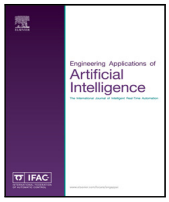
Engineering Applications of Artificial Intelligence 110 (2022) 104697

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai



Enhancing the scalability of fuzzy rough set approximate reduct computation through fuzzy min–max neural network and crisp discernibility relation formulation☆,☆☆

Anil Kumar, P.S.V.S. Sai Prasad *

School of CIS, University of Hyderabad, Hyderabad, India

ARTICLE INFO

Keywords:
Rough set theory
Fuzzy min–max neural network
Fuzzy rough sets
Granular computing
Crisp discernibility matrix
Feature subset selection

ABSTRACT

Fuzzy rough sets (FRS) framework is proven to be useful in computing predictive features in the presence of incompleteness and uncertainty in hybrid systems. However, the existing FRS methods for feature subset selection (reduct computation) are not scalable to large datasets due to higher space and time complexities. Towards increasing the scalability of FRS reduct computation, FMNN-FRS approach is proposed earlier, utilizing fuzzy min–max neural network (FMNN) preprocessing to enable reduct computation in fuzzy hyperbox space instead of object space. FMNN-FRS approach considers fuzzy discernibility matrix (DM) for computation of an approximate reduct. However, it is observed that the space utilization of fuzzy DM limits the scalability of FMNN-FRS. To further increase the scalability of FMNN-FRS method by the reduction in the space complexity, in this work, a novel way of crisp DM construction is proposed from the knowledge derived from FMNN preprocessing (CDM-FMFRS). Extended overlapping criteria, with tolerance parameter, are also designed for arriving at the crisp discernibility relation through fuzzy hyperboxes. The proposed CDM-FMFRS approach computes an approximate reduct using SFS strategy on the generated crisp DM. Empirically, the experimental results established that the classifiability of the induced model from the proposed algorithm is similar or better than FMNN-FRS and other state-of-the-art FRS reduct approaches with a significant reduction in computational time. Results also established better scalability achieved by CDM-FMFRS than FMNN-FRS.

1. Introduction

Feature subset selection is one of the dominant techniques in machine learning and data mining that hugely influences the learning model's performance. Thus it is important to preprocess the data to eliminate irrelevant features that negatively impact the performance of learning models. In 1980s, Pawlak (1982) introduces classical rough set theory (RST), as a mathematical tool useful for feature subset selection (semantic preserving dimensionality reduction) and rule induction in the information/decision systems. RST is primarily applicable to symbolic decision systems (Pawlak, 1991; Yao et al., 2006). However, the induced rules from the feature subset (also called reduct) through the RST approach in the numeric decision system always suffer poor generalizability in classification.

Later, Dubois and Prade (1992, 1990) have extended the RST concept into the fuzzy rough set (FRS) to work on hybrid decision systems. FRS establishes a remarkable role in feature subset selection without any need for additional information. Several aspects of improvement and extension of FRS have been done in search of reliable feature subset selection (Qu et al., 2013; Chen et al., 2007; Jensen et al., 2014; Jensen and Shen, 2004; Jensen and Shen, 2009; Bhatt and Gopal, 2005; Cornelis et al., 2010; Jensen and Shen, 2007; Tsang et al., 2008).

Jensen and Shen (2004) introduce a pioneering work on FRS based feature selection to the domain of web classification. This work shows well in web categorization on web dataset, with promising results. Later, several researchers have done work in the aspects of development and extensions (Bhatt and Gopal, 2005; Jensen and Shen, 2009, 2007) for (Jensen and Shen, 2004). Jensen and Shen (2009) propose a reduct computation approach based on fuzzy similarity-based reduct computation methodology, which attracted considerable attention from researchers and became an effective method for feature subset selection.

☆ The work is supported by DST, Government of India under ICPS project [Grant Number : File No. DST/ICPS/CPS-Individual/2018/579] and UoH-IoE by MHRD, Government of India [Grant Number: F11/9/2019-U3(A)].

☆☆ Authors would like to acknowledge (Zhang et al., 2018 [1,2]) for providing the source code for FRGS and FRSEnt.

* Corresponding author.
E-mail addresses: anilhcu@uohyd.ac.in (A. Kumar), saics@uohyd.ac.in (P.S.V.S.S. Prasad).
URL: http://scis.uohyd.ac.in/~saics/ (P.S.V.S.S. Prasad).

https://doi.org/10.1016/j.engappai.2022.104697
Received 9 November 2020; Received in revised form 8 September 2021; Accepted 23 January 2022
Available online 8 February 2022

Various FRS based algorithms have been developed to perform reduct computation. These methods include information entropy based (Zhang et al., 2018a, 2016) and discernibility matrix based (Dai et al., 2018; Ohrn, 2000; Jensen and Shen, 2009) reduct computation.

Ohrn (2000) propose the concept of a crisp discernibility matrix-based feature selection mechanism in the context of RST. The idea of discernibility matrix (DM) construction establishes a theoretical and logical foundation for reduct computation on symbolic decision systems. Jensen and Shen (2009) introduce a fuzzy discernibility matrix to work on numeric decision systems. Later, several researchers have further continued the design of crisp DM or fuzzy DM for FRS based reduct computation using heuristic-based algorithms (Skowron and Rauszer, 1992; Yang et al., 2018; Chen et al., 2012; Dai et al., 2018; Liu et al., 2020; Tsang et al., 2008).

However, despite these developments and extensions, as mentioned above, the implementation of FRS algorithms still suffered computational overhead to deal with large datasets. The problem is related to the fact that the existing algorithm requires a prior computation of fuzzy similarity matrices/fuzzy discernibility matrix having the space complexity $O(|U|^2|C|)$ where $|C|$ is the size of the attribute space, and $|U|$ is the size of the object space. Hence, these computational overheads become a problem to scale into large decision systems.

Several attempts have been developed in literature in developing a scalable approach for FRS reduct computation. These approaches primarily aim to reduce the requisite space complexity. Some examples in this direction are representative instance-based approaches (Zhang et al., 2018b, 2020), transforming fuzzy DM into crisp DM (Dai et al., 2018) and using granules (Kumar and Prasad, 2020).

The granular computing-based approach replaces the object level computation with information granules level computation that represent objects in the pattern space for enhancing the scalability of FRS methods through reducing space complexity significantly (Lin, 1999). Recently, Anil et al. have proposed a granular computing-based FRS reduct computation approach (FMNN-FRS) to achieve significant computation gain over existing scalable FRS approaches. In Kumar and Prasad (2020), the authors adopt the fuzzy min–max neural network (FMNN) learning model as a preprocessor step that constructs fuzzy hyperboxes (information granules) of the decision system (Simpson, 1992; Reyes-Galaviz and Pedrycz, 2015; Khuat and Gabrys, 2020). The whole motivation of the authors is to define the fuzzy discernibility relation through hyperbox-space instead of object-space to construct the fuzzy DM. Finally, an approximate reduct (Slezak, 1996) is calculated based on constructed fuzzy DM. This way decreased the computation and space overhead of given datasets significantly and achieved better scalability than existing FRS reduct approaches.

In Kumar and Prasad (2020), FMNN-FRS methodology achieves significant better scalability in hyperbox space. Towards the objective of further increasing scalability to large decision systems, our proposed work (CDM-FMFRS) formulates a way to reduce space utilization of fuzzy DM in paving the way to increased scalability. In this paper, we substitute the idea of constructing fuzzy DM (Kumar and Prasad, 2020) to build crisp DM in hyperbox space. The adaptation of crisp DM for FRS reduct computation is motivated from Chen et al. (2012), Yang et al. (2018), Dai et al. (2018), Tsang et al. (2008) that significantly reduces the space utilization. But the formation of crisp DM naturally incurs information loss. The procedure is developed for crisp DM formation in CDM-FMFRS, with the objective of minimizing the inevitable information loss and preserving potential features as part of discernibility matrix entries. Our main contribution can be summarized as follows:

1. We propose a novel approach for the construction of crisp DM over hyperbox space.
2. We enhance the overlapping criteria (Simpson, 1992; Mohammed and Lim, 2015) amidst hyperboxes with three additional rules in achieving crisp DM formulation.
3. Further, enhanced overlapping criteria are enriched with a defined tolerance parameter to facilitate the perseverance of potential attributes in crisp DM entries.
4. We apply SFS based strategy Johnson's algorithm (Ohrn, 2000) on constructed crisp DM to find an approximate reduct.
5. The efficiency and running time of CDM-FMFRS are significantly enhanced in the proposed algorithm.
6. Comprehensive experiments are conducted to assess the impact of information loss in the proposed approach and validate the quality of approximate reduct through extensive comparative analysis with state-of-the-art approaches.

The rest of the paper is designed as follows: In Section 2, we briefly introduce the basics of crisp discernibility matrix, fuzzy discernibility matrix, fuzzy min–max neural network and related work. In Section 3, we briefly describe the functioning of the proposed algorithm CDM-FMFRS. In Section 4, we report a series of experiments and comparative analysis of CDM-FMFRS reduct with other FRS approaches. The future work and conclusion are covered in Section 5 and Section 6.

2. Preliminaries

In this section, we briefly review the basics of crisp discernibility matrix and fuzzy discernibility matrix. Also, we overview the concepts of fuzzy min–max neural network. Lastly, we present a brief introduction of FMNN-FRS paper that is useful for presenting our work.

2.1. Decision relative discernibility matrix approach based on rough set theory

Rough set theory (RST) is a useful technique used to remove the redundant attributes and eliminate the data dependency from a discrete-valued decision system without any prior or extra information (Pawlak, 1982). The theory of RS is based on an indiscernibility equivalence relation.

Let $DT = (U, C \cup \{d\})$ be a symbolic decision system, where U is the non-empty finite set of objects (universe of discourse), C is a non-empty finite set of crisp conditional attributes s.t. $a : U \to V_a$, $\{d\}$ is a decision attribute ($d \notin C$) and V_a is the set of attribute (a) values. Given $B \subseteq C$, a discernibility relation on B is denoted as $DICS(B)$ where a pair of objects u_i and u_j in U belong to $DISC(B)$ if and only if there exists at least one attribute in B having different values for u_i and u_j.

$$DISC(B) = \{(u_i, u_j) \in U \times U \mid \exists\ a \in B,\ a(u_i) \neq a(u_j)\} \tag{1}$$

The decision relative DM (M) of a decision table DT of size $|U| \times |U|$, is a symmetric matrix containing only those entries in which objects belong to different decision classes (Skowron and Rauszer, 1992). Each entry of $M(u_i, u_j)$ contains a subset of attributes in C that distinguish corresponding objects pairs u_i and u_j and have different decision classes, as defined in Eq. (2).

$$M(u_i, u_j) = \begin{cases} \{a \mid a \in C, a(u_i) \neq a(u_j)\}, & \text{if } d(u_i) \neq d(u_j) \\ \emptyset, & \text{otherwise} \end{cases} \tag{2}$$

Given decision system, a set of conditional attributes $A \in C$ is said to be reduct if and only if the following condition holds:

1. $\forall (x, y) \in U \times U : [M(x, y) \neq \emptyset \Rightarrow A \cap M(x, y) \neq \emptyset]$
2. $a \in A,\ \exists (x, y) \in U \times U : [M(x, y) \neq \emptyset \wedge ((A - \{a\}) \cap M(x, y) = \emptyset)]$

For finding reducts, one can create the boolean discernibility function from the discernibility matrix, with attributes as boolean variables. The set of prime applicants from the resulting boolean expression of the discernibility matrix are the set of the attribute reducts.

Even though this operation provides all minimal reducts, it is a costly operation and an NP-Hard problem. This approach is not scalable to large scale decision systems. Researchers have proposed several heuristic algorithms applied to DM to find single reduct, and Johnson Reducer strategy (Ohrn, 2000) is the popular approach.

2.2. Decision relative discernibility matrix approach based on Fuzzy rough sets

Dubois and Prade (1992, 1990) introduce the concept called fuzzy rough sets (FRS) as an extension of RST for hybrid decision systems. RST generalizes to a fuzzy set by allowing that objects can belong to a concept to varying degrees. The notion of FRS appears completely in fuzzy context, having fuzzy sets on a universe U to represent greater indiscernibility, hence, allowing greater flexibility in modeling.

Let $DT = (U, C \cup \{d\})$ be a hybrid decision system (containing categorical and numerical attributes) where U is the non-empty finite set of objects, C is a non-empty finite set of hybrid conditional attributes and $\{d\}$ is a decision attribute and $d \notin C$.

Let us assume R to be the fuzzy relation and decision system DT. An example of the fuzzy similarity relation R_a between two objects $(u_i, u_j) \in U$ w.r.t 'a' attribute is defined as:

$$\mu_{R_a}(u_i, u_j) = exp\left(-\frac{(a(u_i) - a(u_j))^2}{2\sigma_a^2}\right) \tag{3}$$

where μ_{R_a} is the degree to which objects u_i and u_j are similar for feature 'a'. σ_a^2 is the variance for 'a' attribute.

Fuzzy discernibility relation is a generalization of crisp discernibility relation in RST. Fuzzy discernibility relation $DR_a(u_i, u_j)$ is the fuzzy negation of fuzzy similarity relation R_a between a pair of objects u_i and u_j, as given in Eq. (4).

$$\mu_{DR_a}(u_i, u_j) = Neg(\mu_{R_a}(u_i, u_j)) \tag{4}$$

where Neg is a fuzzy negation, and $Neg(\mu_{R_a}(u_i, u_j))$ is the degree of the fuzzy discernibility w.r.t attribute 'a'. Usually, the fuzzy negation is taken as standard negation: $Neg(\mu_{R_a}(u_i, u_j)) = 1 - \mu_{R_a}(u_i, u_j)$.

Fuzzy discernibility relation is represented in the form of a fuzzy DM. Each entry (known as clauses) of fuzzy DM, represented as $M(u_i, u_j)$. $M(u_i, u_j)$ is a set of degrees of discernibility for u_i, u_j over all conditional attributes in C. This fuzzy DM construction considers only those entries which belong to a different class object. Each entry in fuzzy DM $M(u_i, u_j)$, $\forall (u_i, u_j) \in U$ is defined as:

$$M(u_i, u_j) = \begin{cases} \{a_s |\ a \in C, s = Neg(\mu_{R_a}(u_i, u_j))\}, & \text{if } d(u_i) \neq d(u_j) \\ \emptyset, & \text{otherwise} \end{cases} \tag{5}$$

The entry in $M(u_i, u_j)$ shows that objects x and x' are discerning with a certain degree in the range between 0 to 1 w.r.t all attributes. For example, $M(x, x') = \{a_{0.64}, b_{0.35}, c_{0.84}\}$ indicates x and x' have the degree of fuzzy indiscernibility of 0.64 based on attribute 'a', 0.35 based on attribute 'b' and 0.84 based on attribute 'c'.

The degree of satisfaction for $M(u_i, u_j)$ for a given subset of attributes $A \subseteq C$ w.r.t $\{d\}$ is defined as:

$$SAT_{A,\{d\}}(M(u_i, u_j)) = S_{a \in A}\{Neg(\mu_{R_a}(u_i, u_j))\} \tag{6}$$

where S is a t-conorm. Thus, the total satisfiability of entire clauses for subset $A \subseteq C$ is defined as:

$$SAT(A) = \frac{\sum_{u_i, u_j \in U, u_i \neq u_j} SAT_{A,\{d\}}(M(u_i, u_j))}{\sum_{u_i, u_j \in U, u_i \neq u_j} SAT_{C,\{d\}}(M(u_i, u_j))} \tag{7}$$

A minimal subset of conditional attributes $A \subseteq C$ is referred to as a FRS based reduct if and only if the following condition satisfies:

1. $SAT(A) = SAT(C) = 1$
2. $\forall A' \subset A, SAT(A') < SAT(A)$

2.3. Fuzzy min–max neural network

In 1992, Simpson (1992) introduced a single pass dynamic neural network called Fuzzy min–max Neural Network (FMNN) that results in fuzzy hyperboxes of the decision system. Each hyperbox B_j is enclosing the surrounded data of the same class in the n-dimension pattern space, as depicted in Fig. 1(a). Furthermore, it restricts a subregion defined by its minimum (V) and maximum point (W) along with a fuzzy membership function. A hyperbox B_j is represented as:

$$B_j = \{X_h, V_j, W_j, b_j(X_h)\} \quad \forall X \in I^n \tag{8}$$

where $X_h = (x_{h1}, x_{h2}, \ldots, x_{hn})$ is input pattern in n-dimensional space and, $V_j = (v_{j1}, v_{j2}, \ldots, v_{jn})$ and $W_j = (w_{j1}, w_{j2}, \ldots, w_{jn})$ are the corresponding minimum points and maximum points for hyperbox B_j. I^n is the n-dimensional unit pattern space. $b_j(X_h)$ is the fuzzy membership function that describes the membership value of the sample X w.r.t particular jth hyperbox, shown in Eq. (9).

$$b_j(X_h) = \frac{1}{2n}\sum_{i=1}^{n}[max(0, 1 - max(0, \gamma\ min(1, x_{hi} - w_{ji}))) + max(0, 1 - max(0, \gamma\ min(1, v_{ji} - x_{hi})))] \tag{9}$$

where γ provides the sensitive parameter regulating the speed of decreases of the fuzzy membership values, and $0 \leq b_j(X_h) \leq 1$.

FMNN can create a classifier with only one pass through the training data. For each input pattern x, FMNN network is accommodated through three stages for acquiring knowledge, i.e., hyperbox expansion process, overlap test and contraction process. During learning, the network tries to accommodate the input pattern x to one of the existing same class hyperbox that provides the full membership value to one.

Otherwise, the algorithm evaluates the fuzzy membership value of each existing hyperbox of the same decision class. The hyperbox h that having highest membership value takes the preference to accommodate input pattern x through the expansion process if needed.

$$\sum_{i=1}^{n}\left(max(w_{ji}, x_{hi}) - min(v_{ji}, x_{hi})\right) \leq n\theta \tag{10}$$

where theta (θ) is the user-defined parameter with a range of $(0 < \theta < 1)$ for controlling the size of a hyperbox. If the expansion criterion, given in Eq. (10), is satisfied between the input pattern x and hyperbox h, then the minimum and maximum points of hyperbox h are altered to accommodate the input pattern x using Eq. (11) and (12).

$$v_{ji}^{new} = min(v_{ji}^{old}, x_{hi}) \quad \forall i = 1, 2, 3, \ldots, n. \tag{11}$$

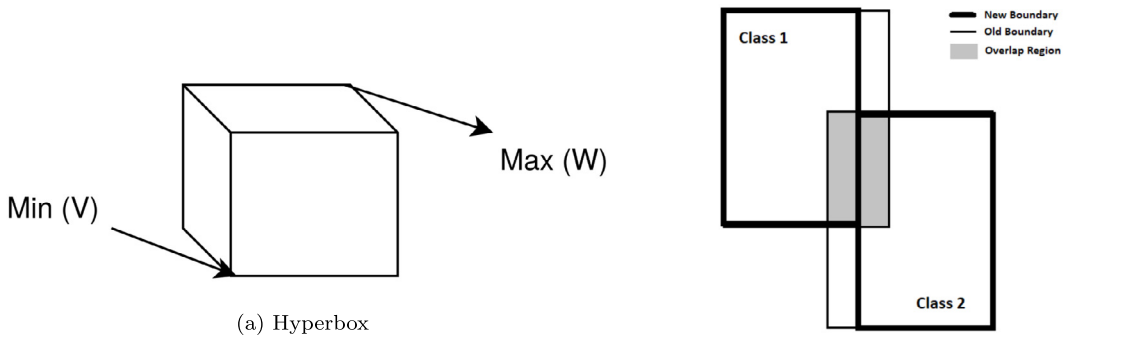
$$w_{ji}^{new} = max(w_{ji}^{old}, x_{hi}) \quad \forall i = 1, 2, 3, \ldots, n. \tag{12}$$

If the existing hyperbox h cannot be expanded using Eq. (10), then a new hyperbox is created to contain x. However, there is a chance that expanded hyperbox h leads to overlapping within adjacent hyperboxes. The overlap test is employed to determine any chance of overlapping between hyperboxes of the different classes. If an overlap is determined between hyperboxes, then the contraction step is applied on the smallest overlap along a dimension (attribute) between hyperboxes to result in non-overlapping.

The overlap and contraction step between hyperboxes h_1 and h_2 of different classes are illustrated in Fig. 1(b). The shaded region showed the overlapping part between hyperboxes. So, the overlap test finds the minimum overlap region along the x-axis dimension. Then, the contraction step altered the hyperboxes along the selected dimension, as shown in bold outline in Fig. 1(b).

2.4. FMNN-FRS

This section introduces the base algorithm (Kumar and Prasad, 2020) of the proposed algorithm, which helps in understanding the idea and formulation of the proposed work. Kumar and Prasad (2020) introduce FMNN-FRS method to compute the FRS reduct computation utilizing FMNN as a preprocessor. In FMNN-FRS, authors restrict FMNN learning till the expansion step for preserving the naturally overlapping regions among hyperboxes of different classes because the overlapping and contraction step in FMNN results loss in valuable representation of discernibility between objects of the overlapped boundary region between hyperboxes of different classes.



(a) Hyperbox

(b) Overlapped region by two hyperboxes

Fig. 1. Depicting of Hyperbox and Overlapping Hyperboxes.

In FMNN-FMFRS, FMNN transforms a numeric decision system into interval-valued decision system (IDS) using FMNN hyperboxes. Objects of IDS correspond to hyperboxes obtained after FMNN training. The benefit of IDS is that it also keeps guiding the boundary knowledge of possible overlapping between intervals to each attribute in the given decision system.

Let IDS $= (HB, C \cup \{d\})$, where $HB = \{h_1, h_2, \ldots, h_k\}$ constitutes a set of hyperboxes where k is the number of hyperboxes and d is the decision class of hyperboxes. Let $[V^h, W^h]$ denotes the minimum and maximum points of hyperbox h. In IDS, the value of a hyperbox $h \in HB$ over an attribute $a \in C$ is represented by the interval v_a^h to w_a^h ($[v_a^h, w_a^h]$), where, v_a^h is component of minimum point V^h and w_a^h is component of maximum point W^h corresponding to the attribute a.

A novel fuzzy discernibility relation is introduced on IDS, which is based on Jaccard similarity measure (Jaccard, 1908) on interval valued data. This paved the way for construction of fuzzy DM on IDS. Each clause or entry in fuzzy DM includes a set of fuzzy discernibility measures between two different classes hyperboxes for all conditional attributes. This fuzzy DM construction is an approximation of fuzzy DM through object-wise. After construction, authors adopt the SFS strategy (SAT measure from Jensen and Shen, 2009) for finding an approximate reduct.

The size of the IDS empirically seen to be much smaller than the original decision system. Hence the experimental analysis (Kumar and Prasad, 2020) has proved that FMNN-FRS could employ on such datasets at which existing object-based FRS approaches failed to compute reduct. The quality of approximate reduct is established in comparative classification experiments. This enhances the scalability of FMNN-FRS without compromising the quality of reduct makes FMNN-FRS useful methodology for reduct computation in a numeric decision system.

3. Proposed CDM-FMFRS reduct algorithm

This section describes the motivation for proposed work, working of proposed algorithm (CDM-FMFRS) and illustration of CDM-FMFRS with toy dataset. Lastly, we present the analysis of time and space complexity of CDM-FMFRS.

3.1. Motivation

The enhanced scalability in FMNN-FRS is because of the construction of fuzzy DM in hyperbox space. But as it will be demonstrated in the experiment section for a very large dataset, the cardinality of hyperbox space itself grows to such large numbers such that the construction of fuzzy DM in hyperbox space itself is not permissible. Our work aims at overcoming this limitation to further increase the scalability of hyperbox space-based FRS reduct computation.

The previous Sections 2.1 and 2.2 introduce the concept of crisp DM and fuzzy DM. Theoretically, the space complexity of both approaches in hyperbox space is $O(|HB|^2|C|)$. An entry of crisp DM is a subset of C, whereas an entry of fuzzy DM is a real-valued array of size C. Adapting characteristic function for the representation of a subset of C and using bitset representation for same, entry of crisp DM requires $|C|$ bits. Assume that a real-valued number is represented in the computer using 'k' bytes. Then it follows that the space utilization of crisp DM is $\frac{1}{8\times k}$ of space utilization of fuzzy DM. Hence, the reduction in space utilization in crisp DM construction is highly significant. Therefore in our work, we adapt crisp DM formation for IDS.

However, the construction of crisp DM for a numerical decision system involves information loss. In the literature, it is arrived at by discretization of numerical decision system or application of threshold fuzzy discernibility value (Hu et al., 2007; Jensen et al., 2014). Either way, a lot of information in fuzzy DM is lost in the conversion of crisp DM. So it becomes imperative to arrive at a crisp DM formulation for availing of space reduction using a methodology aiming at lessening the information loss. With this objective, the formation of crisp DM with FMNN as a preprocessor is given in the next Section 3.2.2.

3.2. Proposed work

This section presents the three-step process for the proposed work CDM-FMFRS reduct computation using FMNN preprocessing step. These steps are FMNN learning, crisp DM construction and reduct computation based on crisp DM.

3.2.1. FMNN learning model

In the proposed CDM-FMFRS, the construction of IDS based on fuzzy hyperboxes is done as per the procedure given and described in Section 2.4.

Illustration for FMNN learning model

Let DT be a toy decision system (10 objects (U), 2 conditional attributes ($length$ and $width$), 3 decision classes (Red, Green and Blue)), as shown in Table 1. By applying FMNN and considering parameters $\theta = 0.3$ and $\gamma = 4$, three hyperboxes $\{h_1, h_2, h_3\}$ are created and visually depicted in Fig. 2. Fig. 2 depicts the 2-D visualization of FMNN hyperboxes between $length$ and $width$ attributes for DT. a and b represent $length$ and $width$ attributes in crisp DM. HB is equal to $\{h_1, h_2, h_3\}$.

Table 2 represents the converted IDS form from Table 1 by transforming object space to hyperbox space.

3.2.2. FMNN preprocessor based crisp discernibility matrix

Here, we provide the procedure for formulation of crisp DM on IDS. This is based on discernibility between two hyperboxes. Each entry in crisp DM $M(h_i, h_j)$ is obtained from a pair of hyperboxes h_i and h_j of different classes. Each clause contains a set of attributes that have non-overlapping or allowable proportion (user defined parameter θ_1) of overlapping intervals between hyperboxes h_i and h_j. Each clause $M(h_i, h_j)$ is defined as:

$$M(h_i,h_j) = \{a \mid a \in C \wedge \ OverlapInDim(h_i,h_j,a) == False \ \vee \ (OverlapInDim(h_i,h_j,a) == True \wedge propoverlap < \theta_1)\} \tag{13}$$

The $OverlapInDim(h_i, h_j, a)$ performs the overlap test between hyperboxes h_i and h_j along 'a' dimension. Simpson (1992) introduces the four conditions to check the overlapping along a particular dimension in FMNN model. In Mohammed and Lim (2015), authors extend the conditions for overlapping cases when min point and max point coincide at the considered dimension. We have further introduced three more conditions for accommodating overlapping in the case at least one of the hyperbox is a point hyperbox. The following are the eleven conditions over which overlapping status is determined and can be considered as a complete set of conditions for checking overlap.

$$\begin{aligned}
&case\ 1: v_a^{h_i} == w_a^{h_i} \ and \ v_a^{hj} == w_a^{hj} \ and \ v_a^{h_i} == v_a^{hj} \\
&\quad overlapdim = True \\
&case\ 2: v_a^{h_i} == w_a^{h_i} \ and \ v_a^{hj} \neq w_a^{hj} \\
&\quad if\ (v_a^{hj} \leq v_a^{h_i} \ and \ v_a^{h_i} \leq w_a^{hj}) \\
&\quad\quad overlapdim = True \\
&case\ 3: v_a^{h_i} \neq w_a^{h_i} \ and \ v_a^{hj} == w_a^{hj} \\
&\quad if\ (v_a^{h_i} \leq v_a^{hj} \ and \ v_a^{hj} \leq w_a^{h_i}) \\
&\quad\quad overlapdim = True \\
&case\ 4: v_a^{h_i} < v_a^{hj} < w_a^{h_i} < w_a^{hj} \\
&\quad o_p = w_a^{h_i} - v_a^{hj} \\
&\quad overlapdim = True \\
&case\ 5: v_a^{hj} < v_a^{h_i} < w_a^{hj} < w_a^{h_i} \\
&\quad o_p = w_a^{hj} - v_a^{h_i} \\
&\quad overlapdim = True \\
&case\ 6: v_a^{h_i} < v_a^{hj} < w_a^{hj} < w_a^{h_i} \\
&\quad o_p = w_a^{hj} - v_a^{hj} \\
&\quad overlapdim = True \\
&case\ 7: v_a^{hj} < v_a^{h_i} < w_a^{h_i} < w_a^{hj} \\
&\quad o_p = w_a^{h_i} - v_a^{h_i} \\
&\quad overlapdim = True \\
&case\ 8: v_a^{h_i} = v_a^{hj} < w_a^{h_i} < w_a^{hj} \\
&\quad o_p = w_a^{h_i} - v_a^{hj} \\
&\quad overlapdim = True \\
&case\ 9: v_a^{hj} < v_a^{h_i} < w_a^{hj} = w_a^{h_i} \\
&\quad o_p = w_a^{hj} - v_a^{h_i} \\
&\quad overlapdim = True \\
&case\ 10: v_a^{h_i} = v_a^{hj} < w_a^{hj} < w_a^{h_i} \\
&\quad o_p = w_a^{hj} - v_a^{hj} \\
&\quad overlapdim = True \\
&case\ 11: v_a^{hj} < v_a^{h_i} < w_a^{h_i} = w_a^{hj} \\
&\quad o_p = w_a^{h_i} - v_a^{h_i} \\
&\quad overlapdim = True
\end{aligned} \tag{14}$$

Table 1
Toy decision system DT.

U	Length (a)	Width (b)	Class
u_1	0.0508	0	Red
u_2	0.2186	0.2546	Red
u_3	0.1786	0.1232	Red
u_4	0.1321	0.205	Red
u_5	0.6023	0.5417	Green
u_6	0.7132	0.7532	Green
u_7	0.6745	0.7038	Blue
u_8	0.8644	0.9346	Blue
u_9	0.6437	0.6752	Green
u_{10}	0.7132	0.8352	Blue

Table 2
Interval valued decision system based on DT.

Hyperboxes	Length (a)	Width (b)	Class
h_1	[0.0508,0.2186]	[0.0000,0.2546]	Red
h_2	[0.6023,0.7132]	[0.5417,0.7532]	Green
h_3	[0.6745,0.8644]	[0.7083,0.9346]	Blue

Cases from first to third correspond to newly introduced overlapping conditions for point hyperboxes, cases 4th to 7th correspond to overlapping conditions in FMNN (Simpson, 1992), and the remaining cases are the additional conditions introduced in EFMNN (Mohammed and Lim, 2015). In each overlap step, we are adding the partial overlapping check (Eq. (15)) based on the proportion between hyperboxes.

The importance of considering partial overlapping steps between hyperboxes lessens the imposition of rigid rules such as sufficient separability between the hyperboxes along the chosen dimensions that can result in significant information loss and possibly a sparse DM. Even if two hyperboxes have a slight overlap in a dimension, then there is a sufficient chance that the attribute is discerning most of the objects of one hyperbox from that of another hyperbox. Preserving such discernible attributes in the crisp DM formation is very important in minimizing the information loss in crisp DM formation. Hence the following properties are arrived at for deciding when an attribute becomes a discernible attribute.

1. An attribute is considered as discerning if it is a non-overlapping dimension.
2. An attribute is considered as discerning if it is an overlapping dimension, but the proportion of overlapping is tolerable based on the user-defined parameter θ_1 $(0 < \theta_1 \leq 1)$.

The partial overlapping check based on proportions is irrelevant to point hyperboxes. In all the other cases 4th to 11th, the proportionality of overlap (propoverlap) is determined as follows:

$$propoverlap = max(\frac{o_p}{(w_a^{h_i} - v_a^{h_i})}, \frac{o_p}{(w_a^{h_j} - v_a^{h_j})}) \tag{15}$$

The amount of overlapping existing in each case is given by o_p. $propoverlap$ gives the maximum of proportionality of overlap in both hyperboxes, and it should be lesser than given θ_1 for an attribute to be included in discernibility matrix entry.

Algorithm 1 presents the structure for computing the crisp DM based on IDS. In Algorithm 1, for every pair of hyperboxes of different classes, an entry $M(h_i, h_j)$ is created by considering only those attributes over which no overlapping exists or permissible partial overlapping exists.

In this paper, crisp DM construction through fuzzy hyperboxes is an approximation of crisp DM based on object space. Therefore, the reduct often computed through crisp DM is a subreduct of the exact reduct, and hence is an approximate reduct for the original decision system.

The advantage of the proposed approach is that the discernibility entry preserves those important attributes which have the potential to discern most of the pair of objects from both hyperboxes. Hence, attributes with higher discerning power are retained in M, thus paving the way for the construction of approximate reduct containing useful attributes.

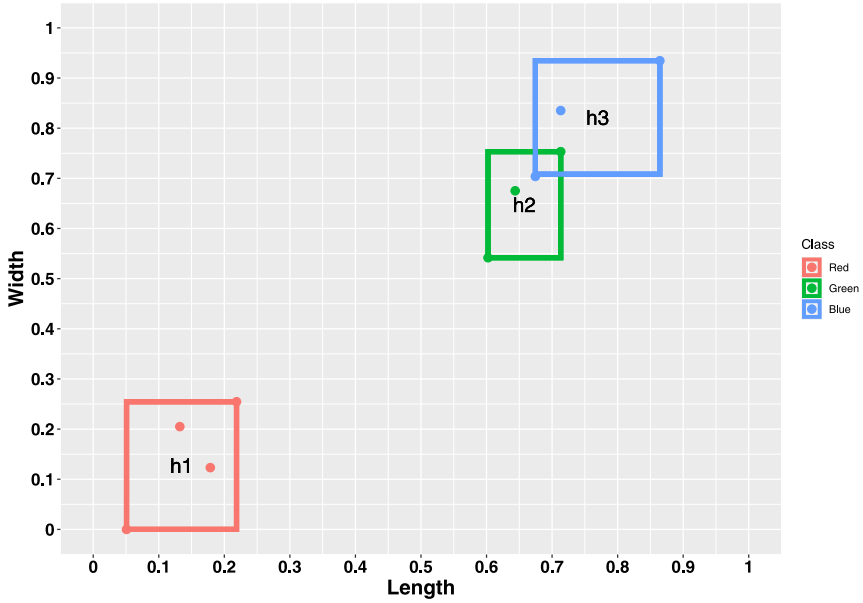


Fig. 2. Hyperboxes created through FMNN learning process for DT.

Algorithm 1: Creating discernibility matrix

Input : HB: set of hyperboxes, θ_1: user-defined tolerance parameter, C: set of conditional attributes
Output: M: Crisp Discernibility Matrix.

```
for every h_i in |HB| do
    for every h_j in |HB| do
        Compute M(h_i, h_j) for i^th hyperbox with each j^th hyperbox of different class labels;
        if d(h_i) ≠ d(h_j) then
            for each a in C do
                if OverlapInDim(h_i, h_j, a) == False then
                    add(M(h_i, h_j),a);
                end
                propoverlap = max( o_p / (w_a^{h_i} − v_a^{h_i}) , o_p / (w_a^{h_j} − v_a^{h_j}) ) ;
                if OverlapInDim(h_i, h_j, a) == True and propoverlap < θ_1 then
                    add(M(h_i, h_j),a);
                end
            end
        end
    end
end
Return M
```

Illustration for discernibility matrix construction

After constructing a IDS on a toy dataset as illustrated in given Section 3.2.1, a crisp discernibility matrix (M) is constructed corresponding to three hyperboxes $\{h_1, h_2$ and $h_3\}$ of different classes through Algorithm 1, as shown in Table 3.

Based on Algorithm 1 and shown in Fig. 2 and given parameter $\theta_1 = 0.25$, h_1 and h_2 hyperboxes are not overlapping in 2-D due to not satisfied any overlapping criteria given in Eq. (14), so their corresponding entries (h_1, h_2) in M are $\{a, b\}$. Similar results for comparing between h_1 and h_3 is obtained. For comparison between h_2 and h_3, their corresponding DM entry (h_2, h_3) is $\{b\}$ based on case no. 4 in Eq. (14). Because, based on Eq. (15), the proportion values for $length(a)$ is 0.34 which is greater than θ_1 and $propoverlap$ (in Eq. (15)) for $width(b)$ attributes is 0.20 which smaller than θ_1. Hence, entry for (h_2, h_3) in M is $\{b\}$. The resulting DM (M) is shown in Table 3.

Table 3
Crisp discernibility matrix M based on IDS.

	h_1	h_2	h_3
h_1	∅	$\{a, b\}$	$\{a, b\}$
h_2	$\{a, b\}$	∅	$\{b\}$
h_3	$\{a, b\}$	$\{b\}$	∅

3.2.3. Reduct computation using johnson's reducer

In the last phase, Johnson's algorithm (Ohrn, 2000) is used to find a single reduct through crisp DM. Johnson's algorithm is given in Algorithm 2.

Johnson's algorithm is a greedy hill-climbing algorithm based on a maximal discernibility (MD) heuristic. MD heuristic is an estimation of the discernibility power of an attribute and is equal to the number of discernibility matrix entries containing the attribute. Johnson's algorithm is a sequential forward selection strategy based algorithm and starts with an empty set reduct. In each iteration, MD heuristic computes for each attribute not already included in reduct. The best discerning attribute is included in the reduct, and the corresponding clauses containing the attribute are removed before proceeding to the next iteration. Removing clauses is needed as the discerning pair of objects (in our case, a pair of hyperboxes) requires only a single attribute of the corresponding matrix entries.

Further, the removal of clauses reduces space complexity for successive iterations. The iteration continues till M becomes empty. After the end condition is reached, the reduct obtained is returned by Johnson's algorithm.

M is an approximation of the crisp DM for the given dataset, the application of Johnson's algorithm on M results in an approximate reduct for the decision systems.

Illustration for reduct computation

After constructing DM (M) shown in Table 3 as illustrated in Section 3.2.2, a reduct R is computed using Johnson's algorithm, describe in Algorithm 2. Based on Algorithm 2, attribute 'b' is selected through $MDHeuristic$ on the first iteration because it appears most frequent in DM entries. After choosing attribute 'b', corresponding entries containing attribute 'b' are removed from M, resulting in an empty matrix. So, the obtained approximate reduct R for DT is {b}.

Algorithm 2: Finding single reduct using Johnson's reducer

```
Input  : M: Crisp discernibility matrix, C: Set of conditional
         attributes
Output: R: Approximate reduct
while M not empty do
    Red = ∅;
    Compute attribute a which have maximum appearances in
     clauses M_ij;
    for each a in C − Red do
        r = MDHeuristic(a);
        BestC = 0;
        if r > BestC then
            BestC = r;
            Best_Attribute = a;
        end
    end
    R = R ∪ {Best_Attribute} ;
    RemoveClauses(M,a);
end
Return R
```

3.3. Analysis of complexity

In CDM-FMFRS approach, the construction of IDS using FMNN preprocessing has a time complexity of $O(|U\|HB\|C|)$. Algorithm 1 constructs crisp DM with time complexity of $O(|HB|^2|C|)$. Algorithm 2 computes a reduct using Johnson's reducer algorithm (SFS computation) with time complexity of $O(|M||C|^2) = O(|HB|^2|C|^2)$. Hence, the time complexity of CDM-FMFRS is $max(O(|U\|HB\|C|), O(|HB|^2|C|^2))$.

The space complexity of CDM-FMFRS is $O(|U||C|)$ for the dataset, $O(|HB||C|)$ for IDS and $O(|HB|^2|C|)$ for crisp DM. Hence the total space complexity is $max(O(|U||C|), O(|HB|^2|C|))$.

4. Experiments

This section presents a comparative experimental evaluation of obtained CDM-FMFRS algorithm results. The proposed algorithm is compared with several recent (2018–2020) FRS methods, i.e., FMNN-FRS (Kumar and Prasad, 2020), FRGS (Zhang et al., 2018b), RMDPS (Dai et al., 2018), WRMDPS (Dai et al., 2018) and FRSEnt (Zhang et al., 2018a) algorithms.

Fifteen datasets are employed for experimental evaluation, outlined in Table 4. Thirteen of these datasets are taken from UCI machine learning repository (Dua and Graff, 2017), while the remaining two high dimensional datasets (Colon and Leukemia) are from Bioinformatics research group repository (www.biogps.org/dataset/).

The details of the system used for experiments is CPU: Intel(R) i7-8500, Clock Speed: 3.40 GHz × 6, RAM: 32 GB DDR4, OS: Ubuntu 18.04 LTS 64 bit and Software: Matlab R2017a.

A suitable combination of value for theta (0.3) and the sensitive parameter Gamma (4) is chosen for CDM-FMFRS algorithm as recommended from original FMNN (Simpson, 1992) and FMNN-FRS (Kumar and Prasad, 2020) papers. Furthermore, the comparative approaches (FMNN-FRS, FRGS, RMDPS, WRMDPS and FRSEnt) follow their own fuzzy model with t-norm, t-conorm and fuzzy similarity relations as given in the respective publications and experiments are conducted in the same environment stated above. In the experiment of CDM-FMFRS, we are empirically using value 0.1 for tolerance parameter (θ_1).

The performance of CDM-FMFRS algorithm is assessed with the following three aspects:

1. To measure the quality of approximate reduct using FRS based Gamma measure technique (given in Section 4.1).
2. Evaluate the performance by aiding in the construction of classifier learners based on ten-fold cross-validation (10-FCV) on several benchmark datasets (given in Section 4.2).
3. To evaluate the performance on big datasets in achieving increased scalability (given in Section 4.3).

Table 4
Details of Numeric Benchmark Datasets.

No.	Dataset	Attributes	Objects	Classes
1	Wine	13	178	3
2	Ionosphere	32	351	2
3	Musk1	166	476	2
4	WDBC	30	569	2
5	Steel	27	1941	7
6	Segment	16	2310	7
7	Robot	24	5456	4
8	Page block	10	5472	5
9	Texture	40	5500	11
10	Thyroid	21	7200	3
11	Colon	2000	62	2
12	leukemia	7129	72	2
13	Shuttle	9	57999	3
14	Susy	18	5000000	2
15	Hepmass[a]	27	100000 (10500000)	2

[a]A random sample of Hepmass dataset.

Table 5
Relevance of CDM-FMFRS reduct through Gamma measure.

Datasets	Gamma measure						
	ALL	CDM-FMFRS	FMNN-FRS	FRGS	RMDPS	WRMDPS	FRSEnt
Wine	1	0.97	0.95	1	1	1	1
Ionosphere	1	0.99	0.98	1	1	1	1
Musk1	1	0.99	0.95	1	1	1	1
WDBC	1	0.99	0.90	1	1	1	1
Steel	0.99	0.99	0.99	0.99	0.99	0.99	0.99
Segment	0.98	0.97	0.96	0.98	0.98	0.98	0.97
Robot	0.97	0.97	0.92	0.97	0.97	0.97	0.97
Page	0.87	0.84	0.85	0.87	0.87	0.87	0.87
Texture	0.99	0.99	0.94	0.99	0.99	0.99	0.99
Thyroid	0.96	0.96	0.94	0.96	0.96	0.96	0.96
Colon	1	0.99	0.91	1	1	1	1
Leukemia	1	0.89	0.89	1	1	1	1

It is to be noted that Susy and Hepmass datasets are very large in space, and none of the compared approaches could obtain reduct on these datasets. Hence, these datasets could not be used in Gamma measure and classification analysis in Section 4.1 and Section 4.2 respectively. These datasets are only applied in Section 4.3 to establish the increased scalability of CDM-FMFRS over FMNN-FRS.

4.1. To assess the quality of approximate CDM-FMFRS reduct

This section assesses the quality of obtained reduct or approximate reduct through gamma measure over the original decision system. The formulation of each algorithm is using its own FRS model to compute reduct. To avoid bias, we chose the Gaussian kernel fuzzy rough (GKFRS) set (Ghosh et al., 2017) for gamma measure for validating the relevance of reducts computed by given algorithms as none of these algorithms using this particular approach (GKFRS) in their model. For comparison perspective, gamma measure is computed for all attributes of the decision system (named as 'Unred') to validate the relevance of the obtained reduct by given algorithms. This way facilitates whether the obtained reduct is satisfying or meeting near to gamma measure of Unred or not.

Table 5 presents the resulting gamma values of reducts obtained by given algorithms on entire datasets. Table 5 reports the gamma value for only twelve datasets out of given fifteen benchmark datasets due to exceeding in memory limit while processing the GKFRS.

Table 6
10-FCV classification accuracy results (%) with CART classifier.

Datasets	CDM-FMFRS	FMNN-FRS		FRGS		RMDPS		WRMDPS		FRSEnt		Unred	
	Mean ± Std	Mean ± Std	*p*-Val	Mean ± Std	*p*-Val	Mean ± Std	*p*-Val	Mean ± Std	*p*-Val	Mean ± Std	*p*-Val	Mean ± Std	*p*-Val
Wine	92.14 ± 5.54	93.71 ± 4.40	0.50^{o}	90.96 ± 4.04	0.16^{o}	92.32 ± 6.83	0.91^{o}	91.74 ± 6.91	0.80^{o}	90.56 ± 4.12	0.22^{o}	89.57 ± 5.88	0.09^{o}
Ionosphere	89.74 ± 3.34	90.01 ± 3.38	0.85^{o}	89.46 ± 3.30	0.84^{o}	86.88 ± 3.39	0.06^{o}	88.30 ± 2.87	0.37^{o}	88.02 ± 4.01	0.37^{o}	88.03 ± 2.95	0.21^{o}
Musk1	79.76 ± 5.49	74.19 ± 5.28	0.03^{-}	76.55 ± 7.74	0.24^{o}	76.00 ± 5.43	0.13^{o}	77.52 ± 4.60	0.16^{o}	77.83 ± 9.56	0.62^{o}	78.32 ± 6.18	0.55^{o}
WDBC	93.15 ± 4.44	90.55 ± 5.11	0.24^{o}	93.31 ± 3.55	0.87^{o}	92.98 ± 2.85	0.88^{o}	92.98 ± 2.85	0.88^{o}	93.13 ± 3.42	0.98^{o}	92.64 ± 4.17	0.61^{o}
Steel	91.75 ± 2.07	91.29 ± 1.69	0.59^{o}	90.88 ± 1.84	0.33^{o}	91.70 ± 1.92	0.95^{o}	91.75 ± 1.88	0.95^{o}	90.88 ± 1.83	0.33^{o}	91.85 ± 2.00	0.91^{o}
Robot	99.39 ± 0.34	97.85 ± 1.06	0.00^{-}	99.35 ± 0.33	0.16^{o}	99.43 ± 0.34	0.34^{o}	99.41 ± 0.33	0.59^{o}	99.39 ± 0.34	1.00^{o}	99.39 ± 0.33	1.00^{o}
Page	96.71 ± 0.64	96.58 ± 0.59	0.64^{o}	96.40 ± 0.66	0.13^{o}	96.43 ± 0.64	0.21^{o}	96.41 ± 0.71	0.21^{o}	96.63 ± 0.79	0.77^{o}	96.45 ± 0.73	0.25^{o}
Texture	92.52 ± 0.90	91.34 ± 1.76	0.07^{o}	92.29 ± 0.67	0.52^{o}	92.16 ± 0.99	0.46^{o}	92.09 ± 0.77	0.26^{o}	92.38 ± 1.04	0.75^{o}	92.12 ± 0.84	0.31^{o}
Segment	96.01 ± 0.97	96.01 ± 0.88	1.00^{o}	95.80 ± 1.22	0.60^{o}	95.84 ± 1.38	0.70^{o}	95.75 ± 1.28	0.52^{o}	95.84 ± 1.19	0.68^{o}	95.75 ± 1.28	0.54^{o}
Thyroid	98.65 ± 0.78	97.27 ± 0.42	0.00^{-}	99.61 ± 0.18	0.00^{+}	98.66 ± 3.00	0.98^{o}	98.66 ± 3.00	0.98^{o}	99.51 ± 0.21	0.01^{+}	99.61 ± 0.18	0.00^{+}
Colon	77.08 ± 9.05	73.75 ± 16.43	0.58^{o}	75.41 ± 16.48	0.78^{o}	67.08 ± 11.85	0.05^{o}	75.41 ± 12.17	0.72^{o}	82.08 ± 9.63	0.19^{o}	67.08 ± 25.18	0.23^{o}
Leukemia	89.52 ± 14.33	86.34 ± 10.95	0.58^{o}	89.20 ± 7.87	0.95^{o}	89.20 ± 10.33	0.95^{o}	90.95 ± 11.09	0.80^{o}	93.17 ± 7.25	0.48^{o}	76.66 ± 17.01	0.08^{o}
Shuttle	99.10 ± 1.12	99.98 ± 0.01	0.02^{+}	[a]		[a]		[a]		[a]		99.90 ± 0.03	0.04^{+}
Average	91.96	90.68		90.76		89.89		90.91		91.61		89.79	
Lose/Win/Tie		3/1/9		0/1/11		0/0/12		0/0/12		0/1/11		0/2/11	

[a]Represents non-executable due to memory overflow.

4.1.1. Analysis of result

As, CDM-FMFRS computes as approximate reduct, based on Table 5, it is observed that CDM-FMFRS has indeed achieved a very much near to required gamma value as compared with Unred gamma value on almost all the datasets. So, CDM-FMFRS reduct has not resulted in any significant loss in quality.

The next Section 4.2 assesses the significance of obtained approximate reduct in inducing the classification model construction through 10-FCV. It is interesting to note that even though all FRS methods achieve the same or similar gamma measure on training data, the generalizability of the induced classifier varies from one reduct to another. Hence classification based analysis is required to assess the generalizability of the classifier model induced by a reduct algorithm.

4.2. The relevance of proposed approach in construction of classifiers

This section details the experimental evaluation conducted among CDM-FMFRS, FMNN-FRS, FRGS, RMDPS, WRMDPS and FRSEnt, respectively. The performance of the respective algorithm is evaluated based on the significance of reduct in inducing a classification model using 10-FCV technique. Also, unreduced dataset (named as 'Unred' in Tables 6–8) are also included to assess the relevance of reduct by respective algorithms over 10-FCV technique.

Three classifiers CART, kNN (k=3) and RBF-SVM (SVM using Radial Basis function) are applied to assess the performance of the respective algorithm. All parameters of classifiers are set to be default values. And, for kNN classifier, k is taken as 3.

Furthermore, a paired t-test with a significance level of 0.05 is performed to analyze the statistical evaluation of CDM-FMFRS results over given compared algorithms result and Unred. Each column in Tables reports the results of the respective algorithm and Unred in the form of mean and standard deviation along with the *p*-value except CDM-FMFRS column. CDM-FMFRS column contained only mean and standard deviation. The *p*-value index is the significant level between the respective algorithm and Unred with CDM-FMFRS algorithm. If the *p*-value is greater than 0.05, then there is no statistically significant difference, marked with the symbol 'o'. For classification, if the *p*-value is less than 0.05, and the result obtained by the respective algorithm is less than CDM-FMFRS, then the particular algorithm is inferior to CDM-FMFRS and marked as a loss '-'. Otherwise, it is representing a win '+'. The contrary is for reduct size and computational time analysis, which means if the *p*-value is less than 0.05, and the result obtained by the respective algorithm is less than CDM-FMFRS, then the particular algorithm is better than CDM-FMFRS and marked as a win '+'; otherwise, it is representing a loss '-'. Also, in each Table, the last two rows summarize total average results for given datasets and the count of loss '-', win '+' and no difference 'o' as competing with proposed CDM-FMFRS. The '*' sign in all Tables reports that the respective algorithm is not able to compute reduct on a given dataset due to memory overflow error.

Tables 6, 7 and 8 reported the comparative average classification accuracies for CART, kNN and RBF-SVM classifiers based on 10-FCV. Also, Tables 9 and 10 show the average reduct size and average computational time obtained by respective algorithms on 10-FCV.

4.2.1. Analysis of results

Table 6 presents the obtained classification results of CART classifier. Based on Table 6, CDM-FMFRS reached the highest total average performance (91.96) over all datasets. Also, it can be seen that CDM-FMFRS achieved statistically similar results with Unred in all datasets except Thyroid and Shuttle datasets. However, the difference in accuracies is trivial on these datasets. CDM-FMFRS algorithm obtained statistically similar results with FRGS, RMDPS, WRMDPS and FRSEnt algorithms in most of the datasets. Only in Thyroid dataset, CDM-FMFRS algorithm is statistically inferior to FRGS, FRSEnt and Unred, but the difference in average classification is minor, like 98.65 (CDM-FMFRS), 99.61 (FRGS) and 99.61 (Unred). Furthermore, in Musk1, Robot and Thyroid datasets, FMNN-FRS performed statistically inferior to CDM-FMFRS.

Similarly, in Table 7 for kNN classifier, CDM-FMFRS obtained the highest total average performance (91.97) than compared algorithms and Unred over all datasets. Furthermore, CDM-FMFRS showed statistically equivalent to Unred and compared algorithms in most of the datasets. CDM-FMFRS achieved statistically significant than FMNN-FRS in Wine, Musk1 and Texture datasets. In Thyroid dataset, CDM-FMFRS performed statistically inferior to compared algorithm and Unred; otherwise, in most of the datasets, it performed statistically similar or significant results.

Based on the classification results reported in Table 8 for RBF-SVM classifier, CDM-FMFRS performed statistically significant than Unred in Colon and Leukemia dataset. CDM-FMFRS showed statistically inferior to FRGS, RMDPS, WRMDPS, FRSEnt and Unred in Segment, Page and Texture dataset, although the difference in average accuracy is insignificant. For example, in Page dataset, CDM-FMFRS obtained 94.15 accuracy where other FRGS, RMDPS, WRMDPS, FRSEnt and Unred achieved 94.33, 94.31, 94.31, 94.20 and 94.31 average accuracies. Furthermore, FMNN-FRS performed statistically inferior to CDM-FMFRS in Musk1, WDBC, Robot and Texture datasets.

In Shuttle dataset, CDM-FMFRS achieved statistically equivalent with Unred in kNN and RBF-SVM classifiers where compared algorithms except FMNN-FRS are not possible to obtain reduct.

In terms of computational time in Table 9, CDM-FMFRS algorithm achieved statistically extremely significant in computational time than

Table 7
10-FCV classification accuracy results (%) with kNN (k=3) classifier.

Datasets	CDM-FMFRS	FMNN-FRS		FRGS		RMDPS		WRMDPS		FRSEnt		Unred	
	Mean ± Std	Mean ± Std	*p*-Val	Mean ± Std	*p*-Val	Mean ± Std	*p*-Val	Mean ± Std	*p*-Val	Mean ± Std	*p*-Val	Mean ± Std	*p*-Val
Wine	98.23 ± 2.84	93.50 ± 6.49	0.05^-	97.64 ± 3.04	0.34^o	95.29 ± 4.64	0.01^-	96.47 ± 3.04	0.08^o	97.05 ± 3.10	0.16^o	97.05 ± 3.10	0.16^o
Ionosphere	85.46 ± 3.92	87.46 ± 3.36	0.23^o	84.03 ± 5.91	0.34^o	84.33 ± 5.41	0.42^o	83.19 ± 5.93	0.15^o	84.33 ± 5.24	0.42^o	83.76 ± 5.71	0.29^o
Musk1	82.79 ± 5.14	74.37 ± 7.30	0.01^-	82.67 ± 7.55	0.96^o	83.03 ± 6.54	0.92^o	81.11 ± 3.91	0.30^o	83.28 ± 7.39	0.83^o	83.33 ± 5.33	0.79^o
WDBC	95.78 ± 3.37	92.87 ± 3.63	0.08^o	96.32 ± 1.91	0.49^o	95.94 ± 2.01	0.80^o	95.94 ± 2.09	0.80^o	97.01 ± 1.19	0.17^o	96.47 ± 1.88	0.38^o
Steel	92.53 ± 1.43	92.89 ± 1.86	0.63^o	92.78 ± 1.36	0.32^o	92.78 ± 1.43	0.32^o	92.78 ± 1.43	0.06^o	92.89 ± 1.15	0.43^o	92.78 ± 1.34	0.34^o
Segment	95.62 ± 0.94	96.32 ± 1.63	0.25^o	96.58 ± 0.65	0.02^+	96.53 ± 0.70	0.02^+	96.53 ± 0.70	0.02^+	96.66 ± 0.73	0.00^+	96.62 ± 0.83	0.01^+
Robot	87.26 ± 0.99	89.38 ± 1.74	0.00^+	87.26 ± 0.99	1.00^o	87.26 ± 0.99	1.00^o	87.26 ± 0.99	1.00^o	87.26 ± 0.99	1.00^o	87.26 ± 0.99	1.00^o
Page	95.86 ± 0.88	95.75 ± 0.93	0.78^o	95.94 ± 0.97	0.37^o	95.94 ± 0.97	0.37^o	95.94 ± 0.97	0.37^o	95.90 ± 0.93	0.55^o	95.94 ± 0.97	0.37^o
Texture	98.70 ± 0.57	97.54 ± 1.33	0.02^-	98.83 ± 0.43	0.00^+	98.83 ± 0.43	0.00^+	98.83 ± 0.43	0.00^+	98.90 ± 0.38	0.00^+	98.81 ± 0.51	0.00^+
Thyroid	96.59 ± 1.57	95.38 ± 1.39	0.08^o	94.06 ± 0.94	0.00^-	93.79 ± 1.54	0.00^-	93.79 ± 1.54	0.06^o	94.19 ± 0.98	0.00^-	94.06 ± 0.94	0.06^o
Colon	77.08 ± 22.67	77.08 ± 19.76	1.00^o	77.08 ± 14.33	1.00^o	78.75 ± 19.48	0.84^o	80.41 ± 10.76	0.61^o	82.08 ± 14.70	0.49^o	74.16 ± 11.41	0.65^o
Leukemia	90.95 ± 12.98	85.23 ± 10.39	0.29^o	84.12 ± 20.93	0.39^o	82.38 ± 11.88	0.14^o	86.66 ± 13.24	0.47^o	86.98 ± 22.01	0.62^o	84.92 ± 15.26	0.35^o
Shuttle	98.83 ± 1.20	99.91 ± 0.03	0.01^+	[a]		[a]		[a]		[a]		99.98 ± 0.01	0.66^o
Average	91.97	90.59		90.60		90.40		90.74		91.37		91.16	
Lose/Win/Tie		3/2/8		1/2/9		2/2/8		0/2/10		1/2/9		0/2/11	

[a]Represents non-executable due to memory overflow.

Table 8
10-FCV classification accuracy results (%) with RBF-SVM classifier.

Datasets	CDM-FMFRS	FMNN-FRS		FRGS		RMDPS		WRMDPS		FRSEnt		Unred	
	Mean ± Std	Mean ± Std	*p*-Val	Mean ± Std	*p*-Val	Mean ± Std	*p*-Val	Mean ± Std	*p*-Val	Mean ± Std	*p*-Val	Mean ± Std	*p*-Val
Wine	96.47 ± 3.03	93.31 ± 4.36	0.76^o	98.82 ± 2.48	0.07^o	97.64 ± 4.11	0.44^o	97.64 ± 4.11	0.44^o	98.82 ± 2.48	0.07^o	98.82 ± 2.48	0.07^o
Ionosphere	94.30 ± 3.54	92.31 ± 4.02	0.25^o	94.58 ± 2.10	0.78^o	94.58 ± 2.10	0.78^o	94.58 ± 2.10	0.78^o	94.30 ± 2.33	0.99^o	94.30 ± 2.33	0.99^o
Musk1	87.56 ± 3.02	71.42 ± 8.03	0.00^-	89.43 ± 5.18	0.33^o	84.54 ± 5.07	0.06^o	83.19 ± 3.39	0.00^-	88.84 ± 5.24	0.43^o	87.47 ± 4.81	0.95^o
WDBC	96.32 ± 3.38	93.03 ± 3.31	0.04^-	97.72 ± 1.66	0.15^o	97.90 ± 1.36	0.12^o	97.90 ± 1.36	0.12^o	97.90 ± 1.36	0.12^o	97.90 ± 1.36	0.12^o
Steel	93.56 ± 1.89	92.58 ± 2.66	0.35^o	93.61 ± 2.08	0.95^o	93.71 ± 2.12	0.86^o	93.71 ± 2.12	0.86^o	93.66 ± 1.93	0.90^o	93.71 ± 2.07	0.69^o
Segment	92.38 ± 2.22	93.07 ± 2.87	0.55^o	94.19 ± 1.35	0.00^+	94.19 ± 1.51	0.01^+	94.19 ± 1.51	0.01^+	94.02 ± 1.39	0.01^+	94.15 ± 1.46	0.01^+
Robot	90.28 ± 0.94	87.66 ± 0.97	0.00^-	90.25 ± 0.99	0.34^o	90.25 ± 0.99	0.34^o	90.25 ± 0.99	0.34^o	90.25 ± 0.99	0.34^-	90.25 ± 0.99	0.34^o
Page	94.15 ± 0.84	94.22 ± 0.80	0.85^o	94.33 ± 0.79	0.00^+	94.31 ± 0.81	0.00^+	94.31 ± 0.81	0.00^+	94.20 ± 0.72	0.49^o	94.31 ± 0.81	0.00^+
Texture	98.45 ± 0.67	96.36 ± 1.32	0.00^-	98.98 ± 0.21	0.02^+	98.98 ± 0.21	0.02^+	98.98 ± 0.21	0.02^+	98.96 ± 0.28	0.03^+	99.05 ± 0.18	0.01^+
Thyroid	93.02 ± 1.16	92.97 ± 0.99	0.91^o	93.08 ± 1.11	0.10^o	92.98 ± 1.33	0.66^o	92.98 ± 1.33	0.66^o	93.06 ± 1.14	0.08^o	93.08 ± 1.11	0.10^o
Colon	75.41 ± 22.77	70.41 ± 24.88	0.64^o	67.08 ± 18.05	0.09^o	78.75 ± 19.48	0.70^o	83.75 ± 11.18	0.27^o	64.16 ± 20.80	0.01^-	64.16 ± 20.80	0.01^-
Leukemia	85.23 ± 12.38	79.84 ± 14.09	0.37^o	65.55 ± 18.16	0.01^-	85.23 ± 14.09	1.00^o	88.09 ± 13.88	0.63^o	65.55 ± 18.16	0.01^-	65.55 ± 18.16	0.01^-
Shuttle	95.31 ± 2.92	97.88 ± 0.51	0.01^+	[a]		[a]		[a]		[a]		97.09 ± 0.19	0.07^o
Average	91.72	88.85		89.80		91.92		92.46		89.42		89.98	
Lose/Win/Tie		4/1/8		2/3/7		0/3/9		1/3/8		3/2/7		2/3/8	

[a]Represents non-executable due to memory overflow.

Table 9
Results of computational times in 10-FCV experiment (in seconds).

Datasets	CDM-FMFRS	FMNN-FRS		FRGS		RMDPS		WRMDPS		FRSEnt	
	Mean ± Std	Mean ± Std	*p*-Val	Mean ± Std	*p*-Val	Mean ± Std	*p*-Val	Mean ± Std	*p*-Val	Mean ± Std	*p*-Val
Wine	0.01 ± 0.00	0.01 ± 0.00	1.00^o	0.15 ± 0.01	0.00^-	0.06 ± 0.00	0.00^-	0.07 ± 0.00	0.00^-	0.31 ± 0.01	0.00^-
Ionosphere	0.07 ± 0.01	0.10 ± 0.01	0.00^-	0.98 ± 0.06	0.00^-	0.21 ± 0.01	0.00^-	0.21 ± 0.01	0.00^-	2.22 ± 0.15	0.00^-
Musk1	0.14 ± 0.01	0.56 ± 0.06	0.00^-	42.35 ± 3.15	0.00^-	0.69 ± 0.01	0.00^-	0.74 ± 0.01	0.00^-	64.5 ± 3.21	0.00^-
WDBC	0.05 ± 0.00	0.05 ± 0.00	1.00^o	1.06 ± 0.04	0.00^-	0.51 ± 0.01	0.00^-	0.57 ± 0.01	0.00^-	2.85 ± 0.04	0.00^-
Steel	0.48 ± 0.02	0.60 ± 0.03	0.00^-	4.81 ± 0.11	0.00^-	1.92 ± 0.05	0.00^-	2.11 ± 0.06	0.00^-	10.23 ± 0.14	0.00^-
Segment	0.02 ± 0.00	0.03 ± 0.00	0.38^o	17.64 ± 0.06	0.00^-	14.69 ± 0.05	0.00^-	16.12 ± 0.05	0.00^-	21.76 ± 0.05	0.00^-
Robot	4.01 ± 0.21	5.56 ± 0.26	0.00^-	111.6 ± 0.68	0.00^-	68.81 ± 0.57	0.00^-	73.60 ± 0.41	0.00^-	153.0 ± 0.85	0.00^-
Page	0.06 ± 0.01	0.06 ± 0.00	1.00^o	21.60 ± 0.16	0.00^-	17.26 ± 0.19	0.00^-	18.29 ± 0.18	0.00^-	30.37 ± 0.26	0.00^-
Texture	0.06 ± 0.00	0.12 ± 0.02	0.00^-	160.8 ± 1.27	0.00^-	105.4 ± 0.82	0.00^-	112.6 ± 0.95	0.00^-	220.1 ± 1.41	0.00^-
Thyroid	0.06 ± 0.00	0.08 ± 0.02	0.00^-	45.75 ± 1.12	0.00^-	25.22 ± 0.68	0.00^-	26.37 ± 0.67	0.00^-	87.03 ± 1.38	0.00^-
Colon	0.02 ± 0.01	0.12 ± 0.02	0.00^-	62.75 ± 14.08	0.00^-	0.07 ± 0.01	0.00^-	0.08 ± 0.00	0.00^-	131.5 ± 15.50	0.00^-
Leukemia	0.09 ± 0.01	0.24 ± 0.05	0.00^-	14767.0 ± 1.37	0.00^-	0.31 ± 0.01	0.00^-	0.33 ± 0.01	0.00^-	767.5 ± 42.88	0.00^-
Shuttle	0.44 ± 0.01	0.45 ± 0.01	0.03^-	[a]		[a]		[a]		[a]	
Average	0.42	0.65		1296		19.59		20.92		124.37	
Lose/Win/Tie		10/0/3		12/0/0		12/0/0		12/0/0		12/0/0	

[a]Represents non-executable due to memory overflow.

compared algorithms except FMNN-FRS in all datasets. Especially in Robot, Page, Texture and Thyroid datasets, CDM-FMFRS has gained around 99% of the computational time than compared algorithms. Also, it can able to compute reduct on the Shuttle dataset where these algorithms except FMNN-FRS could not perform due to exceed in space complexity. CDM-FMFRS obtained statistically significant less in computational time than FMNN-FRS in most of the datasets. Moreover, CDM-FMFRS incurred the lowest total average computation time in reduct computation than compared algorithms for all datasets. Even the resulting standard deviation of computation time in 10-FCV has carried

Table 10
Results of reduct length in 10-FCV experiment.

Datasets	CDM-FMFRS	FMNN-FRS		FRGS		RMDPS		WRMDPS		FRSEnt	
	Mean ± Std	Mean ± Std	p-Val	Mean ± Std	p-Val	Mean ± Std	p-Val	Mean ± Std	p-Val	Mean ± Std	p-Val
Wine	4.80 ± 0.63	3.30 ± 0.48	0.00^{+}	12.90 ± 0.31	0.00^{-}	9.90 ± 0.31	0.00^{-}	10.10 ± 0.31	0.00^{-}	13.00 ± 0.00	0.00^{-}
Ionosphere	12.50 ± 1.50	6.50 ± 0.52	0.00^{+}	30.60 ± 0.84	0.00^{-}	26.80 ± 1.39	0.00^{-}	27.40 ± 1.42	0.00^{-}	31.00 ± 0.00	0.00^{-}
Musk1	39.20 ± 4.01	8.00 ± 0.00	0.00^{+}	93.00 ± 3.59	0.00^{-}	18.00 ± 0.47	0.00^{+}	18.30 ± 0.67	0.00^{+}	103.6 ± 2.63	0.00^{-}
WDBC	12.50 ± 2.54	3.20 ± 0.42	0.00^{+}	27.00 ± 0.66	0.00^{-}	23.80 ± 0.78	0.00^{-}	23.80 ± 0.78	0.00^{-}	27.10 ± 0.31	0.00^{-}
Steel	21.70 ± 0.94	14.60 ± 1.34	0.00^{+}	17.50 ± 0.70	0.00^{+}	20.00 ± 0.66	0.00^{+}	20.00 ± 0.66	0.00^{+}	15.90 ± 0.31	0.00^{+}
Segment	7.80 ± 1.47	7.60 ± 0.51	0.68^{o}	14.10 ± 0.31	0.00^{-}	15.00 ± 0.00	0.00^{-}	15.00 ± 0.00	0.00^{-}	12.90 ± 0.31	0.00^{-}
Robot Nav.	23.90 ± 0.31	12.70 ± 0.94	0.00^{+}	24.00 ± 0.00	0.32^{o}	24.00 ± 0.00	0.32^{o}	24.00 ± 0.00	0.32^{o}	24.00 ± 0.00	0.32^{o}
Page	7.50 ± 0.70	8.00 ± 0.00	0.03^{-}	9.90 ± 0.31	0.00^{-}	9.90 ± 0.31	0.00^{-}	9.90 ± 0.31	0.00^{-}	7.90 ± 0.31	0.11^{o}
Texture	16.30 ± 1.94	10.00 ± 1.24	0.00^{+}	37.00 ± 0.00	0.00^{-}	37.00 ± 0.00	0.00^{-}	37.00 ± 0.00	0.00^{-}	30.50 ± 0.52	0.00^{-}
Thyroid	9.80 ± 1.03	8.20 ± 2.04	0.04^{+}	20.00 ± 0.00	0.00^{-}	18.10 ± 6.01	0.00^{-}	18.10 ± 6.01	0.00^{-}	18.00 ± 0.00	0.00^{-}
Colon	4.20 ± 0.78	2.90 ± 0.31	0.00^{+}	104.8 ± 10.92	0.00^{-}	6.00 ± 0.47	0.00^{-}	6.60 ± 0.51	0.00^{-}	129.6 ± 4.69	0.00^{-}
Leukemia	2.00 ± 0.00	2.00 ± 0.00	1.00^{o}	979.5 ± 688.5	0.00^{-}	4.10 ± 0.31	0.00^{-}	4.20 ± 0.42	0.00^{-}	437.7 ± 20.56	0.00^{-}
Shuttle	3.90 ± 0.56	5.50 ± 0.52	0.00^{-}	[a]		[a]		[a]		[a]	
Average	12.77	7.11		114.19		17.71		17.86		70.93	
Lose/Win/Tie		2/9/2		10/1/1		9/2/1		9/2/1		9/1/2	

[a]Represents non-executable due to memory overflow.

very less variation as compared with compared approaches showing that methodology is reliable. It can be seen that FMNN-FRS and CDM-FMFRS algorithms have achieved much less computational times than other algorithms. This is due to utility arising from FMNN preprocessing which makes both algorithms to operate in hyperbox space rather than object space where $|HB| \ll |U|$ in all datasets. Further, it is observed that FMNN-FRS achieved better computation time than FMNN-FRS. This is attributed to using crisp DM in CDM-FMFRS in contrast to fuzzy DM in FMNN-FRS.

Moreover, considering the average reduct size, as shown in Table 10, the total average reduct size through CDM-FMFRS algorithm for all datasets is smaller than compared algorithms except for FMNN-FRS. CDM-FMFRS has gained around 50% of the average reduct size than compared algorithms except for FMNN-FRS in Wine, Ionosphere, WDBC, Segment, Texture and Thyroid datasets. FMNN-FRS achieved statistically significant in average reduct size than CDM-FMFRS in most of the datasets but, in terms of classification accuracies and computational time, CDM-FMFRS performed significantly better. In FMNN-FRS, the partial fuzzy membership is calculated based on s-norm computation and satisfy the total required s-norm for entire discernibility matrix entry with a fewer number of attributes. In FMNN-FRS, if one attribute is selected, then it contributes some partial membership value to all the entries in fuzzy DM. Whereas in CDM-FMFRS, if one attribute is selected, then it contributes only to the entries in crisp DM containing that attribute and has no effect on the remaining entries. Hence it is observed that the average reduct size in FMNN-FRS is lesser than CDM-FMFRS in most of the datasets.

4.3. Role of crisp DM in increased scalability of CDM-FMFRS over FMNN-FRS

On given datasets, we have seen that except for getting little higher reduct lengths, CDM-FMFRS methodology has obtained significant gain in computational time over FMNN-FRS and other start-of-art of FRS approaches. It is seeing that despite restricting to crisp DM construction, CDM-FMFRS is able to give good quality approximate reduct, which could induce a classification model with comparable or better accuracy. So far, all the experiments are conducted on such datasets over which compared algorithms can be executed for demonstrating the quality of CDM-FMFRS approximate reduct comparatively.

In this section, we demonstrate the improved scalability of CDM-FMFRS over FMNN-FRS, which is the prime objective for the proposed work. It is also to be noted that all the other FRS reduct approaches are not executable on considered datasets in this experiment owing to memory overflow.

Table 4 also gives details of the big numeric datasets (Susy and Hepmass) used in this experiment. Both datasets Susy and Hepmass, along with few given datasets from Table 4, are considered for experiments. A random sample of Hepmass dataset is considered with 100000 objects.

We have applied both CDM-FMFRS and FMNN-FRS on these datasets, and detailed results are reported in Table 11. As the stage for FMNN preprocessing is common to both algorithms, the number of hyperboxes and computational time (in seconds) for FMNN preprocessing is specified only once. Each column in Table 11 reports the result size of DM matrix (in MB), DM construction time (in seconds), Reduct Computation Time (in seconds), Reduct Size and Total Time (in seconds) of both CDM-FMFRS and FMNN-FRS algorithms along with percentage gain of CDM-FMFRS over FMNN-FRS.

Analysis of results

Based on the results in Table 11, CDM-FMFRS achieved significant gain in DM memory size as compared with FMNN-FRS with the same percentage of 87.50% in all datasets. Because in Matlab environment, a real number is represented in 8 bytes, and the logical number is represented in 1 byte. Hence, crisp DM size of CDM-FMFRS is 1/8 of fuzzy DM size of FMNN-FRS. In other programming environments where the logical value is represented in 1 bit, we would obtain a reduction of 1/64 size of fuzzy DM. Also, CDM-FMFRS obtained a significant gain on time for the construction of DM over FMNN-FRS with a range of 8%–63% on given datasets.

Furthermore, there is a significant percentage gain in reduct computation time on constructed DM in CDM-FMFRS over FMNN-FRS, more than 90% in all datasets. This is due to Johnson algorithms (Ohrn, 2000) having lesser computations when applied to crisp DM in comparison to being applied on fuzzy DM. Hence, the total computation time of CDM-FMFRS was achieved with a range of 4 to 47% in given datasets against FMNN-FRS algorithm.

In Susy dataset, CDM-FMFRS computed reduct in a significantly lower time of around 5 seconds than 831 seconds in FMNN-FRS. Additionally, DM construction time for CDM-FMNN has a slight gain of 8% against FMNN-FRS, and also a significant reduction in the size of DM in RAM is obtained in CDM-FMFRS. DM size in CDM-FMFRS (630.78 MB) is 1/8 of DM size of FMNN-FRS (5048.32 MB).

CDM-FMFRS could obtain reduct in Hepmass dataset, whereas FMNN-FRS failed to do so because of memory overflow. Because the current system employed with 32 GB RAM and hence the requirement of DM for FMNN-FRS for Hepmass dataset in FMNN-FRS would have been $4741.12MB$ $(4.63GB) \times 8$ $(> 32GB)$ and which is the reason for FMNN-FRS is failing the reduct computation as required memory size exceeds available memory of 32 GB. This experiment vividly demonstrates the increased applicability of CDM-FMFRS to much larger numeric datasets and establishes the relevance of CDM-FMNN over FMNN-FRS.

Table 11
Experiment Results of CDM-FMFRS and FMNN-FRS.

Datasets	FMNN		DM size			DM time		
	NoH	ToFM	CDM-FMFRS	FMNN-FRS	Gain (%)	CDM-FMFRS	FMNN-FRS	Gain (%)
Robot	611	2.64	2.7161	21.7289	87.50	0.4034	1.0650	62.12
Page	29	0.06	0.0031	0.0244	87.50	0.0220	0.0291	24.41
Texture	42	0.08	0.0296	0.2371	87.50	0.0209	0.0248	15.92
Thyroid	38	0.08	0.0056	0.0450	87.50	0.0212	0.0245	13.72
Colon	18	0.03	0.1469	1.1749	87.50	0.0251	0.0281	10.74
Leukemia	26	0.08	1.0878	8.7024	87.50	0.0842	0.0942	10.63
Shuttle	13	0.33	0.0006	0.0045	87.50	0.0185	0.0201	7.92
Susy	13356	7254.8	630.78	5048.32	87.50	82.30	90.42	8.92
Hepmass	26212	2998.8	4741.12	[a]	[a]	478.8	[a]	[a]

Datasets	Reduct time			Reduct size			Total time		
	CDM-FMFRS	FMNN-FRS	Gain (%)	CDM-FMFRS	FMNN-FRS	Gain (%)	CDM-FMFRS	FMNN-FRS	Gain (%)
Robot	0.0050	2.1117	99.77	24	12	−100.00	3.0850	5.8117	46.92
Page	0.0013	0.0325	95.97	8	7	−14.29	0.0816	0.1195	31.72
Texture	0.0015	0.0642	97.64	17	12	−41.67	0.0883	0.1656	46.66
Thyroid	0.0012	0.0261	95.25	12	9	−33.33	0.1020	0.1354	24.72
Colon	0.0021	0.1375	98.47	6	3	−100.00	0.0533	0.1920	72.25
Leukemia	0.0010	0.3594	99.71	2	2	0.00	0.1626	0.5334	69.52
Shuttle	0.0007	0.0125	94.65	4	5	20.00	0.3505	0.3658	4.19
Susy	4.82	831.01	99.41	18	18	0	7341.12	8175.43	10.20
Hepmass	6.47	[a]	[a]	26	[a]	[a]	3484.07	[a]	[a]

[a]Represents non-executable due to memory overflow. NoH: Number of Hyperboxes, ToFM: Time for FMNN construction (in Seconds), Gain (%): Percentage gain of CDM-FMFRS over FMNN-FRS, DM: Discernibility Matrix, DM Memory Size (in MegaBytes), DM Time (in Seconds), Reduct Time (in Seconds), Total Time (in Seconds).

Based on these results, one can clearly say that crisp DM formulation significantly reduces the size of DM and reduct computation time. CDM-FMFRS facilitates increased scalability with the disadvantage of a slightly higher length reduct than FMNN-FRS due to information loss in the crisp DM formulation. Even though we obtain a higher size reduct in some datasets through crisp formulation, the quality of reduct is not compromised as clearly established in obtained Gamma measure showing in Section 4.1 and comparable classification model construction in Section 4.2. Even, tolerance parameter enriched the quality of reduct. Hence we recommend CDM-FMFRS as an alternative to FMNN-FRS in a situation where FMNN-FRS fails to obtain reduct owing to memory overflow error.

5. Future work

1. CDM-FMFRS algorithm only works on numeric decision system, not hybrid decision system. This is owing to the FMNN algorithm being only applicable to numerical attributes. In future, we will investigate our proposed method to work on the hybrid decision system.
2. After achieving increased scalability in CDM-FMFRS, we are aiming in the future to design an approach to overcome the limitation that arises due to CDM-FMFRS being a standalone reduct computation method. This is because as CDM-FMFRS requires the entire DM to be loaded into RAM, we can apply an algorithm to only those datasets whose DM size is under the system limit of RAM size. For this, we require a distributed/parallel solution wherein DM can be partitioned across the cluster node, and reduct computation is achieved with horizontal scalability (Lenz et al., 2019). This will be explored in future.

6. Conclusion

The proposed work (CDM-FMFRS) is an improved mechanism of the existing FMNN-FRS method to enhance the scalability of reduct computation in hyperbox-space. In CDM-FMFRS, a novel approach for crisp DM formulation in IDS is proposed subject to tolerance criteria for preserving maximal discernible attributes. Hence, the approach achieved significant gain in computation time over FMNN-FRS and other existing FRS reduct approaches on given benchmark datasets with similar or better classification accuracies over induced different classifiers. Even the space utilization of crisp DM is $\frac{1}{8\times k}$ of space utilization of fuzzy DM. Moreover, CDM-FMFRS approach can handle very large datasets where FMNN-FRS and other existing state-of-art FRS reduct approaches fail to obtain reduct. In future, distributed/parallel algorithms for CDM-FMFRS will be investigated for scaling to such voluminous datasets requiring memory beyond the availability in a single system.

CRediT authorship contribution statement

Anil Kumar: Conceptualization, Methodology, Writing – original draft. **P.S.V.S. Sai Prasad:** Supervision, Conceptualization, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Bhatt, R.B., Gopal, M., 2005. On the compact computational domain of fuzzy-rough sets. Pattern Recognit. Lett. 26 (11), 1632–1640.
Chen, D., Wang, X., Zhao, S., 2007. Attribute reduction based on fuzzy rough sets. In: Rough Sets And Intelligent Systems Paradigms. vol. 4585, pp. 381–390.
Chen, D., Zhang, L., Zhao, S., Hu, Q., Zhu, P., 2012. A novel algorithm for finding reducts with fuzzy rough sets. IEEE Trans. Fuzzy Syst. 20 (2), 385–389.
Cornelis, C., Jensen, R., Hurtado, G., Ślezak, D., 2010. Attribute selection with fuzzy decision reducts. Inform. Sci. 180 (2), 209–224.
Dai, J., Hu, H., Wu, W., Qian, Y., Huang, D., 2018. Maximal-discernibility-pair-based approach to attribute reduction in fuzzy rough sets. IEEE Trans. Fuzzy Syst. 26 (4), 2174–2187.
Dua, D., Graff, C., 2017. UCI machine learning repository. URL http://archive.ics.uci.edu/ml.
Dubois, D., Prade, H., 1990. Rough fuzzy sets and fuzzy rough sets. Int. J. Gen. Syst. 17 (2–3), 191–209.
Dubois, D., Prade, H., 1992. Putting rough sets and fuzzy sets together. In: Intelligent Decision Support: Handbook Of Applications And Advances Of The Rough Sets Theory. Springer Netherlands, Dordrecht, pp. 203–232.
Ghosh, S., Sai Prasad, P.S.V.S., Rao, C.R., 2017. Third order backward elimination approach for fuzzy-rough set based feature selection. In: Shankar, B.U., Ghosh, K., Mandal, D.P., Ray, S.S., Zhang, D., Pal, S.K. (Eds.), Pattern Recognition And Machine Intelligence. Springer International Publishing, Cham, pp. 254–262.

Hu, Q., Xie, Z., Yu, D., 2007. Hybrid attribute reduction based on a novel fuzzy-rough model and information granulation. Pattern Recognit. 40 (12), 3509–3521.

Jaccard, P., 1908. Nouvelles recherches sur la distribution florale. Bull. Soc. Vaudoise Sci. Nat. 44, 223–270. http://dx.doi.org/10.5169/seals-268384.

Jensen, R., Shen, Q., 2004. Fuzzy-rough attribute reduction with application to web categorization. Fuzzy Sets Syst. 141 (3), 469–485.

Jensen, R., Shen, Q., 2007. Fuzzy-rough sets assisted attribute selection. IEEE Trans. Fuzzy Syst. 15 (1), 73–89.

Jensen, R., Shen, Q., 2009. New approaches to fuzzy-rough feature selection. IEEE Trans. Fuzzy Syst. 17 (4), 824–838.

Jensen, R., Tuson, A., Shen, Q., 2014. Finding rough and fuzzy-rough set reducts with SAT. Inform. Sci. 255, 100–120.

Khuat, T.T., Gabrys, B., 2020. A comparative study of general fuzzy min-max neural networks for pattern classification problems. Neurocomputing 386, 110–125.

Kumar, A., Prasad, P.S.V.S.S., 2020. Scalable fuzzy rough set reduct computation using fuzzy min–max neural network preprocessing. IEEE Trans. Fuzzy Syst. 28 (5), 953–964.

Lenz, O.U., Peralta, D., Cornelis, C., 2019. A scalable approach to fuzzy rough nearest neighbour classification with ordered weighted averaging operators. In: Mihályde ák, T., Min, F., Wang, G., Banerjee, M., Düntsch, I., Suraj, Z., Ciucci, D. (Eds.), Rough Sets. Springer International Publishing, Cham, pp. 197–209.

Lin, T.Y., 1999. Granular computing: Fuzzy logic and rough sets. In: Zadeh, L.A., Kacprzyk, J. (Eds.), Computing With Words In Information/Intelligent Systems 1: Foundations. Physica-Verlag HD, Heidelberg, pp. 183–200.

Liu, Y., Zheng, L., Xiu, Y., Yin, H., Zhao, S., Wang, X., Chen, H., Li, C., 2020. Discernibility matrix based incremental feature selection on fused decision tables. Int. J. Approx. Reason. 118, 1–26.

Mohammed, M.F., Lim, C.P., 2015. An enhanced fuzzy min–max neural network for pattern classification. IEEE Trans. Neural Netw. Learn. Syst. 26 (3), 417–429.

Ohrn, A., 2000. Discernibility and rough sets in medicine: Tools and applications. Dep. Comput. Inf. Sci. 239, 1–239.

Pawlak, Z., 1982. Rough sets. Int. J. Comput. Inf. Sci. 11 (5), 341–356.

Pawlak, Z., 1991. Rough Sets-Theoretical Aspects Of Reasoning About Data. In: Theory and Decision Library : Series D, vol. 9, Kluwer, pp. 1–229.

Qu, Y., Shen, Q., Parthaláin, N.M., Shang, C., Wu, W., 2013. Fuzzy similarity-based nearest-neighbour classification as alternatives to their fuzzy-rough parallels. Int. J. Approx. Reason. 54 (1), 184–195.

Reyes-Galaviz, O.F., Pedrycz, W., 2015. Granular fuzzy modeling with evolving hyperboxes in multi-dimensional space of numerical data. Neurocomputing 168, 240–253.

Simpson, P.K., 1992. Fuzzy min-max neural networks. I. Classification. IEEE Trans. Neural Netw. 3 (5), 776–786.

Skowron, A., Rauszer, C., 1992. The discernibility matrices and functions in information systems. In: Slowinski, R. (Ed.), Intelligent Decision Support. In: Theory and Decision Library, vol. 11, Springer, pp. 331–362.

Slezak, D., 1996. Approximate reducts in decision tables. In: Proc. Sixth Int'L Conf. Information Processing And Management Of Uncertainty In Knowledge-Based Systems. pp. 1159–1164.

Tsang, E.C.C., Chen, D., Yeung, D.S., Wang, X., Lee, J.W.T., 2008. Attributes reduction using fuzzy rough sets. IEEE Trans. Fuzzy Syst. 16 (5), 1130–1141.

Tsang, E.C.C., Chen, D., Yeung, D.S., Wang, X.-Z., Lee, J.W.T., 2008. Attributes reduction using fuzzy rough sets. IEEE Trans. Fuzzy Syst. 16 (5), 1130–1141. http://dx.doi.org/10.1109/TFUZZ.2006.889960.

Yang, Y., Chen, D., Wang, H., Wang, X., 2018. Incremental perspective for feature selection based on fuzzy rough sets. IEEE Trans. Fuzzy Syst. 26 (3), 1257–1273.

Yao, Y., Zhao, Y., Wang, J., 2006. On reduct construction algorithms. In: Wang, G.-Y., Peters, J.F., Skowron, A., Yao, Y. (Eds.), Rough Sets And Knowledge Technology. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 297–304.

Zhang, X., Liu, X., Yang, Y., 2018a. A fast feature selection algorithm by accelerating computation of fuzzy rough set-based information entropy. Entropy 20 (10), 1–13.

Zhang, X., Mei, C., Chen, D., Li, J., 2016. Feature selection in mixed data: A method using a novel fuzzy rough set-based information entropy. Pattern Recognit. 56, 1–15.

Zhang, X., Mei, C., Chen, D., Yang, Y., 2018b. A fuzzy rough set-based feature selection method using representative instances. Knowl.-Based Syst. 151, 216–229.

Zhang, X., Mei, C., Chen, D., Yang, Y., Li, J., 2020. Active incremental feature selection using a fuzzy-rough-set-based information entropy. IEEE Trans. Fuzzy Syst. 28 (5), 901–915. http://dx.doi.org/10.1109/TFUZZ.2019.2959995.