FP, Gyenesei A, Abonyi J. Compact fuzzy association rule-based classifier. Expert Syst Appl. 2008 May 1;34(4):2406–2416.
[9] El-Sappagh S, Alonso JM, Ali F, et al. An ontology-based interpretable fuzzy decision support system for diabetes diagnosis. IEEE Access. 2018 Jul 4;6:37371–37394.
[10] Li Z, Wen G, Xie N. An approach to fuzzy soft sets in decision making based on grey relational analysis and Dempster–Shafer theory of evidence: an application in medical diagnosis. Artif Intell Med. 2015 Jul 1;64(3):161–171.
[11] Kour H, Manhas J, Sharma V. Usage and implementation of neuro-fuzzy systems for classification and prediction in the diagnosis of different types of medical disorders: a decade review. Artif Intell Rev. 2020 Feb 4;53(7):4651–4706.
[12] Van Pham H., Hoang Son L, Minh Tuan L. A proposal of expert system using deep learning neural networks and fuzzy rules for diagnosing Heart disease. In Frontiers in intelligent computing: theory and applications. Singapore: Springer; 2020. p. 189–198. Available from: https://link.springer.com/book/10.1007/978-981-32-9186-7
[13] Huang YP, Singh A, Liu SI, et al. Developing transformed fuzzy neural networks to enhance medical data classification accuracy. Int J Fuzzy Syst. 2018 Aug 1;20(6):1925–1937.
[14] Kishore AH, Jayanthi VE. Neuro-fuzzy based medical decision support system for coronary artery disease diagnosis and risk level prediction. J Comput Theor Nanosci. 2018 Mar 1;15(3):1027–1037.
[15] Salem H, Attiya G, El-Fishawy N. A survey of multi-agent based intelligent decision support system for medical classification problems. Int J Comput Appl. 2015 Jan 1;123(10):20–25.
[16] Kumar PM, Lokesh S, Varatharajan R, et al. Cloud and IoT based disease prediction and diagnosis system for healthcare using fuzzy neural classifier. Future Gener Comput Syst. 2018 Sep 1;86:527–534.
[17] Nilashi M, Ahmadi H, Shahmoradi L, et al. A predictive method for hepatitis disease diagnosis using ensembles of neuro-fuzzy technique. J Infect Public Health. 2019 Jan;12(1):13–20.

[18] Asogbon MG, Samuel OW, Omisore MO, et al. Enhanced neuro-fuzzy system based on genetic algorithm for medical diagnosis. J Med Diagn Meth. 2016 Feb;5(205):2.

[19] Mansourypoor F, Asadi S. Development of a reinforcement learning-based evolutionary fuzzy rule-based system for diabetes diagnosis. Comput Biol Med. 2017 Dec;1:91:337–352.

[20] Amirkhani A, Papageorgiou EI, Mosavi MR, et al. A novel medical decision support system based on fuzzy cognitive maps enhanced by intuitive and learning capabilities for modeling uncertainty. Appl Math Comput. 2018 Nov;15:337:562–582.

[21] Mitrikas NE, Theocharis JB, Petridis V. A multilayered neuro-fuzzy classifier with self-organizing properties. Fuzzy Sets Syst. 2008 Dec;1:159(23):3132–3159.

[22] Tenenbaum AS, Van Steen M. Distributed systems: principles and paradigms. Upper Saddle River (NJ): Prentice-Hall; 2007.

[23] Yang CC, Veltri P. Intelligent healthcare informatics in big data era. Artif Intell Med. 2015 Oct;1:65(2):75–77.

[24] Van Uyen NT, Lee SG, Chung T. A new boosting algorithm for classification on distributed databases. Int J Soft Eng Appl. 2008 Apr;2(2):13–20.

[25] Sharif S, Akbarzadeh-T MR. Distributed fuzzy rule miner (DFRM). In IEEE international conference on fuzzy systems (FUZZ-IEEE); IEEE; 2013 Jul 7. p. 1–7.

[26] Sharif S, Distributed decision making using probabilistic fuzzy rules [MS Thesis]. Ferdowsi University of Mashhad; 2013.

[27] Kovalchuk SV, Krotov E, Smirnov PA, et al. Distributed data-driven platform for urgent decision making in cardiological ambulance control. Fut Gener Comput Syst. 2018 Feb;1:79:144–154.

[28] Mu Y, Liu X, Wang L, et al. A parallel fuzzy rule-base based decision tree in the framework of map-reduce. Pattern Recognit. 2020 Mar;9:103:107326.

[29] Elkano M, Sanz JA, Barrenechea E, et al. CFM-BD: a distributed rule induction algorithm for building compact fuzzy models in big data classification problems. IEEE Trans Fuzzy Syst. 2019 Feb;22(1):163–177.

[30] Segatori A, Bechini A, Ducange P, et al. A distributed fuzzy associative classifier for big data. IEEE Trans Cybern. 2017 Sep;1:48(9):2656–2669.

[31] Ferranti A, Marcelloni F, Segatori A, et al. A distributed approach to multi-objective evolutionary generation of fuzzy rule-based classifiers from big data. Inf Sci (Ny). 2017 Nov;1:415:319–340.

[32] Raghuram B, Gyani J, Hanumanthu B. Fuzzy associative classifier for distributed mining. JCA Proceeding on ICWET; 2012 Feb. Mumbai, India: 1–5.

[33] RaghuRam B, Aghila G. Mobile-agent based distributed fuzzy associative classification rules generation for OLAM. In 2009 international conference on intelligent agent & multi-agent systems, IEEE; 2009 Jul 22. p. 1–5.

[34] Thakur G, Ramesh CJ. A framework for fast classification algorithms. Int J Infor Theo Appl. 2008;40:363–369.

[35] Lu JJ, Xu BW, Zou XF, et al. Parallel mining and application of fuzzy association rules. Front Electr Electron Eng Chin. 2006 Jun;1(1):177–182.

[36] Gan W, Lin JC, Fournier-Viger P, et al. A survey of parallel sequential pattern mining. ACM Trans Know Dis Data (TKDD). 2019 Jun;13(3):1–34.

[37] Tatari F, Akbarzadeh-T MR, Sabahi A. Fuzzy-probabilistic multi agent system for breast cancer risk assessment and insurance premium assignment. J Biomed Inform. 2012 Dec;1:45(6):1021–1034.

[38] Tatari F, Akbarzadeh T MR, Mazouchi M. A self-organized multi agent decision making system based on fuzzy probabilities: the case of aphasia diagnosis. Iranian J Fuzzy Syst. 2014 Dec;30(11):6:21–46.

[39] Hirota K. Concepts of probabilistic sets. Fuzzy Sets Syst. 1981 Jan;1:5(1):31–46.

[40] Czogala E, Zimmermann HJ. Some aspects of synthesis of probabilistic fuzzy controllers. Fuzzy Sets Syst. 1984 Jul;13(2):169–177.

[41] Meghdadi AH, Akbarzadeh-T MR. Probabilistic fuzzy logic and probabilistic fuzzy systems. In 10th IEEE international conference on fuzzy systems (Cat. No. 01CH37297); IEEE; 2001 Dec 2;3:1127–1130.
[42] Meghdadi AH, Akbarzadeh-T MR. Uncertainty modeling through probabilistic fuzzy systems. In Fourth International Symposium on Uncertainty Modeling and Analysis, 2003. ISUMA 2003. 2003 Sep 21 (pp. 56–61). IEEE.

[43] Lee HE, Park KH, Bien ZZ. Iterative fuzzy clustering algorithm with supervision to construct probabilistic fuzzy rule base from numerical data. IEEE Trans Fuzzy Syst. 2008 Feb 7;16(1):263–277.

[44] Tung SW, Quek C, Guan C. SaFIN: a self-adaptive fuzzy inference network. IEEE Trans Neural Netw. 2011 Oct 18;22(12):1928–1940.

[45] Ishibuchi H, Yamamoto T. Fuzzy rule selection by multi-objective genetic local search algorithms and rule evaluation measures in data mining. Fuzzy Sets Syst. 2004 Jan 1;141(1):59–88.

[46] Pelekis N, Theodoulakis B, Kopanakis I, et al. Fuzzy miner: extracting fuzzy rules from numerical patterns. Int J Data Warehouse Mini (UDWM). 2005 Jan 1;1(1):57–81.

[47] Hu YC, Tzeng GH. Elicitation of classification rules by fuzzy data mining. Eng Appl Artif Intell. 2003 Oct 1;16(7-8):709–716.

[48] Cordón O, Del Jesus MJ, Herrera F. A proposal on reasoning methods in fuzzy rule-based classification systems. Int J Approx Reason. 1999 Jan 1;20(1):21–45.

[49] Alcala-Fdez J, Alcala R, Herrera F. A fuzzy association rule-based classification model for high-dimensional problems with genetic rule selection and lateral tuning. IEEE Trans Fuzzy Syst. 2011 Apr 29;19(5):857–872.

[50] Available from: https://archive.ics.uci.edu/ml/ (UCI Machine Learning Repository).

[51] Ma BL, Liu B, Hsu Y. Integrating classification and association rule mining. In Proceedings of the fourth international conference on knowledge discovery and data mining; 1998 Aug 27.

[52] Zaiane OR, Antonie ML, Coman A. Mammography classification by an association rulebased classifier. In Proc. MDK/KDD: Int. Workshop Multimedia Data Mining. 2002 Jul 23. Edmonton, Canada: 62–69.

[53] Hu H, Li J. Using association rules to make rule-based classifiers robust. In Conferences in Research and Practice in Information Technology. Australian Computer Society Inc. 2005;39:47–54.

[55] Estahbanati HK, Bouduhi N. Role of artificial neural networks in prediction of survival of burn patients—a new approach. Burns. 2002 Sep 1;28(6):579–586.

[56] McGwin Jr G, George RL, Cross JM, Rue LW. Improving the ability to predict mortality among burn patients. Burns. 2008 May 1;34(3):320–327.

[57] Patil BM, Joshi RC, Toshniwal D, et al. A new approach: role of data mining in prediction of survival of burn patients. J Med Syst. 2011 Dec 1;35(6):1531–1542.

[58] Azizi AA, Zarei J, Nabovati E, et al. Determining of the factors affecting mortality in burn patients using a decision tree data mining algorithm. J Health Admin. 2014 Jan 10;16(54):34–45.

[59] Rashidi HH, Sen S, Palmieri TL, et al. Early recognition of burn-and trauma-related acute kidney injury: a pilot comparison of machine learning techniques. Sci Rep. 2020 Jan 14;10(1):1–9.

[60] Liu NT, Salinas J. Machine learning in burn care and research: a systematic review of the literature. Burns. 2015 Dec 1;41(8):1636–1641.

[61] Stylianou N, Akbarov A, Kontopantelis E, et al. Mortality risk prediction in burn injury: comparison of logistic regression with machine learning approaches. Burns. 2015 Aug 1;41(5):925–934.