Received May 7, 2021, accepted June 12, 2021, date of publication June 16, 2021, date of current version June 28, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3089849

Numerical Data Classification via Distance-Based Similarity Measures of Fuzzy Parameterized Fuzzy Soft Matrices

SAMET MEMİŞ[1], SERDAR ENGİNOĞLU[1], AND UĞUR ERKAN[2]

[1]Department of Mathematics, Faculty of Arts and Sciences, Çanakkale Onsekiz Mart University, 17100 Çanakkale, Turkey

[2]Department of Computer Engineering, Faculty of Engineering, Karamanoğlu Mehmetbey University, 70100 Karaman, Turkey

Corresponding author: Samet Memiş (samettmemis@gmail.com)

This work was supported by the 2211-C Domestic Doctoral Fellowship for Priority Areas by the Scientific and Technological Research Council of Turkey (TÜBİTAK) under Grant 1649B031905299.

ABSTRACT In this paper, we first define eight pseudo-metrics and eight pseudo-similarities based on these pseudo-metrics over *fpfs*-matrices. We then propose a new classification algorithm, i.e. Fuzzy Parameterized Fuzzy Soft Euclidean Classifier (FPFS-EC), based on Euclidean pseudo-similarity. After that, we compare FPFS-EC with Support Vector Machines (SVM), Fuzzy k-Nearest Neighbor (Fuzzy kNN), Fuzzy Soft Set Classifier (FSSC), FussCyier, Fuzzy Soft Set Classification Using Hamming Distance (HDFSSC), and Fuzzy kNN Based on the Bonferroni Mean (BM-Fuzzy kNN) in terms of the performance criteria - namely accuracy, precision, recall, macro F-score, and micro F-score - and running time by using 18 real-world datasets in the UCI machine learning repository. The results show that FPFS-EC performs better in the occurrence of the 13 of 18 datasets in question than SVM, Fuzzy kNN, FSSC, FussCyier, HDFSSC, and BM-Fuzzy kNN.

INDEX TERMS Fuzzy sets, soft sets, *fpfs*-matrices, similarity measure, classification, supervised learning.

I. INTRODUCTION

It is encountered with various types of uncertainty in many fields, such as medicine, the defense industry, psychology, finance, astronomy, meteorology, and space sciences. The concept of soft sets [1] is a standard and practical mathematical tool used for modeling such uncertainties. Moreover, research on some generalizations of this concept, such as fuzzy soft sets (*fs*-sets) [2], [3], fuzzy parameterized soft sets [4], fuzzy parameterized fuzzy soft sets [5], soft matrices [6], fuzzy soft matrices [7], fuzzy parameterized fuzzy soft matrices (*fpfs*-matrices) [8], have been introduced. Due to these generalizations, problems' modeling containing fuzzy parameters and/or fuzzy alternatives (objects) have been possible. Since *fpfs*-matrices successfully model problems where both parameters and alternatives are uncertain, they are prominent in their substructures. Furthermore, recent research has studied the configuration of soft decision-making methods to *fpfs*-matrices space [9]–[14], the simplification of the configured methods [15]–[18], and their applications to performance-based value assignment (PVA) problem in image denoising [19]–[22]. These studies have corroborated that *fpfs*-matrices successfully model the decision-making problems where both parameters and alternatives are uncertain.

So far, many studies have been conducted on the concept of soft sets in such fields as soft algebra [23]–[26], soft topology [27]–[31], decision-making [32]–[36], similarity measure [37]–[40], distance measure [38], [41], medical diagnosis [42], texture classification [43], and data classification [44]–[46]. Although the studies mentioned above have been carried out in a great variety of fields, these studies feature modeled problems often similar to each other and fictitious, except Fuzzy Soft Set Classifier (FSSC) [44], FussCyier [45], Fuzzy Soft Set Classification using Hamming Distance (HDFSSC) [46]. In particular, most studies on decision-making problems and similarity measures have been applied only to fashioned problems. Since similarity and

The associate editor coordinating the review of this manuscript and approving it for publication was Chun-Hao Chen.

TABLE 1. Properties of the proposed and compared classifiers based on fuzzy sets and *fs*-sets.

Ref.	Year	Classifier	Crisp	Fuzzy set	*fs*-set	*fpfs*-matrix	Inverse Distance	Distance-Based Similarity	Parameters' Impact
[47]	1985	Fuzzy kNN		✓			✓		
[49]	1995	SVM	✓						
[44]	2012	FSSC			✓			✓	
[45]	2017	FussCyier			✓			✓	
[46]	2018	HDFSSC			✓			✓	
[48]	2020	BM-Fuzzy kNN		✓			✓		
Proposed	2021	FPFS-EC				✓		✓	✓

distance measures play an essential role in machine learning and soft sets can effectively model problems containing uncertainties, applying similarity and distance measures of soft sets to real problems should be attended to. For example, recently, [45] have developed a classification algorithm, i.e. FussCyier, using a similarity measure of *fs*-sets for medical data classification. However, *fs*-sets cannot model problems containing fuzzy parameters. That is, *fs*-sets cannot consider the question "Which parameters are capable of effectively classifying data?", but *fpfs*-sets can. Therefore, it yields more successful results. Taking all of these into account, *fpfs*-sets are more suitable for a highly successful modeling and outperform the aforementioned. On the other hand, the matrix representations of *fpfs*-sets, i.e. *fpfs*-matrices, are needed to process a large number of data. To this end, we put forward distance measures and distance-based similarity measures of *fpfs*-matrices and apply the similarity measures to real numerical data classification. It can be summed up the major theoretical and applied contributions of the present paper as follows:

- The concepts of quasi-metric, semi-metric, pseudo-metric, and metric over *fpfs*-matrices spaces are defined. Afterward, eight pseudo-metrics over *fpfs*-matrices are proposed.
- The concepts of quasi-similarity, semi-similarity, pseudo-similarity, and similarity over *fpfs*-matrices spaces are defined. Afterward, eight pseudo-similarities over *fpfs*-matrices are proposed.
- This study is one of the pioneer studies combining soft sets and machine learning.
- In opposition to many studies working on a fictitious problem, this paper has applied the distance-based similarity measures of *fpfs*-matrices to classification problems in machine learning.
- A new classifier, referred to as Fuzzy Parameterized Fuzzy Soft Euclidean Classifier (FPFS-EC), employing Euclidean pseudo-similarity of *fpfs*-matrices and considering parameters' impact on classification, has been developed.

To demonstrate FPFS-EC's classification performance, besides the state-of-the-art fuzzy soft-based classifiers such as FSSC [44], FussCiyer [45], and HDFSSC [46], we compare it with a well-known fuzzy-based classifier and its state-of-the-art version, i.e., Fuzzy k-Nearest Neighbor (Fuzzy kNN) [47] and Fuzzy kNN based on the Bonferroni Mean (BM-Fuzzy kNN) [48], respectively. Moreover, we compare the proposed method with Support Vector Machines (SVM) [49]. We detail the classifiers in Table 1. In comparison, we utilize 18 real-world datasets from the University of California-Irvine (UCI) Machine Learning Repository [50]. Additionally, we provide a statistical evaluation of the comparison results.

The rest of the paper is organized as follows: In Section 2, we present definitions of fuzzy parameterized fuzzy soft sets and fuzzy parameterized fuzzy soft matrices. In Section 3, we define eight pseudo-metrics of *fpfs*-matrices and in Section 4, eight pseudo-similarities of *fpfs*-matrices based on these pseudo-metrics. In Section 5, we propose FPFS-EC using the Euclidean pseudo-similarity of *fpfs*-matrices. In Section 6, we first compare FPFS-EC with SVM [49], FSSC [44], FussCyier [45], HDFSSC [46], Fuzzy kNN [47], and BM-Fuzzy kNN [48] classifiers in terms of running time and performance criteria, such as accuracy, precision, recall, macro F-score, and micro F-score by processing 18 numerical datasets in the UCI database. We then present the statistical analyses and their results. Finally, we provide the conclusive remarks and make some suggestions for further research. This study is a part of the first author's PhD dissertation.

II. PRELIMINARIES

In this section, we present some of the basic definitions needed for the following sections. Throughout this paper, let E be a parameter set, $F(E)$ be the set of all fuzzy sets over E, and $\mu \in F(E)$. Here, $\mu := \{{}^{\mu(x)}x : x \in E\}$.

Definition 1 [5]: Let U be a universal set, $\mu \in F(E)$, and α be a function from μ to $F(U)$. Then, the set $\left\{({}^{\mu(x)}x, \alpha({}^{\mu(x)}x)) : x \in E\right\}$, the graphic of α, is called a fuzzy parameterized fuzzy soft set (fpfs-*set) parameterized via E over U (or briefly over U).*

From now on, the set of all *fpfs*-sets over U is denoted by $FPFS_E(U)$. In $FPFS_E(U)$, since the graph(α) and α generate each other uniquely, the notations are interchangeable. Therefore, as long as it causes no confusion, we denote a *fpfs*-set graph(α) by α.

Example 2: Let $E = \{x_1, x_2, x_3\}$ *and* $U = \{u_1, u_2, u_3\}$*. Then,*

$$\alpha = \Big\{({}^{1}x_1, \{{}^{0.5}u_1, {}^{0.2}u_2, {}^{0.4}u_3\}), ({}^{0.2}x_2, \{{}^{0.1}u_1, {}^{0.1}u_2, {}^{0.8}u_3\}), ({}^{0.4}x_3, \{{}^{1}u_1, {}^{0.5}u_2, {}^{1}u_3\})\Big\}$$

and

$$\beta = \Big\{({}^{0.1}x_1, \{{}^{0.6}u_1, {}^{1}u_2, {}^{0.7}u_3\}), ({}^{0.9}x_2, \{{}^{0.8}u_1, {}^{1}u_2, {}^{0.3}u_3\}), ({}^{0.7}x_3, \{{}^{1}u_1, {}^{0.6}u_2, {}^{0.3}u_3\})\Big\}$$

are two fpfs*-sets over* U.

Definition 3 [8]: Let $\alpha \in FPFS_E(U)$*. Then,* $[a_{ij}]$ *is called the* fpfs*-matrix of* α *and is defined by*

$$[a_{ij}] := \begin{bmatrix} a_{01} & a_{02} & a_{03} & \dots & a_{0n} & \dots \\ a_{11} & a_{12} & a_{13} & \dots & a_{1n} & \dots \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{m1} & a_{m2} & a_{m3} & \dots & a_{mn} & \dots \\ \vdots & \vdots & \vdots & \ddots & \vdots & \ddots \end{bmatrix}$$

such that for $i \in \{0, 1, 2, \cdots\}$ *and* $j \in \{1, 2, \cdots\}$,

$$a_{ij} := \begin{cases} \mu(x_j), & i = 0 \\ \alpha({}^{\mu(x_j)}x_j)(u_i), & i \neq 0 \end{cases}$$

Here, if $|U| = m - 1$ *and* $|E| = n$*, then* $[a_{ij}]$ *has order* $m \times n$.

Hereinafter, the set of all *fpfs*-matrices parameterized via E over U is denoted by $FPFS_E[U]$ and let $[a_{ij}], [b_{ij}], [c_{ij}] \in FPFS_E[U]$, $I_m := \{1, 2, 3, \dots, m\}$, and $I_m^* := \{0, 1, 2, \dots, m\}$.

Example 4: The fpfs*-matrices of* α *and* β *provided in Example 2 are as follows*:

$$[a_{ij}] = \begin{bmatrix} 1 & 0.2 & 0.4 \\ 0.5 & 1 & 1 \\ 0.2 & 0.1 & 0.5 \\ 0.4 & 0.8 & 1 \end{bmatrix} \quad \text{and}$$

$$[b_{ij}] = \begin{bmatrix} 0.1 & 0.9 & 0.7 \\ 0.6 & 0.8 & 1 \\ 1 & 1 & 0.6 \\ 0.7 & 0.3 & 0.3 \end{bmatrix}$$

Definition 5 [8]: Let $[a_{ij}] \in FPFS_E[U]$*. For all i and j, if* $a_{ij} = \lambda$*, then* $[a_{ij}]$ *is called* λ-fpfs*-matrix and is denoted by* $[\lambda]$*. Here,* $[0]$ *and* $[1]$ *are called empty* fpfs*-matrix and universal* fpfs*-matrix, respectively.*

Definition 6 [8]: Let $[a_{ij}], [b_{ij}] \in FPFS_E[U]$*. For all i and j,*

If $a_{ij} = b_{ij}$*, then* $[a_{ij}]$ *and* $[b_{ij}]$ *are called equal* fpfs*-matrices and is denoted by* $[a_{ij}] = [b_{ij}]$.

If $a_{ij} \leq b_{ij}$*, then* $[a_{ij}]$ *is called a submatrix of* $[b_{ij}]$ *and is denoted by* $[a_{ij}] \tilde{\subseteq} [b_{ij}]$.

If $[a_{ij}] \tilde{\subseteq} [b_{ij}]$ *and* $[a_{ij}] \neq [b_{ij}]$*, then* $[a_{ij}]$ *is called a proper submatrix of* $[b_{ij}]$ *and is denoted by* $[a_{ij}] \tilde{\subsetneq} [b_{ij}]$.

Example 7: Let E and U be as in Example 4 and let $[c_{ij}] \in FPFS_E[U]$ *such that*

$$[c_{ij}] = \begin{bmatrix} 1 & 1 & 0.8 \\ 0.9 & 1 & 1 \\ 1 & 1 & 0.7 \\ 0.8 & 0.9 & 1 \end{bmatrix}$$

Then, $[a_{ij}] \tilde{\subseteq} [c_{ij}]$, $[b_{ij}] \tilde{\subseteq} [c_{ij}]$*, and* $[a_{ij}] \tilde{\nsubseteq} [b_{ij}]$.

III. DISTANCE MEASURES OF FUZZY PARAMETERIZED FUZZY SOFT MATRICES

In this section, we first define concepts of quasi-metric, semi-metric, pseudo-metric, and metric over $FPFS_E[U]$. Our goals herein are both to contribute theoretically to the soft set theory and to avail of *fpfs*-matrices in classification problems in machine learning. The metrics and similarities of *fpfs*-matrices yield advantages of using the modeling ability of *fpfs*-matrices.

Definition 8: Let $d : FPFS_E[U] \times FPFS_E[U] \to \mathbb{R}$ *be a mapping. Then, for all* $[a_{ij}], [b_{ij}], [c_{ij}] \in FPFS_E[U]$*, d is quasi-metric over* $FPFS_E[U]$ *if and only if d satisfies the following properties*:

i) $d([a_{ij}], [b_{ij}]) = 0 \Leftrightarrow [a_{ij}] = [b_{ij}]$
ii) $d([a_{ij}], [b_{ij}]) \leq d([a_{ij}], [c_{ij}]) + d([c_{ij}], [b_{ij}])$

Definition 9: Let $d : FPFS_E[U] \times FPFS_E[U] \to \mathbb{R}$ *be a mapping. Then, for all* $[a_{ij}], [b_{ij}], [c_{ij}] \in FPFS_E[U]$*, d is semi-metric over* $FPFS_E[U]$ *if and only if d satisfies the following properties*:

i) $d([a_{ij}], [b_{ij}]) = 0 \Leftrightarrow [a_{ij}] = [b_{ij}]$
ii) $d([a_{ij}], [b_{ij}]) = d([b_{ij}], [a_{ij}])$

Definition 10: Let $d : FPFS_E[U] \times FPFS_E[U] \to \mathbb{R}$ *be a mapping. Then, for all* $[a_{ij}], [b_{ij}], [c_{ij}] \in FPFS_E[U]$*, d is pseudo-metric over* $FPFS_E[U]$ *if and only if d satisfies the following properties*:

i) $d([a_{ij}], [a_{ij}]) = 0$
ii) $d([a_{ij}], [b_{ij}]) = d([b_{ij}], [a_{ij}])$
iii) $d([a_{ij}], [b_{ij}]) \leq d([a_{ij}], [c_{ij}]) + d([c_{ij}], [b_{ij}])$

Definition 11: Let $d : FPFS_E[U] \times FPFS_E[U] \to \mathbb{R}$ *be a mapping. Then, for all* $[a_{ij}], [b_{ij}], [c_{ij}] \in FPFS_E[U]$*, d is metric over* $FPFS_E[U]$ *if and only if d satisfies the following properties*:

i) $d([a_{ij}], [b_{ij}]) = 0 \Leftrightarrow [a_{ij}] = [b_{ij}]$
ii) $d([a_{ij}], [b_{ij}]) = d([b_{ij}], [a_{ij}])$
iii) $d([a_{ij}], [b_{ij}]) \leq d([a_{ij}], [c_{ij}]) + d([c_{ij}], [b_{ij}])$

Secondly, we propose eight pseudo-metrics over $FPFS_E[U]$ by using some distance measures of fuzzy soft sets as given in [37], [38], [41] and present some of their basic properties.

Proposition 12: The mapping d_1 *defined by*

$$d_1([a_{ij}], [b_{ij}]) := \sum_{i=1}^{m-1} \sum_{j=1}^{n} |a_{0j}a_{ij} - b_{0j}b_{ij}|$$

is a pseudo-metric over $FPFS_E[U]$ *and is called Hamming pseudo-metric. Moreover, the normalized Hamming* *pseudo-metric is as follows*:

$$\hat{d}_1([a_{ij}],[b_{ij}]) := \frac{1}{(m-1)n}\sum_{i=1}^{m-1}\sum_{j=1}^{n}|a_{0j}a_{ij}-b_{0j}b_{ij}|$$

Proposition 13: The mapping d_2 defined by

$$d_2([a_{ij}],[b_{ij}]) := \max_{i\in I_{m-1}}\left\{\max_{j\in I_n}\{|a_{0j}a_{ij}-b_{0j}b_{ij}|\}\right\}$$

is a pseudo-metric over $FPFS_E[U]$ and is called Chebyshev pseudo-metric.

Proposition 14: The mapping d_3 defined by

$$d_3([a_{ij}],[b_{ij}]) := \left(\sum_{i=1}^{m-1}\sum_{j=1}^{n}|a_{0j}a_{ij}-b_{0j}b_{ij}|^2\right)^{\frac{1}{2}}$$

is a pseudo-metric over $FPFS_E[U]$ and is called Euclidean pseudo-metric. Moreover, the normalized Euclidean pseudo-metric is as follows:

$$\hat{d}_3([a_{ij}],[b_{ij}]) := \frac{1}{\sqrt{(m-1)n}}\left(\sum_{i=1}^{m-1}\sum_{j=1}^{n}|a_{0j}a_{ij}-b_{0j}b_{ij}|^2\right)^{\frac{1}{2}}$$

Proposition 15: The mapping d_4 defined by

$$d_4([a_{ij}],[b_{ij}]) := \sum_{i=1}^{m-1}\left(\sum_{j=1}^{n}|a_{0j}a_{ij}-b_{0j}b_{ij}|^2\right)^{\frac{1}{2}}$$

is a pseudo-metric over $FPFS_E[U]$ and is called type-2 Euclidean pseudo-metric. Moreover, the normalized type-2 Euclidean pseudo-metric is as follows:

$$\hat{d}_4([a_{ij}],[b_{ij}]) := \frac{1}{(m-1)\sqrt{n}}\sum_{i=1}^{m-1}\left(\sum_{j=1}^{n}|a_{0j}a_{ij}-b_{0j}b_{ij}|^2\right)^{\frac{1}{2}}$$

Proposition 16: The mapping d_5 defined by

$$d_5([a_{ij}],[b_{ij}]) := \sum_{i=1}^{m-1}\max_{j\in I_n}\{|a_{0j}a_{ij}-b_{0j}b_{ij}|\}$$

is a pseudo-metric over $FPFS_E[U]$ and is called Hausdorff pseudo-metric. Moreover, the normalized Hausdorff pseudo-metric is as follows:

$$\hat{d}_5([a_{ij}],[b_{ij}]) := \frac{1}{m-1}\sum_{i=1}^{m-1}\max_{j\in I_n}\{|a_{0j}a_{ij}-b_{0j}b_{ij}|\}$$

Proposition 17: The mapping d_6^p defined by

$$d_6^p([a_{ij}],[b_{ij}]) := \left(\sum_{i=1}^{m-1}\sum_{j=1}^{n}|a_{0j}a_{ij}-b_{0j}b_{ij}|^p\right)^{\frac{1}{p}}, \quad p\in\mathbb{N}^+$$

is a pseudo-metric over $FPFS_E[U]$ and is called Minkowski pseudo-metric. Moreover, the normalized Minkowski pseudo-metric is as follows:

$$\hat{d}_6^p([a_{ij}],[b_{ij}]) := \frac{1}{\sqrt[p]{(m-1)n}}\left(\sum_{i=1}^{m-1}\sum_{j=1}^{n}|a_{0j}a_{ij}-b_{0j}b_{ij}|^p\right)^{\frac{1}{p}}$$

such that $p\in\mathbb{N}^+$

Proposition 18: The mapping d_7^p defined by

$$d_7^p([a_{ij}],[b_{ij}]) := \sum_{i=1}^{m-1}\left(\sum_{j=1}^{n}|a_{0j}a_{ij}-b_{0j}b_{ij}|^p\right)^{\frac{1}{p}}, \quad p\in\mathbb{N}^+$$

is a pseudo-metric over $FPFS_E[U]$ and is called type-2 Minkowski pseudo-metric. Moreover, the normalized type-2 Minkowski pseudo-metric is as follows:

$$\hat{d}_7^p([a_{ij}],[b_{ij}]) := \frac{1}{(m-1)\sqrt[p]{n}}\sum_{i=1}^{m-1}\left(\sum_{j=1}^{n}|a_{0j}a_{ij}-b_{0j}b_{ij}|^p\right)^{\frac{1}{p}}$$

such that $p\in\mathbb{N}^+$

Proposition 19: The mapping d_8^p defined by

$$d_8^p([a_{ij}],[b_{ij}]) := \left(\sum_{i=1}^{m-1}\max_{j\in I_n}\{|a_{0j}a_{ij}-b_{0j}b_{ij}|^p\}\right)^{\frac{1}{p}}, \quad p\in\mathbb{N}^+$$

is a pseudo-metric over $FPFS_E[U]$ and is called generalized Hausdorff pseudo-metric. Moreover, the normalized generalized Hausdorff pseudo-metric is as follows:

$$\hat{d}_8^p([a_{ij}],[b_{ij}]) := \frac{1}{\sqrt[p]{(m-1)}}\left(\sum_{i=1}^{m-1}\max_{j\in I_n}\{|a_{0j}a_{ij}-b_{0j}b_{ij}|^p\}\right)^{\frac{1}{p}}$$

such that $p\in\mathbb{N}^+$

Proposition 20: For all $[a_{ij}],[b_{ij}]\in FPFS_E[U]$ and $p\in\mathbb{N}^+$,

i. $d_1([a_{ij}],[b_{ij}])\leq(m-1)n$
ii. $d_2([a_{ij}],[b_{ij}])\leq 1$
iii. $d_3([a_{ij}],[b_{ij}])\leq\sqrt{(m-1)n}$
iv. $d_4([a_{ij}],[b_{ij}])\leq(m-1)\sqrt{n}$
v. $d_5([a_{ij}],[b_{ij}])\leq(m-1)$
vi. $d_6^p([a_{ij}],[b_{ij}])\leq\sqrt[p]{(m-1)n}$
vii. $d_7^p([a_{ij}],[b_{ij}])\leq(m-1)\sqrt[p]{n}$
viii. $d_8^p([a_{ij}],[b_{ij}])\leq\sqrt[p]{m-1}$

Proof: The proof is straight forward from the proofs of Proposition 12-19. □

Proposition 21: Let us consider the pseudo-metrics mentioned above. Then, the following conditions are held for all $[a_{ij}],[b_{ij}]\in FPFS_E[U]$, $k\in\{1,2,3,4,5\}$, $t\in\{6,7,8\}$, $p,r\in\mathbb{N}^+$, and $p\leq r$.

i. $d_k([0],[1])=1$ *and* $d_t^p([0],[1])=1$
ii. $d_t^p([a_{ij}],[b_{ij}])\leq d_t^r([a_{ij}],[b_{ij}])$

Proposition 22: For all $[a_{ij}],[b_{ij}]\in FPFS_E[U]$,

i. $d_1([a_{ij}],[b_{ij}])=d_6^1([a_{ij}],[b_{ij}])=d_7^1([a_{ij}],[b_{ij}])$

ii. $d_3([a_{ij}],[b_{ij}]) = d_6^2([a_{ij}],[b_{ij}])$
iii. $d_4([a_{ij}],[b_{ij}]) = d_7^2([a_{ij}],[b_{ij}])$
iv. $d_5([a_{ij}],[b_{ij}]) = d_8^1([a_{ij}],[b_{ij}])$

Proposition 23: For all $[a_{ij}],[b_{ij}] \in FPFS_E[U]$ *and* $p \in \mathbb{N}^+$,

i. $[a_{ij}]\tilde{\subseteq}[b_{ij}]\tilde{\subseteq}[c_{ij}] \Rightarrow \big(d_1([a_{ij}],[b_{ij}]) \le d_1([a_{ij}],[c_{ij}]) \wedge d_1([b_{ij}],[c_{ij}]) \le d_1([a_{ij}],[c_{ij}])\big)$
ii. $[a_{ij}]\tilde{\subseteq}[b_{ij}]\tilde{\subseteq}[c_{ij}] \Rightarrow \big(d_2([a_{ij}],[b_{ij}]) \le d_2([a_{ij}],[c_{ij}]) \wedge d_2([b_{ij}],[c_{ij}]) \le d_2([a_{ij}],[c_{ij}])\big)$
iii. $[a_{ij}]\tilde{\subseteq}[b_{ij}]\tilde{\subseteq}[c_{ij}] \Rightarrow \big(d_3([a_{ij}],[b_{ij}]) \le d_3([a_{ij}],[c_{ij}]) \wedge d_3([b_{ij}],[c_{ij}]) \le d_3([a_{ij}],[c_{ij}])\big)$
iv. $[a_{ij}]\tilde{\subseteq}[b_{ij}]\tilde{\subseteq}[c_{ij}] \Rightarrow \big(d_4([a_{ij}],[b_{ij}]) \le d_4([a_{ij}],[c_{ij}]) \wedge d_4([b_{ij}],[c_{ij}]) \le d_4([a_{ij}],[c_{ij}])\big)$
v. $[a_{ij}]\tilde{\subseteq}[b_{ij}]\tilde{\subseteq}[c_{ij}] \Rightarrow \big(d_5([a_{ij}],[b_{ij}]) \le d_5([a_{ij}],[c_{ij}]) \wedge d_5([b_{ij}],[c_{ij}]) \le d_5([a_{ij}],[c_{ij}])\big)$
vi. $[a_{ij}]\tilde{\subseteq}[b_{ij}]\tilde{\subseteq}[c_{ij}] \Rightarrow \big(d_6^p([a_{ij}],[b_{ij}]) \le d_6^p([a_{ij}],[c_{ij}]) \wedge d_6^p([b_{ij}],[c_{ij}]) \le d_6^p([a_{ij}],[c_{ij}])\big)$
vii. $[a_{ij}]\tilde{\subseteq}[b_{ij}]\tilde{\subseteq}[c_{ij}] \Rightarrow \big(d_7^p([a_{ij}],[b_{ij}]) \le d_7^p([a_{ij}],[c_{ij}]) \wedge d_7^p([b_{ij}],[c_{ij}]) \le d_7^p([a_{ij}],[c_{ij}])\big)$
viii. $[a_{ij}]\tilde{\subseteq}[b_{ij}]\tilde{\subseteq}[c_{ij}] \Rightarrow \big(d_8^p([a_{ij}],[b_{ij}]) \le d_8^p([a_{ij}],[c_{ij}]) \wedge d_8^p([b_{ij}],[c_{ij}]) \le d_8^p([a_{ij}],[c_{ij}])\big)$

Example 24: For $[a_{ij}]$ *and* $[b_{ij}]$ *provided in Example 4,*

$$
\begin{array}{ll}
d_1([a_{ij}],[b_{ij}]) = 3.0900 & \hat{d}_1([a_{ij}],[b_{ij}]) = 0.3433 \\
d_2([a_{ij}],[b_{ij}]) = 0.8800 & d_3([a_{ij}],[b_{ij}]) = 1.2425 \\
\hat{d}_3([a_{ij}],[b_{ij}]) = 0.4142 & d_4([a_{ij}],[b_{ij}]) = 2.0532 \\
\hat{d}_4([a_{ij}],[b_{ij}]) = 0.3951 & d_5([a_{ij}],[b_{ij}]) = 1.7300 \\
\hat{d}_5([a_{ij}],[b_{ij}]) = 0.5767 & d_6^3([a_{ij}],[b_{ij}]) = 0.9967 \\
\hat{d}_6^3([a_{ij}],[b_{ij}]) = 0.4791 & d_7^3([a_{ij}],[b_{ij}]) = 1.8707 \\
\hat{d}_7^3([a_{ij}],[b_{ij}]) = 0.4323 & d_8^3([a_{ij}],[b_{ij}]) = 0.9502 \\
\hat{d}_8^3([a_{ij}],[b_{ij}]) = 0.6589 &
\end{array}
$$

IV. DISTANCE-BASED SIMILARITY MEASURES OF FUZZY PARAMETERIZED FUZZY SOFT MATRICES

In this section, we first define concepts of quasi-similarity, semi-similarity, pseudo-similarity, and similarity over $FPFS_E[U]$ using pseudo-metrics of *fpfs*-matrices provided in Section III. Thus, the modeling success of pseudo-metrics of *fpfs*-matrices can be transferred to the classification problems in machine learning.

Definition 25: Let $s : FPFS_E[U] \times FPFS_E[U] \to \mathbb{R}$ *be a mapping. Then, for all* $[a_{ij}],[b_{ij}],[c_{ij}] \in FPFS_E[U]$, *s is quasi-similarity over* $FPFS_E[U]$ *if and only if s satisfies the following properties*:

i) $s([a_{ij}],[b_{ij}]) = 1 \Leftrightarrow [a_{ij}] = [b_{ij}]$
ii) $0 \le s([a_{ij}],[b_{ij}]) \le 1$

Definition 26: Let $s : FPFS_E[U] \times FPFS_E[U] \to \mathbb{R}$ *be a mapping. Then, for all* $[a_{ij}],[b_{ij}],[c_{ij}] \in FPFS_E[U]$, *s is semi-similarity over* $FPFS_E[U]$ *if and only if s satisfies the following properties*:

i) $s([a_{ij}],[b_{ij}]) = 1 \Leftrightarrow [a_{ij}] = [b_{ij}]$
ii) $s([a_{ij}],[b_{ij}]) = s([b_{ij}],[a_{ij}])$

Definition 27: Let $s : FPFS_E[U] \times FPFS_E[U] \to \mathbb{R}$ *be a mapping. Then, for all* $[a_{ij}],[b_{ij}],[c_{ij}] \in FPFS_E[U]$, *s is pseudo-similarity over* $FPFS_E[U]$ *if and only if s satisfies the following properties*:

i) $s([a_{ij}],[a_{ij}]) = 1$
ii) $s([a_{ij}],[b_{ij}]) = s([b_{ij}],[a_{ij}])$
iii) $0 \le s([a_{ij}],[b_{ij}]) \le 1$

Definition 28: Let $s : FPFS_E[U] \times FPFS_E[U] \to \mathbb{R}$ *be a mapping. Then, for all* $[a_{ij}],[b_{ij}],[c_{ij}] \in FPFS_E[U]$, *s is similarity over* $FPFS_E[U]$ *if and only if s satisfies the following properties*:

i) $s([a_{ij}],[b_{ij}]) = 1 \Leftrightarrow [a_{ij}] = [b_{ij}]$
ii) $s([a_{ij}],[b_{ij}]) = s([b_{ij}],[a_{ij}])$
iii) $0 \le s([a_{ij}],[b_{ij}]) \le 1$

Secondly, we propose eight pseudo-similarities over $FPFS_E[U]$ by using the pseudo-metrics of *fpfs*-matrices available in Section III and provide some of their basic properties.

Proposition 29 [51]: The mapping s_1 *defined by*

$$s_1([a_{ij}],[b_{ij}]) := 1 - \frac{1}{(m-1)n}\sum_{i=1}^{m-1}\sum_{j=1}^{n}|a_{0j}a_{ij} - b_{0j}b_{ij}|$$

is a pseudo-similarity over $FPFS_E[U]$ *and is called Hamming pseudo-similarity.*

Proof: The proof is straight forward from the proof of Proposition 12. □

Proposition 30 [52]: The mapping s_2 *defined by*

$$s_2([a_{ij}],[b_{ij}]) := 1 - \max_{i\in I_{m-1}}\left\{\max_{j\in I_n}\{|a_{0j}a_{ij} - b_{0j}b_{ij}|\}\right\}$$

is a pseudo-similarity over $FPFS_E[U]$ *and is called Chebyshev pseudo-similarity.*

Proof: The proof is straight forward from the proof of Proposition 13. □

Proposition 31: The mapping s_3 *defined by*

$$s_3([a_{ij}],[b_{ij}]) := 1 - \frac{1}{\sqrt{(m-1)n}}\left(\sum_{i=1}^{m-1}\sum_{j=1}^{n}|a_{0j}a_{ij} - b_{0j}b_{ij}|^2\right)^{\frac{1}{2}}$$

is a pseudo-similarity over $FPFS_E[U]$ *and is called Euclidean pseudo-similarity.*

Proof: The proof is straight forward from the proof of Proposition 14. □

Proposition 32: The mapping s_4 *defined by*

$$s_4([a_{ij}],[b_{ij}]) := 1 - \frac{1}{(m-1)\sqrt{n}}\sum_{i=1}^{m-1}\left(\sum_{j=1}^{n}|a_{0j}a_{ij} - b_{0j}b_{ij}|^2\right)^{\frac{1}{2}}$$

is a pseudo-similarity over $FPFS_E[U]$ *and is called type-2 Euclidean pseudo-similarity.*

Proof: The proof is straight forward from the proof of Proposition 15. □

Proposition 33: The mapping s_5 *defined by*

$$s_5([a_{ij}],[b_{ij}]) := 1 - \frac{1}{m-1}\sum_{i=1}^{m-1}\max_{j\in I_n}\{|a_{0j}a_{ij} - b_{0j}b_{ij}|\}$$

is a pseudo-similarity over $FPFS_E[U]$ and is called Hausdorff pseudo-similarity.

Proof: The proof is straight forward from the proof of Proposition 16. □

Proposition 34: The mapping s_6^p defined by

$$s_6^p([a_{ij}],[b_{ij}]):=1-\frac{1}{\sqrt[p]{(m-1)n}}\left(\sum_{i=1}^{m-1}\sum_{j=1}^{n}|a_{0j}a_{ij}-b_{0j}b_{ij}|^p\right)^{\frac{1}{p}}$$

is a pseudo-similarity over $FPFS_E[U]$ and is called Minkowski pseudo-similarity. Here $p \in \mathbb{N}^+$.

Proof: The proof is straight forward from the proof of Proposition 17. □

Proposition 35: The mapping s_7^p defined by

$$s_7^p([a_{ij}],[b_{ij}]) \coloneqq 1-\frac{1}{(m-1)\sqrt[p]{n}}\sum_{i=1}^{m-1}\left(\sum_{j=1}^{n}|a_{0j}a_{ij}-b_{0j}b_{ij}|^p\right)^{\frac{1}{p}}$$

is a pseudo-similarity over $FPFS_E[U]$ and is called type-2 Minkowski pseudo-similarity. Here $p \in \mathbb{N}^+$.

Proof: The proof is straight forward from the proof of Proposition 18. □

Proposition 36: The mapping s_8^p defined by

$$s_8^p([a_{ij}],[b_{ij}]) \coloneqq 1-\frac{1}{\sqrt[p]{(m-1)}}\left(\sum_{i=1}^{m-1}\max_{j\in I_n}\{|a_{0j}a_{ij}-b_{0j}b_{ij}|^p\}\right)^{\frac{1}{p}}$$

is a pseudo-similarity over $FPFS_E[U]$ and is called generalized Hausdorff pseudo-similarity. Here $p \in \mathbb{N}^+$.

Proof: The proof is straight forward from the proof of Proposition 19. □

Proposition 37: Let $[a_{ij}],[b_{ij}] \in FPFS_E[U]$. Then, for all $[a_{ij}],[b_{ij}]$, $k \in \{1,2,3,4,5\}$, $t \in \{6,7,8\}$, $p, r \in \mathbb{N}^+$, and $p \le r$,

- *i.* $s_k([0],[1]) = 0$ *and* $s_t^p([0],[1]) = 0$
- *ii.* $s_t^p([a_{ij}],[b_{ij}]) \ge s_t^r([a_{ij}],[b_{ij}])$

Proposition 38: For all $[a_{ij}],[b_{ij}] \in FPFS_E[U]$,

- *i.* $s_1([a_{ij}],[b_{ij}]) = s_6^1([a_{ij}],[b_{ij}]) = s_7^1([a_{ij}],[b_{ij}])$
- *ii.* $s_3([a_{ij}],[b_{ij}]) = s_6^2([a_{ij}],[b_{ij}])$
- *iii.* $s_4([a_{ij}],[b_{ij}]) = s_7^2([a_{ij}],[b_{ij}])$
- *iv.* $s_5([a_{ij}],[b_{ij}]) = s_8^1([a_{ij}],[b_{ij}])$

Proposition 39: For all $[a_{ij}],[b_{ij}],[c_{ij}] \in FPFS_E[U]$ and $p \in \mathbb{N}^+$,

- *i.* $[a_{ij}]\tilde{\subseteq}[b_{ij}]\tilde{\subseteq}[c_{ij}] \Rightarrow \big(s_1([a_{ij}],[c_{ij}]) \le s_1([a_{ij}],[b_{ij}]) \wedge s_1([a_{ij}],[c_{ij}]) \le s_1([b_{ij}],[c_{ij}])\big)$
- *ii.* $[a_{ij}]\tilde{\subseteq}[b_{ij}]\tilde{\subseteq}[c_{ij}] \Rightarrow \big(s_2([a_{ij}],[c_{ij}]) \le s_2([a_{ij}],[b_{ij}]) \wedge s_2([a_{ij}],[c_{ij}]) \le s_2([b_{ij}],[c_{ij}])\big)$
- *iii.* $[a_{ij}]\tilde{\subseteq}[b_{ij}]\tilde{\subseteq}[c_{ij}] \Rightarrow \big(s_3([a_{ij}],[c_{ij}]) \le s_3([a_{ij}],[b_{ij}]) \wedge s_3([a_{ij}],[c_{ij}]) \le s_3([b_{ij}],[c_{ij}])\big)$
- *iv.* $[a_{ij}]\tilde{\subseteq}[b_{ij}]\tilde{\subseteq}[c_{ij}] \Rightarrow \big(s_4([a_{ij}],[c_{ij}]) \le s_4([a_{ij}],[b_{ij}]) \wedge s_4([a_{ij}],[c_{ij}]) \le s_4([b_{ij}],[c_{ij}])\big)$
- *v.* $[a_{ij}]\tilde{\subseteq}[b_{ij}]\tilde{\subseteq}[c_{ij}] \Rightarrow \big(s_5([a_{ij}],[c_{ij}]) \le s_5([a_{ij}],[b_{ij}]) \wedge s_5([a_{ij}],[c_{ij}]) \le s_5([b_{ij}],[c_{ij}])\big)$
- *vi.* $[a_{ij}]\tilde{\subseteq}[b_{ij}]\tilde{\subseteq}[c_{ij}] \Rightarrow \big(s_6^p([a_{ij}],[c_{ij}]) \le s_6^p([a_{ij}],[b_{ij}]) \wedge s_6^p([a_{ij}],[c_{ij}]) \le s_6^p([b_{ij}],[c_{ij}])\big)$
- *vii.* $[a_{ij}]\tilde{\subseteq}[b_{ij}]\tilde{\subseteq}[c_{ij}] \Rightarrow \big(s_7^p([a_{ij}],[c_{ij}]) \le s_7^p([a_{ij}],[b_{ij}]) \wedge s_7^p([a_{ij}],[c_{ij}]) \le s_7^p([b_{ij}],[c_{ij}])\big)$
- *viii.* $[a_{ij}]\tilde{\subseteq}[b_{ij}]\tilde{\subseteq}[c_{ij}] \Rightarrow \big(s_8^p([a_{ij}],[c_{ij}]) \le s_8^p([a_{ij}],[b_{ij}]) \wedge s_8^p([a_{ij}],[c_{ij}]) \le s_8^p([b_{ij}],[c_{ij}])\big)$

Proof: The other proofs are straight forward from the proof of Proposition 23. □

Example 40: For $[a_{ij}]$ and $[b_{ij}]$ provided in Example 4,

$s_1([a_{ij}],[b_{ij}]) = 0.6567$ $\quad$ $s_2([a_{ij}],[b_{ij}]) = 0.1200$
$s_3([a_{ij}],[b_{ij}]) = 0.5858$ $\quad$ $s_4([a_{ij}],[b_{ij}]) = 0.6049$
$s_5([a_{ij}],[b_{ij}]) = 0.4233$ $\quad$ $s_6^3([a_{ij}],[b_{ij}]) = 0.5209$
$s_7^3([a_{ij}],[b_{ij}]) = 0.5677$ $\quad$ $s_8^3([a_{ij}],[b_{ij}]) = 0.3411$

V. FUZZY PARAMETERIZED FUZZY SOFT EUCLIDEAN CLASSIFIER (FPFS-EC)

In this section, we first present the definitions and notations occurring in FPFS-EC. Across the present paper, let $D = [d_{ij}]_{m\times(n+1)}$ denotes a data matrix and its last column contains class labels of the data. Here, m and n stand for the number of the samples and the number of the attributes in the data matrix, respectively. $(D_{train})_{m_1\times n}$, $(C)_{m_1\times 1}$, and $(D_{test})_{m_2\times n}$ represent the training matrix, class labels of the train matrix, and the test matrix obtained from D, respectively such that $m_1 + m_2 = m$. $D_{i-train}$ and D_{i-test} denote i^{th} row of D_{train} and D_{test}, respectively. Similarly, $D_{train-j}$ and D_{test-j} denote j^{th} column of D_{train} and D_{test}, respectively. $T_{m_2\times 1}$ stands for assigned class matrix obtained from D_{train} and D_{test}. Let I_m denote the set of all unsigned integer numbers from 1 to m, i.e. $I_m := \{1,2,\ldots,m\}$. Similarly, let $I_m^* := \{0,1,2,\ldots,m\}$.

Definition 41: Let $u, v \in \mathbb{R}^n$. Then, the Pearson correlation coefficient between u and v is defined by

$$P(u,v) := \frac{n\sum_{i=1}^n u_iv_i-(\sum_{i=1}^n u_i)(\sum_{i=1}^n v_i)}{\sqrt{\left[n\sum_{i=1}^n u_i^2-(\sum_{i=1}^n u_i)^2\right]\left[n\sum_{i=1}^n v_i^2-(\sum_{i=1}^n v_i)^2\right]}}$$

Definition 42: Let D_{train} has order $m_1 \times n$ and $C_{m_1\times 1}$ be the class column vector of D_{train}. fw is called the feature weight vector based on the Pearson correlation coefficient of D_{train} and is defined by

$$fw_{j1} := \left|P\left(D_{train-j}, C\right)\right|, \quad j \in I_n$$

Definition 43: Let D_{train} has order $m_1 \times n$ and D_{test} has order $m_2 \times n$. $\widetilde{D}_{train}$ is called the feature fuzzifications of D_{train} and is defined by

$$\widetilde{d}_{ij-train} := \frac{d_{ij-train}-\min\limits_{r,s}\{d_{rj-train},d_{sj-test}\}}{\max\limits_{r,s}\{d_{rj-train},d_{sj-test}\}-\min\limits_{r,s}\{d_{rj-train},d_{sj-test}\}}$$

such that $i, r \in I_{m_1}$, $s \in I_{m_2}$, and $j \in I_n$.

Definition 44: Let D_{train} has order $m_1 \times n$ and D_{test} has order $m_2 \times n$. $\widetilde{D}_{test}$ is called the feature fuzzifications of D_{test}, and is defined by

$$\widetilde{d}_{ij-test} := \frac{d_{ij-test}-\min\limits_{r,s}\{d_{rj-train},d_{sj-test}\}}{\max\limits_{r,s}\{d_{rj-train},d_{sj-test}\}-\min\limits_{r,s}\{d_{rj-train},d_{sj-test}\}}$$

such that $r \in I_{m_1}$, $i, s \in I_{m_2}$, and $j \in I_n$.

We then propose a new classification algorithm, i.e. FPFS-EC, via Euclidean pseudo-similarity defined in Section IV. FPFS-EC uses the Pearson correlation coefficient to obtain feature weight based on parameters' impact on classification. After that, it constructs two *fpfs*-matrices, i.e. train *fpfs*-matrix and test *fpfs*-matrix, via normalized train sample, normalized test sample, and feature weights. Next, the proposed classifier assigns the class label of the train sample, whose Euclidean pseudo-similarity to the test sample is at the highest level, to the test sample. This process proceeds similarly for all the test samples. Finally, the assigned class matrix of the test data is constructed. Its algorithm steps (Algorithm 1) and flowchart (Fig. 1) are as follows:

Algorithm 1 FPFS-EC's Pseudocode

Input: $(D_{train})_{m_1\times n}$, $C_{m_1\times 1}$, and $(D_{test})_{m_2\times n}$
Output: $T_{m_2\times 1}$

procedure FPFSEC(D_{train}, C, D_{test})
  Compute *fw* using D_{train} and C
  Compute feature fuzzification of D_{train} and D_{test}, i.e., $\widetilde{D}_{train}$ and $\widetilde{D}_{test}$
  for i from 1 to m_2 **do**
    Compute the test *fpfs*-matrix $[a_{ij}]$ using *fw* and $\widetilde{D}_{i-test}$
    for j from 1 to m_1 **do**
      Compute the train *fpfs*-matrix $[b_{ij}]$ using *fw* and $\widetilde{D}_{j-train}$
      $sm_{j1} \leftarrow s_3([a_{ij}], [b_{ij}])$ ▷ $[sm_{j1}]$ represents similarity matrix
    end for
    $w \leftarrow \underset{j\in I_{m_1}}{\operatorname{argmax}} \{sm_{j1}\}$
    $t_{i1} \leftarrow$ the class of w
  end for
  return $T_{m_2\times 1}$
end procedure

VI. EXPERIMENTAL STUDY

This section presents the properties of the 18 classification datasets in the UCI Machine Learning Repository [50]. We then offer five performance metrics for performance evaluation in machine learning. Next, we perform some experiments to show that our proposed method is more efficient than SVM [49], Fuzzy kNN [47], FSSC [44], FussCyier [45], HDFSSC [46], and BM-Fuzzy kNN [48]. Finally, we provide the statistical evaluation of the experimental results based on the Friedman test [53] and the Nemenyi post-hoc test [54].

A. UCI DATASETS AND PERFORMANCE MEASURES

In Table 2, we firstly present the properties of the datasets [50] used in the simulation herein: "Breast Cancer Wisconsin (Diagnostic)", "Breast Tissue", "Diabetic Retinopathy Debrecen", "Immunotherapy", "Breast Cancer Coimbra", "Parkinsons[sic]", "Connectionist Bench (Sonar, Mines vs. Rocks)", "Wine", "Statlog (German Credit Data)", "Hayes-Roth", "Iris", "Mice Protein Expression", "Parkinson's Disease", "Teaching Assistant Evaluation", "Vehicle", "Semeion Handwritten Digit", "Ionosphere", and "Connectionist Bench (Vowel Recognition-Deterding Data)".

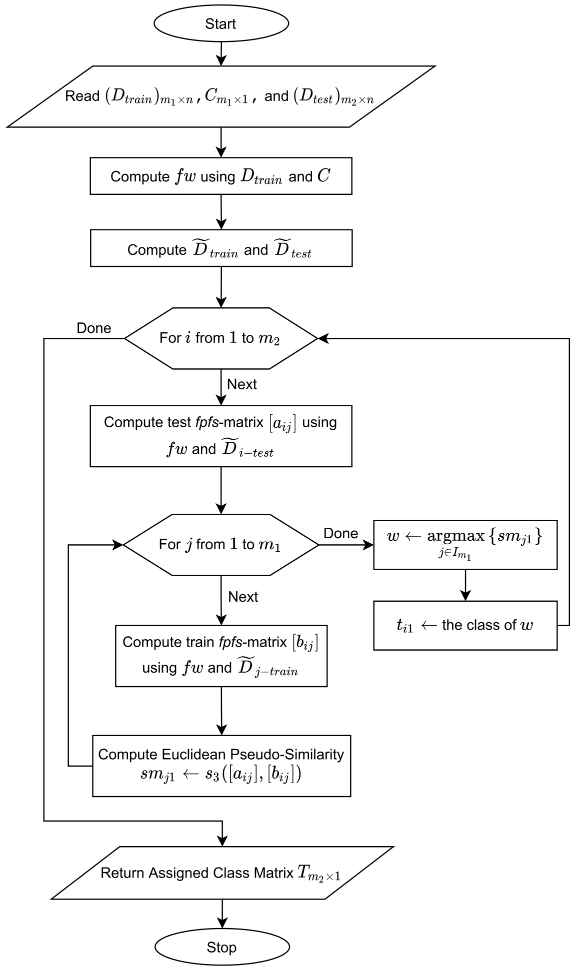

FIGURE 1. The flowchart of FPFS-EC.

We subsequently provide the mathematical notations of five performance metrics, i.e. accuracy (Acc), precision (Pre), recall (Rec), macro F-score (MacF), and micro F-score (MicF), to compare the aforementioned methods. Let $X = \{x_1, x_2, \dots, x_n\}$, $\mathbb{Y} = \{\mathbb{Y}_1, \mathbb{Y}_2, \dots, \mathbb{Y}_n\}$, $\hat{\mathbb{Y}} = \{\hat{\mathbb{Y}}_1, \hat{\mathbb{Y}}_2, \dots, \hat{\mathbb{Y}}_n\}$, and l be n samples to be classified, ground truth class sets of the samples, prediction class sets of the samples, and the number of the class of the samples, respectively.

$$\mathrm{Acc}(\mathbb{Y}, \hat{\mathbb{Y}}) := \frac{1}{l}\sum_{i=1}^{l} \frac{TP_i + TN_i}{TP_i + TN_i + FP_i + FN_i}$$

$$\mathrm{Pre}(\mathbb{Y}, \hat{\mathbb{Y}}) := \frac{1}{l}\sum_{i=1}^{l} \frac{TP_i}{TP_i + FP_i}$$

$$\mathrm{Rec}(\mathbb{Y}, \hat{\mathbb{Y}}) := \frac{1}{l}\sum_{i=1}^{l} \frac{TP_i}{TP_i + FN_i}$$

TABLE 2. Description of UCI data sets.

No.	Name	#Instance	#Attribute	#Class
1	Wisconsin	569	30	2
2	Breast Tissue	106	9	6
3	Diabetic Retinopathy	1151	19	2
4	Immunotherapy	90	7	2
5	Coimbra	116	9	2
6	Parkinsons[sic]	195	22	2
7	Sonar	208	60	2
8	Wine	178	13	3
9	German Credit	1000	20	2
10	Hayes-Roth	132	5	3
11	Iris	150	4	3
12	Mice	1077	72	8
13	Parkinson's Disease	756	754	2
14	Teaching	151	5	3
15	Vehicle	846	17	4
16	Semeion	1593	265	2
17	Ionosphere	351	34	2
18	Vowel	990	13	11

stands for the number of.

$$\mathrm{MacF}(\mathbb{Y}, \hat{\mathbb{Y}}) := \frac{1}{l}\sum_{i=1}^{l} \frac{2TP_i}{2TP_i + FP_i + FN_i}$$

$$\mathrm{MicF}(\mathbb{Y}, \hat{\mathbb{Y}}) := \frac{2\sum_{i=1}^{l} TP_i}{2\sum_{i=1}^{l} TP_i + \sum_{i=1}^{l} FP_i + \sum_{i=1}^{l} FN_i}$$

where TP_i, TN_i, FP_i, and FN_i are the number of true positive, true negative, false positive, and false negative for the class i, respectively and their mathematical notations are as follows:

$$TP_i := \left|\left\{x_t \mid i \in \mathbb{Y}_t \wedge i \in \hat{\mathbb{Y}}_t, 1 \le t \le l\right\}\right|$$
$$TN_i := \left|\left\{x_t \mid i \notin \mathbb{Y}_t \wedge i \notin \hat{\mathbb{Y}}_t, 1 \le t \le l\right\}\right|$$
$$FP_i := \left|\left\{x_t \mid i \notin \mathbb{Y}_t \wedge i \in \hat{\mathbb{Y}}_t, 1 \le t \le l\right\}\right|$$
$$FN_i := \left|\left\{x_t \mid i \in \mathbb{Y}_t \wedge i \notin \hat{\mathbb{Y}}_t, 1 \le t \le l\right\}\right|$$

B. SIMULATION RESULTS

In this part of the present study, we focus on the comparison between our proposed FPFS-EC and the well-known methods, i.e. SVM [49] and Fuzzy kNN [47], and other the state-of-the-art classifiers based on fuzzy sets or soft sets, i.e. FSSC [44], FussCyier [45], HDFSSC [46], and BM-Fuzzy kNN [48]. We simulate the algorithms by utilizing MATLAB R2020b and a workstation with I(R) Xeon(R) CPU E5-1620 v4 @ 3.5 GHz and 64 GB RAM. Each classifier is trained and tested by means of the k-fold cross-validation [55], [56].

In the simulation, we carry out 5-fold cross-validation and record the mean results for 5 iterations. In each iteration in cross-validation, the training and testing phase is carried out independently from other stages. Finally, We repeat this process 30 times and obtain the mean accuracy, precision, recall, macro F-score, micro F-score, and running time results.

Table 3 presents accuracy, precision, recall, macro F-score, micro F-score, and running time results of the methods for "Wisconsin", "Breast Tissue", "Diabetic Retinopathy", "Immunotherapy", "Coimbra", "Parkinsons[sic]", "Sonar", "Wine", "German Credit", "Hayes-Roth", "Iris", "Mice Protein", "Parkinson's Disease", "Teaching", "Vehicle", "Semeion", "Ionosphere", and "Vowel" datasets. In "Wisconsin", "Parkinsons[sic]", "Wine", "Parkinson's Disease", "Semeion", "Ionosphere", and "Vowel" datasets, FPSEC exhibits the best performance by about 95% in terms of all the performance metrics. Especially in "Parkinsons[sic]", "Parkinson's Disease", "Hayes-Roth", "Vowel" datasets, FPFS-EC outperforms the others to a great extent. In the case of improving FPFS-EC, FPFS-EC is believed to be capable of exhibiting better performance in these four datasets. In the other datasets too, where the overall performance results are not over 90%, FPFS-EC outperforms the others. Besides, in "Mice Protein" dataset, the performance of FPFS-EC, just as of SVM and HDFSSC, is 100% as far as the performance metrics are concerned.

FPFS-EC achieves remarkable classification success thanks to its using Euclidean pseudo-similarity of *fpfs*-matrices based on the Pearson correlation coefficient and evaluating all the train samples separately. On the other hand, evaluating all the train samples separately results in FPFS-EC's running slightly slower than the others. Although FPFS-EC, in general, seems to operate slightly slower than the other classifiers except for SVM and BM-Fuzzy kNN, classifying all the test samples in a considered dataset takes about from 0.00414 to 2.17023 seconds.

Table 4 provides the scores concerning the performance advantages of FPFS-EC over the other classifiers for all the datasets. The results show that FPFS-EC produces the best scores in the datasets in terms of accuracy, precision, recall, macro F-score, and micro F-score performance. In Table 4, FPFS-EC performs notably better in "Parkinsons[sic]", "Parkinson's Disease", and "Hayes-Roth", datasets than the others do, just as FPFS-EC in Table 3. For example, in "Parkinson's Disease" dataset, the accuracy, precision, recall, macro F-score, and micro F-score values concerning its performance advantages over the classifier with the nearest score are 19.24%, 17.50%, 32.75%, 6.40%, and 19.24%, respectively. Similarly, the values are 11.99%, 15.70%, 16.06%, 16.10%, and 17.98% in "Hayes-Roth" dataset and 8.51%, 7.03%, 18.16%, 14.54%, and 8.51% in "Parkinsons[sic]" dataset.

Figure 2 presents the graphical results concerning the accuracy, precision, recall, macro F-score, micro F-score, and running time performances of the compared classifiers in Table 3. As the figure reveals, FPFS-EC outperforms SVM,

TABLE 3. Comparative results for the datasets.

Datasets	Classifiers	Acc±SD	Pre ±SD	Rec±SD	MacF±SD	MicF±SD	Running Time±SD
Wisconsin	SVM	95.30±1.92	95.09±2.03	94.71±2.23	95.00±2.08	95.30±1.92	2.26340±0.04540
	Fuzzy kNN	91.58±2.37	91.56±2.50	90.45±2.81	90.87±2.62	91.58±2.37	0.00605±0.00113
	FSSC	93.63±1.93	93.44±2.15	93.00±2.13	93.15±2.07	93.63±1.93	0.00126±0.00315
	FussCyier	93.60±1.83	94.38±1.85	92.07±2.27	92.98±2.05	93.60±1.83	0.00053±0.00080
	HDFSSC	92.90±1.91	93.10±2.09	91.75±2.24	92.29±2.10	92.90±1.91	0.00075±0.00108
	BM-Fuzzy kNN	91.86±2.10	91.41±2.28	91.35±2.34	91.30±2.25	91.86±2.10	0.16708±0.00818
	FPFS-EC	**95.34±1.51**	**95.18±1.67**	**94.92±1.70**	**95.01±1.62**	**95.34±1.51**	0.18110±0.00619
Breast Tissue	SVM	88.18±3.12	66.19±10.81	63.47±9.64	68.20±8.32	64.15±9.35	2.57618±0.63478
	Fuzzy kNN	84.39±3.49	57.24±10.89	51.81±10.72	57.51±8.89	53.17±10.46	0.00043±0.00023
	FSSC	87.76±2.87	63.73±10.26	61.69±8.79	66.35±8.28	63.28±8.61	0.00044±0.00021
	FussCyier	86.75±2.85	61.40±9.04	58.85±8.95	64.28±8.31	60.25±8.56	0.00020±0.00018
	HDFSSC	87.34±2.80	66.26±9.47	61.07±8.43	64.15±8.02	62.01±8.40	0.00029±0.00018
	BM-Fuzzy kNN	87.25±3.37	63.14±12.06	60.24±10.65	64.32±8.44	61.74±10.12	0.01225±0.00407
	FPFS-EC	**88.25±2.96**	**66.67±9.29**	**63.57±9.26**	**70.37±7.56**	**64.75±8.88**	0.00445±0.00171
Diabetic Retinopathy	SVM	**74.22±2.88**	**75.17±2.88**	**74.73±2.86**	**74.17±2.90**	**74.22±2.88**	6.33758±0.18442
	Fuzzy kNN	62.04±2.63	61.99±2.63	61.97±2.62	61.91±2.63	62.04±2.63	0.01790±0.00053
	FSSC	57.43±3.01	57.66±3.04	57.63±3.03	57.40±3.01	57.43±3.01	0.00172±0.00006
	FussCyier	57.02±2.93	57.32±2.96	57.28±2.94	56.98±2.93	57.02±2.93	0.00067±0.00002
	HDFSSC	56.88±3.11	57.05±3.13	57.04±3.13	56.84±3.12	56.88±3.11	0.00110±0.00004
	BM-Fuzzy kNN	64.90±2.93	64.84±2.96	64.75±2.93	64.71±2.94	64.90±2.93	0.21462±0.00991
	FPFS-EC	65.54±2.64	65.63±2.66	65.62±2.64	65.48±2.63	65.54±2.64	0.73343±0.03136
Immunotherapy	SVM	**80.67±6.90**	**77.07±20.89**	58.79±10.62	**77.76±10.84**	**80.67±6.90**	0.31092±0.00080
	Fuzzy kNN	60.82±8.57	43.02±8.05	43.58±8.94	63.63±14.06	60.82±8.57	0.00022±0.00008
	FSSC	61.06±10.50	61.38±7.52	65.89±10.70	56.72±9.38	61.06±10.50	0.00021±0.00001
	FussCyier	67.02±10.16	62.89±9.48	67.48±12.24	61.38±10.30	67.02±10.16	0.00011±0.00001
	HDFSSC	65.20±10.29	61.96±8.94	66.79±12.27	60.05±10.21	65.20±10.29	0.00015±0.00001
	BM-Fuzzy kNN	64.15±10.11	47.45±11.39	46.91±10.33	63.17±13.45	64.15±10.11	0.00734±0.00144
	FPFS-EC	75.91±9.57	65.49±15.46	**67.56±13.34**	67.02±12.42	75.91±9.57	0.00447±0.00029
Coimbra	SVM	**72.43±7.68**	**73.27±7.91**	**72.73±7.86**	**72.08±7.86**	**72.43±7.68**	0.83737±0.02639
	Fuzzy kNN	54.03±9.32	53.52±9.65	53.30±9.20	52.77±9.37	54.03±9.32	0.00039±0.00023
	FSSC	62.76±8.48	67.33±8.90	64.69±8.06	61.71±9.09	62.76±8.48	0.00021±0.00016
	FussCyier	61.35±8.37	67.21±9.03	63.73±7.88	59.69±9.49	61.35±8.37	0.00013±0.00012
	HDFSSC	59.60±9.32	62.56±10.28	61.10±9.16	58.71±9.83	59.60±9.32	0.00016±0.00011
	BM-Fuzzy kNN	53.75±9.17	53.95±9.72	53.84±9.24	53.11±9.11	53.75±9.17	0.01118±0.00136
	FPFS-EC	68.24±9.53	68.73±9.80	68.11±9.54	67.68±9.70	68.24±9.53	0.00414±0.00059
Parkinsons[sic]	SVM	86.56±4.19	86.29±7.07	76.09±7.02	78.94±7.20	86.56±4.19	0.82134±0.19197
	Fuzzy kNN	84.23±4.94	79.50±6.87	79.32±7.44	78.76±6.73	84.23±4.94	0.00070±0.00011
	FSSC	73.83±5.81	72.58±4.24	79.82±5.45	71.47±5.62	73.83±5.81	0.00040±0.00003
	FussCyier	74.12±5.76	73.29±3.82	80.81±4.84	71.95±5.47	74.12±5.76	0.00019±0.00003
	HDFSSC	78.41±5.52	75.08±4.65	82.14±5.42	75.54±5.61	78.41±5.52	0.00026±0.00004
	BM-Fuzzy kNN	79.50±5.77	73.28±7.04	74.81±8.16	73.36±7.42	79.50±5.77	0.04341±0.00233
	FPFS-EC	**95.08±3.55**	**93.32±4.81**	**94.25±4.83**	**93.48±4.64**	**95.08±3.55**	0.02123±0.00091
Sonar	SVM	78.03±5.56	79.09±5.79	77.51±5.59	77.52±5.69	78.03±5.56	0.01522±0.00041
	Fuzzy kNN	82.24±5.80	82.81±5.67	81.91±5.99	81.93±6.04	82.24±5.80	0.00159±0.00012
	FSSC	75.08±7.11	75.79±7.24	74.56±7.23	74.50±7.40	75.08±7.11	0.00047±0.00005
	FussCyier	71.58±7.15	73.25±7.35	72.24±7.13	71.36±7.22	71.58±7.15	0.00024±0.00001
	HDFSSC	70.12±7.98	70.46±8.06	70.17±8.07	69.89±8.11	70.12±7.98	0.00033±0.00002
	BM-Fuzzy kNN	82.82±5.62	83.30±5.56	82.76±5.68	82.65±5.73	82.82±5.62	0.10805±0.00374
	FPFS-EC	**87.15±4.58**	**87.88±4.44**	**86.80±4.72**	**86.93±4.74**	**87.15±4.58**	0.02591±0.00096
Wine	SVM	97.08±1.91	95.85±2.78	95.79±2.76	95.65±2.83	95.61±2.87	0.45670±0.18431
	Fuzzy kNN	85.29±3.58	62.32±12.30	71.93±6.63	72.30±6.77	73.20±6.38	0.00049±0.00011
	FSSC	96.34±2.37	94.97±3.02	95.41±2.96	94.75±3.44	94.52±3.56	0.00044±0.00002
	FussCyier	96.38±2.47	94.92±3.41	95.30±3.26	94.80±3.58	94.57±3.70	0.00017±0.00001
	HDFSSC	95.46±2.61	93.65±3.70	93.93±3.56	93.48±3.79	93.19±3.91	0.00027±0.00001
	BM-Fuzzy kNN	82.57±4.52	73.73±7.31	73.06±6.68	72.54±6.98	73.85±6.79	0.02270±0.00145
	FPFS-EC	**97.55±2.00**	**96.50±2.71**	**96.93±2.51**	**96.47±2.92**	**96.32±3.01**	0.01753±0.00091
German Credit	SVM	58.01±11.94	55.76±6.79	55.40±6.53	53.39±8.85	58.01±11.94	6.37604±0.37067
	Fuzzy kNN	61.20±2.89	53.71±3.20	53.68±3.21	53.61±3.20	61.20±2.89	0.01384±0.00118
	FSSC	63.45±3.28	62.13±3.00	62.27±3.57	61.22±3.23	63.45±3.28	0.00156±0.00024
	FussCyier	63.58±3.29	62.13±3.02	64.24±3.59	61.28±3.25	63.58±3.29	0.00061±0.00009
	HDFSSC	64.94±3.56	62.01±3.37	63.82±3.90	61.84±3.60	64.94±3.56	0.00097±0.00015
	BM-Fuzzy kNN	62.15±3.01	54.87±3.26	54.80±3.23	54.73±3.23	62.15±3.01	0.19410±0.00559
	FPFS-EC	**69.18±2.94**	**63.23±3.46**	**64.98±3.36**	**63.02±3.37**	**69.18±2.94**	0.55650±0.02603

Acc, Pre, Rec, MacF, and MicF results and their standard deviations (SD) are presented in percentage. Running time and its SD are presented in seconds. The best performances are shown in bold.

TABLE 3. *(Continued.)* **Comparative results for the datasets.**

Datasets	Classifiers	Acc±SD	Pre ±SD	Rec±SD	MacF±SD	MicF±SD	Running Time±SD
Hayes-Roth	SVM	73.84±4.86	66.50±6.88	62.07±7.61	62.87±7.11	60.75±7.29	0.12284±0.11463
	Fuzzy kNN	65.65±7.08	33.63±12.51	35.86±7.36	40.54±7.10	39.39±7.43	0.00032±0.00010
	FSSC	70.08±7.02	58.27±11.56	57.55±10.14	54.99±10.03	55.12±10.53	0.00035±0.00002
	FussCyier	72.65±8.10	62.66±12.65	59.15±11.84	59.24±11.63	58.98±12.15	0.00014±0.00001
	HDFSSC	69.35±6.11	57.43±9.17	56.71±9.25	56.03±8.93	54.02±9.17	0.00021±0.00001
	BM-Fuzzy kNN	61.43±5.44	45.06±13.43	39.40±8.01	42.02±7.95	42.14±8.16	0.00897±0.00257
	FPFS-EC	**85.82±5.01**	**82.19±7.25**	**78.13±7.76**	**78.98±7.67**	**78.74±7.52**	0.00962±0.00062
Iris	SVM	**98.18±1.83**	**97.54±2.51**	**97.27±2.75**	**97.25±2.77**	**97.27±2.75**	0.06465±0.00406
	Fuzzy kNN	95.09±1.94	69.70±11.04	89.53±4.50	91.97±3.95	89.53±4.50	0.00036±0.00011
	FSSC	96.80±2.39	95.55±3.39	95.20±3.59	95.18±3.62	95.20±3.59	0.00037±0.00002
	FussCyier	96.95±2.38	95.80±3.32	95.42±3.57	95.40±3.60	95.42±3.57	0.00015±0.00001
	HDFSSC	97.33±2.24	96.31±3.15	96.00±3.37	95.98±3.39	96.00±3.37	0.00022±0.00001
	BM-Fuzzy kNN	96.98±2.17	95.87±3.05	95.47±3.26	95.44±3.28	95.47±3.26	0.00795±0.00129
	FPFS-EC	97.35±2.15	96.40±2.91	96.02±3.22	96.00±3.26	96.02±3.22	0.01200±0.00047
Mice Protein	SVM	**100.00±0.00**	**100.00±0.00**	**100.00±0.00**	**100.00±0.00**	**100.00±0.00**	1.73284 ±0.02842
	Fuzzy kNN	99.12±0.16	56.74±4.12	93.19±1.39	96.22±0.85	93.78±1.27	0.02220±0.00226
	FSSC	98.67±0.42	94.98±1.58	94.88±1.63	94.81±1.65	94.67±1.67	0.00619±0.00172
	FussCyier	98.74±0.41	95.30±1.55	95.20±1.59	95.12±1.62	94.97±1.64	0.00143±0.00026
	HDFSSC	**100.00±0.00**	**100.00±0.00**	**100.00±0.00**	**100.00±0.00**	**100.00±0.00**	0.00329±0.00098
	BM-Fuzzy kNN	99.98±0.04	99.94±0.17	99.93±0.17	99.93±0.17	99.93±0.17	0.75664±0.03485
	FPFS-EC	**100.00±0.00**	**100.00±0.00**	**100.00±0.00**	**100.00±0.00**	**100.00±0.00**	0.73020±0.03058
Parkinson's Disease	SVM	74.60±0.29	74.60±0.29	50.00±0.00	85.45±0.19	74.60±0.29	0.03864±0.00430
	Fuzzy kNN	69.05±2.97	59.17±3.70	59.08±3.77	59.00±3.71	69.05±2.97	0.18199±0.00516
	FSSC	38.50±6.37	47.35±4.27	47.51±4.28	37.89±5.94	38.50±6.37	0.01119±0.00014
	FussCyier	61.54±16.49	45.99±5.32	48.54±1.97	44.51±13.16	61.54±16.49	0.00993±0.00013
	HDFSSC	61.83±16.52	46.52±6.21	48.63±2.17	44.67±13.09	61.83±16.52	0.01052±0.00013
	BM-Fuzzy kNN	50.58±3.75	51.06±3.10	51.38±4.04	47.53±3.38	50.58±3.75	5.89811±0.16920
	FPFS-EC	**93.84±1.81**	**92.11±2.66**	**91.83±2.74**	**91.86±2.39**	**93.84±1.81**	0.74882±0.01312
Teaching	SVM	68.57±5.23	54.51±7.89	53.02±7.83	51.73±8.38	52.85±7.85	0.14484±0.04950
	Fuzzy kNN	76.16±5.38	50.30±10.52	57.77±7.98	58.16±8.18	57.79±8.04	0.00040±0.00012
	FSSC	63.35±5.38	48.90±13.79	45.62±8.07	44.21±8.47	45.03±8.07	0.00038±0.00002
	FussCyier	63.40±5.37	49.00±12.92	45.68±8.05	43.65±8.08	45.11±8.06	0.00016±0.00001
	HDFSSC	69.55±5.07	55.74±8.25	54.53±7.60	53.67±7.76	54.33±7.60	0.00023±0.00002
	BM-Fuzzy kNN	61.12±5.49	41.78±8.79	41.74±8.19	41.05±8.28	41.68±8.23	0.01093±0.00259
	FPFS-EC	**76.76±5.45**	**65.44±8.59**	**64.10±8.18**	**63.77±8.36**	**64.23±8.17**	0.01224±0.00042
Vehicle	SVM	**89.14±1.23**	**78.55±2.56**	**78.51±2.46**	**78.30±2.50**	**78.29±2.47**	3.96660±0.58215
	Fuzzy kNN	84.35±1.16	42.25±2.37	59.73±3.18	61.05±3.04	59.27±3.20	0.00952±0.00060
	FSSC	69.67±1.63	39.82±4.58	40.10±3.28	36.49±3.70	39.35±3.25	0.00218±0.00020
	FussCyier	69.79±1.69	40.15±4.94	40.36±3.40	36.47±3.92	39.57±3.38	0.00060±0.00004
	HDFSSC	70.48±1.69	42.06±4.11	41.62±3.40	39.31±3.60	40.96±3.39	0.00120±0.00012
	BM-Fuzzy kNN	84.28±1.54	67.69±3.06	68.13±3.04	67.69±3.03	67.55±3.07	0.13934±0.00483
	FPFS-EC	86.08±1.44	67.77±2.91	68.50±2.84	67.97±2.85	68.16±2.87	0.39590±0.02085
Semeion	SVM	97.47±0.69	94.94±2.51	89.50±2.55	92.42±1.98	97.47±0.69	0.15675±0.03688
	Fuzzy kNN	96.52±2.27	91.96±12.87	88.05±4.49	89.78±4.32	96.52±4.49	0.30063±0.00010
	FSSC	44.03±2.45	57.50±3.52	68.86±3.67	40.52±3.67	44.03±3.67	0.01041±0.00003
	FussCyier	76.15±2.47	64.02±3.55	84.36±3.71	64.48±3.70	76.15±3.71	0.00786±0.00001
	HDFSSC	89.51±2.63	73.54±3.76	88.28±3.94	78.06±3.94	89.51±3.94	0.00858±0.00002
	BM-Fuzzy kNN	97.04±0.74	95.94±1.07	85.40±3.68	90.38±2.75	97.04±0.74	3.52961±0.15261
	FPFS-EC	**97.51±2.78**	**96.19±4.20**	**89.55±4.16**	**92.45±4.28**	**97.51±4.16**	2.17023±0.00129
Ionosphere	SVM	86.90±3.65	88.83±3.95	83.06±4.56	84.71±4.43	86.90±3.65	0.02434±0.00151
	Fuzzy kNN	84.70±3.40	88.64±3.31	79.36±4.56	81.39±4.63	84.70±3.40	0.00337±0.00064
	FSSC	64.10±0.36	64.10±0.36	50.00±0.00	78.13±0.27	64.10±0.36	0.00055±0.00019
	FussCyier	64.10±0.36	64.10±0.36	50.00±0.00	78.13±0.27	64.10±0.36	0.00029±0.00011
	HDFSSC	64.10±0.36	64.10±0.36	50.00±0.00	78.13±0.27	64.10±0.36	0.00040±0.00014
	BM-Fuzzy kNN	81.17±4.02	82.04±5.05	76.54±4.84	77.87±4.95	81.17±4.02	0.13471±0.01031
	FPFS-EC	**89.56±3.22**	**91.81±2.70**	**86.06±4.35**	**87.82±4.04**	**89.56±3.22**	0.03336±0.00467
Vowel	SVM	96.37±0.41	81.10±2.20	80.02±2.27	79.91±2.29	80.02±2.27	2.98539±0.04392
	Fuzzy kNN	99.20±0.27	95.84±1.43	95.63±1.50	95.57±1.52	95.63±1.50	0.01196±0.00144
	FSSC	89.40±0.57	44.71±4.25	41.72±3.14	40.27±3.28	41.72±3.14	0.00442±0.00086
	FussCyier	90.47±0.54	48.24±3.48	47.60±3.00	46.05±3.17	47.60±3.00	0.00088±0.00017
	HDFSSC	90.40±0.61	47.68±3.83	47.22±3.33	45.81±3.52	47.22±3.33	0.00240±0.00041
	BM-Fuzzy kNN	85.36±0.47	18.78±3.43	19.46±2.60	22.18±3.16	19.46±2.60	0.16896±0.00793
	FPFS-EC	**99.67±0.18**	**98.28±0.90**	**98.16±0.97**	**98.16±0.97**	**98.16±0.97**	0.21829±0.01606
Mean Performance Results	SVM	84.20±3.57	80.02±5.32	75.70±4.73	79.19±4.79	79.62±4.47	1.62398±0.13914
	Fuzzy kNN	79.76±3.79	65.22±6.87	69.23±5.35	71.50±5.42	72.68±5.01	0.03180±0.00079
	FSSC	72.55±4.00	66.68±5.37	66.47±4.98	64.43±5.12	64.60±5.16	0.00238±0.00040
	FussCyier	75.84±4.59	67.34±5.45	67.68±5.01	66.54±5.65	68.14±5.78	0.00135±0.00011
	HDFSSC	76.86±4.57	68.08±5.14	68.38±4.96	68.03±5.50	69.51±5.65	0.00174±0.00019
	BM-Fuzzy kNN	77.05±3.90	66.90±5.71	65.55±5.39	66.89±5.36	68.32±4.98	0.63533±0.02357
	FPFS-EC	**87.16±3.41**	**82.94±4.80**	**81.95±4.78**	**82.36±4.63**	**83.54±4.34**	0.32663±0.00872

Acc, Pre, Rec, MacF, and MicF results and their standard deviations (SD) are presented in percentage. Running time and its SD are presented in seconds. The best performances are shown in bold.

TABLE 4. FPFS-EC's performance advantages over the other classifiers for the datasets.

Datasets	Classifiers	Acc	Pre	Rec	MacF	MicF
Wisconsin	FPFS-EC versus SVM	0.05	0.09	0.21	0.00	0.05
	FPFS-EC versus Fuzzy kNN	3.76	3.63	4.47	4.14	3.76
	FPFS-EC versus FSSC	1.72	1.75	1.92	1.85	1.72
	FPFS-EC versus FussCyier	1.74	0.80	2.85	2.03	1.74
	FPFS-EC versus HDFSSC	2.44	2.09	3.17	2.72	2.44
	FPFS-EC versus BM-Fuzzy kNN	3.48	3.78	3.57	3.71	3.48
Breast Tissue	FPFS-EC versus SVM	0.07	0.48	0.10	2.17	0.60
	FPFS-EC versus Fuzzy kNN	3.86	9.44	11.76	12.86	11.58
	FPFS-EC versus FSSC	0.49	2.94	1.88	4.02	1.47
	FPFS-EC versus FussCyier	1.50	5.27	4.72	6.09	4.50
	FPFS-EC versus HDFSSC	0.91	0.41	2.50	6.22	2.74
	FPFS-EC versus BM-Fuzzy kNN	1.00	3.53	3.33	6.05	3.01
Diabetic Retinopathy	FPFS-EC versus SVM	-8.68	-9.54	-9.11	-8.69	-8.68
	FPFS-EC versus Fuzzy kNN	3.51	3.64	3.65	3.58	3.51
	FPFS-EC versus FSSC	8.11	7.98	7.99	8.09	8.11
	FPFS-EC versus FussCyier	8.52	8.31	8.34	8.50	8.52
	FPFS-EC versus HDFSSC	8.66	8.58	8.58	8.64	8.66
	FPFS-EC versus BM-Fuzzy kNN	0.64	0.80	0.88	0.77	0.64
Immunotherapy	FPFS-EC versus SVM	-4.76	-11.58	22.98	-10.74	-4.76
	FPFS-EC versus Fuzzy kNN	15.09	22.46	23.98	3.39	15.09
	FPFS-EC versus FSSC	14.85	4.11	1.66	10.29	14.85
	FPFS-EC versus FussCyier	8.89	2.59	0.08	5.63	8.89
	FPFS-EC versus HDFSSC	10.71	3.52	0.77	6.97	10.71
	FPFS-EC versus BM-Fuzzy kNN	11.76	18.03	1.77	3.85	11.76
Coimbra	FPFS-EC versus SVM	-4.19	-4.54	-4.62	-4.40	-4.19
	FPFS-EC versus Fuzzy kNN	14.21	15.21	14.81	14.90	14.21
	FPFS-EC versus FSSC	5.48	1.40	3.42	5.97	5.48
	FPFS-EC versus FussCyier	6.89	1.52	4.38	7.99	6.89
	FPFS-EC versus HDFSSC	8.65	6.18	7.01	8.96	8.65
	FPFS-EC versus BM-Fuzzy kNN	14.49	14.78	14.27	14.57	14.49
Parkinsons[sic]	FPFS-EC versus SVM	8.51	7.03	18.16	14.54	8.51
	FPFS-EC versus Fuzzy kNN	10.85	13.82	14.93	14.73	10.85
	FPFS-EC versus FSSC	21.24	20.73	14.43	22.01	21.24
	FPFS-EC versus FussCyier	20.96	20.02	13.43	21.53	20.96
	FPFS-EC versus HDFSSC	16.67	18.24	12.10	17.94	16.67
	FPFS-EC versus BM-Fuzzy kNN	15.57	20.03	19.44	20.12	15.57
Sonar	FPFS-EC versus SVM	9.12	8.79	9.29	9.41	9.12
	FPFS-EC versus Fuzzy kNN	4.91	5.07	4.89	5.00	4.91
	FPFS-EC versus FSSC	12.07	12.09	12.23	12.43	12.07
	FPFS-EC versus FussCyier	15.57	14.63	14.55	15.57	15.57
	FPFS-EC versus HDFSSC	17.03	17.42	16.63	17.04	17.03
	FPFS-EC versus BM-Fuzzy kNN	4.34	4.58	4.04	4.29	4.34
Wine	FPFS-EC versus SVM	0.47	0.65	1.14	0.81	0.71
	FPFS-EC versus Fuzzy kNN	12.26	34.18	24.99	24.16	23.12
	FPFS-EC versus FSSC	1.20	1.53	1.51	1.72	1.81
	FPFS-EC versus FussCyier	1.17	1.58	1.63	1.67	1.75
	FPFS-EC versus HDFSSC	2.09	2.85	2.99	2.99	3.13
	FPFS-EC versus BM-Fuzzy kNN	14.98	22.77	23.87	23.93	22.47
German Credit	FPFS-EC versus SVM	11.17	7.47	9.58	9.64	11.17
	FPFS-EC versus Fuzzy kNN	7.97	9.52	11.30	9.41	7.97
	FPFS-EC versus FSSC	5.73	1.10	2.71	1.80	5.73
	FPFS-EC versus FussCyier	5.60	1.11	0.73	1.74	5.60
	FPFS-EC versus HDFSSC	4.24	1.22	1.16	1.19	4.24
	FPFS-EC versus BM-Fuzzy kNN	7.03	8.37	10.18	8.29	7.03
Hayes-Roth	FPFS-EC versus SVM	11.99	15.70	16.06	16.10	17.98
	FPFS-EC versus Fuzzy kNN	20.17	48.56	42.27	38.44	39.34
	FPFS-EC versus FSSC	15.75	23.93	20.58	23.99	23.62
	FPFS-EC versus FussCyier	13.17	19.54	18.98	19.73	19.76
	FPFS-EC versus HDFSSC	16.48	24.76	21.42	22.95	24.71
	FPFS-EC versus BM-Fuzzy kNN	24.40	37.13	38.73	36.96	36.60
Iris	FPFS-EC versus SVM	-0.83	-1.14	-1.24	-1.26	-1.24
	FPFS-EC versus Fuzzy kNN	2.26	26.70	6.49	4.03	6.49
	FPFS-EC versus FSSC	0.55	0.85	0.82	0.82	0.82
	FPFS-EC versus FussCyier	0.40	0.59	0.60	0.60	0.60
	FPFS-EC versus HDFSSC	0.01	0.09	0.02	0.02	0.02
	FPFS-EC versus BM-Fuzzy kNN	0.37	0.52	0.56	0.55	0.56
Mice Protein	FPFS-EC versus SVM	0.00	0.00	0.00	0.00	0.00
	FPFS-EC versus Fuzzy kNN	0.88	43.26	6.81	3.78	6.22
	FPFS-EC versus FSSC	1.33	5.02	5.12	5.19	5.33
	FPFS-EC versus FussCyier	1.26	4.70	4.80	4.88	5.03
	FPFS-EC versus HDFSSC	0.00	0.00	0.00	0.00	0.00
	FPFS-EC versus BM-Fuzzy kNN	0.02	0.06	0.07	0.07	0.07
Parkinson's Disease	FPFS-EC versus SVM	19.24	17.50	41.83	6.40	19.24
	FPFS-EC versus Fuzzy kNN	24.80	32.94	32.75	32.86	24.80
	FPFS-EC versus FSSC	55.34	44.76	44.32	53.97	55.34
	FPFS-EC versus FussCyier	32.30	46.12	43.29	47.34	32.30
	FPFS-EC versus HDFSSC	32.01	45.58	43.20	47.19	32.01
	FPFS-EC versus BM-Fuzzy kNN	43.27	41.05	40.45	44.33	43.27
Teaching	FPFS-EC versus SVM	8.19	10.94	11.08	12.03	11.37
	FPFS-EC versus Fuzzy kNN	0.60	15.14	6.33	5.61	6.43
	FPFS-EC versus FSSC	13.41	16.54	18.48	19.56	19.20
	FPFS-EC versus FussCyier	13.36	16.45	18.42	20.12	19.12
	FPFS-EC versus HDFSSC	7.21	9.70	9.57	10.10	9.90
	FPFS-EC versus BM-Fuzzy kNN	15.64	23.67	22.36	22.71	22.54
Vehicle	FPFS-EC versus SVM	-3.06	-10.78	-10.01	-10.33	-10.13
	FPFS-EC versus Fuzzy kNN	1.73	25.53	8.76	6.92	8.89
	FPFS-EC versus FSSC	16.41	27.95	28.40	31.48	28.81
	FPFS-EC versus FussCyier	16.29	27.62	28.13	31.49	28.59
	FPFS-EC versus HDFSSC	15.60	25.71	26.87	28.66	27.20
	FPFS-EC versus BM-Fuzzy kNN	1.80	0.08	0.37	0.28	0.61
Semeion	FPFS-EC versus SVM	0.04	1.25	0.05	0.03	0.04
	FPFS-EC versus Fuzzy kNN	0.99	4.23	1.50	2.67	0.99
	FPFS-EC versus FSSC	53.48	38.69	20.69	51.93	53.48
	FPFS-EC versus FussCyier	21.36	32.18	5.19	27.97	21.36
	FPFS-EC versus HDFSSC	8.00	22.66	1.28	14.39	8.00
	FPFS-EC versus BM-Fuzzy kNN	0.47	0.25	4.15	2.07	0.47
Ionosphere	FPFS-EC versus SVM	2.66	2.98	3.01	3.11	2.66
	FPFS-EC versus Fuzzy kNN	4.85	3.17	6.70	6.43	4.85
	FPFS-EC versus FSSC	25.45	27.71	36.06	9.70	25.45
	FPFS-EC versus FussCyier	25.45	27.71	36.06	9.70	25.45
	FPFS-EC versus HDFSSC	25.45	27.71	36.06	9.70	25.45
	FPFS-EC versus BM-Fuzzy kNN	8.39	9.77	9.52	9.96	8.39
Vowel	FPFS-EC versus SVM	3.30	17.18	18.14	18.25	18.14
	FPFS-EC versus Fuzzy kNN	0.46	2.45	2.54	2.59	2.54
	FPFS-EC versus FSSC	10.26	53.57	56.44	57.89	56.44
	FPFS-EC versus FussCyier	9.19	50.04	50.57	52.11	50.57
	FPFS-EC versus HDFSSC	9.26	50.61	50.95	52.35	50.95
	FPFS-EC versus BM-Fuzzy kNN	14.31	79.50	78.70	75.98	78.70
Mean Advantage Scores	FPFS-EC versus SVM	2.96	2.92	7.04	3.17	3.92
	FPFS-EC versus Fuzzy kNN	7.40	17.72	12.72	10.86	10.86
	FPFS-EC versus FSSC	14.60	16.26	15.48	17.93	18.94
	FPFS-EC versus FussCyier	11.31	15.60	14.27	15.82	15.40
	FPFS-EC versus HDFSSC	10.30	14.85	13.57	14.33	14.03
	FPFS-EC versus BM-Fuzzy kNN	10.11	16.04	15.35	15.47	15.22

The results are presented in percentage.

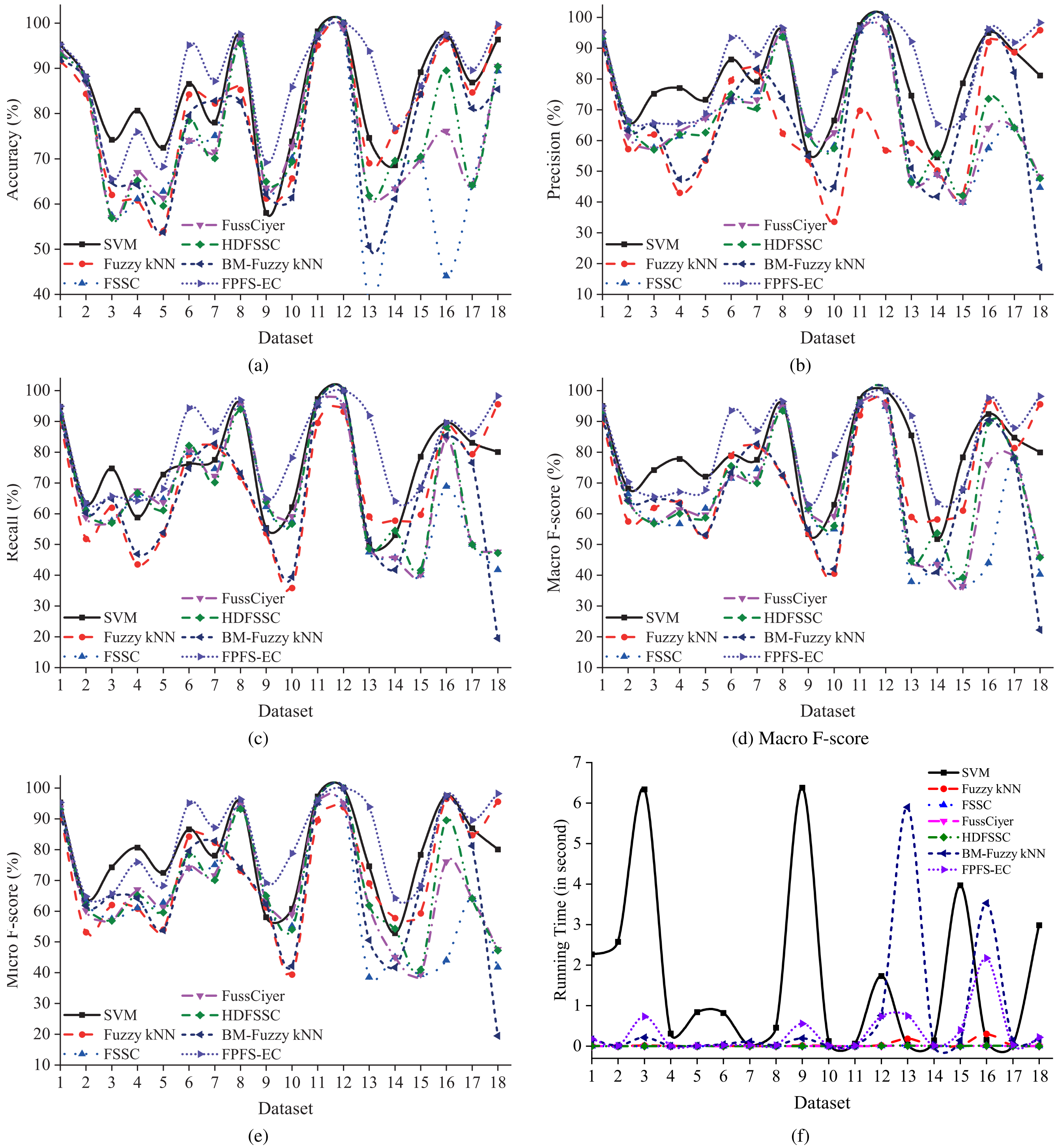

FIGURE 2. **Accuracy (a), Precision (b), Recall (c), Macro F-score (d), Micro F-score (e), and running time (in second) (f) performances of the classifiers related to Table 3.**

Fuzzy kNN, FSSC, FussCyier, HDFSSC, and BM-Fuzzy kNN when operated in the studied datasets except for 3-5, 12, and 15. Although SVM performs better than the others in the datasets 3-5, 12, and 15, FPFS-EC generally produces more reliable classification results than SVM, and the former operates faster than the latter. Moreover, Fuzzy kNN, FSSC, FussCyier, and HDFSSC run faster than SVM and FPFS-EC. However, their performance results are not stable, and they exhibit a lower classification performance compared to SVM and FPFS-EC.

As clear from the mean results in Table 3, 4, and Figure 2, FPFS-EC is a more efficacious method than SVM, Fuzzy kNN, FSSC, FussCyier, HDFSSC, and BM-Fuzzy kNN.

C. STATISTICAL EVALUATION

In this subsection, we employ the corrected Friedman test [53] and the Nemenyi post-hoc test [54] in a manner recommended by [57] to evaluate whether the overall differences in the performance results obtained in view of five performance metrics and running time are statistically significant. The Friedman test, a non-parametric test for multiple hypotheses testing, produces a performance-based ranking of the algorithms for each data set. Thereby, the rank of 1 refers to the best performing algorithm, the rank of 2 to the second best, etc. It assigns average ranks in the event that the ranks of the algorithms are equal.

Afterward, the Friedman test first compares the average ranks of the algorithms and secondly calculates the Friedman statistic χ_F^2, distributed according to the χ_F^2 distribution with $k-1$ degrees of freedom. Here k is the number of algorithms. If a statistically significant difference is detected in the performance, a post-hoc test should be used to detect which difference belong to which algorithm. The Nemenyi test is one of the post-hoc tests commonly used to compare all the classifiers with each other. In this test, if the average ranks of the two algorithms occur more than the critical distance, then the test shows that their performance is considerably different.

We first calculate the average rank of each algorithm considered in our experiments with $k = 7$ and $N = 18$ since the total number of the methods is 7 and the total number of the datasets is 18. If the accuracy, precision, recall, macro F-score, micro F-score, and running time values of the Friedman test statistic are $\chi_F^2 = 55.61$,$\chi_F^2 = 56.15$, $\chi_F^2 = 45.00$, $\chi_F^2 = 54.31$, $\chi_F^2 = 55.25$, and $\chi_F^2 = 98.79$, respectively, with 6 $(k - 1)$ degrees of freedom and the critical value for the Friedman test [53] given for $k = 7$ and $N = 18$ is 12.59 at a significance level of $\alpha = 0.05$, we can conclude that the accuracy ($55.61 > 12.59$), precision ($56.15 > 12.59$), recall ($45.00 > 12.59$), macro F-measure ($54.31 > 12.59$), micro F-measure ($55.25 > 12.59$), and running time ($98.79 > 12.59$) values of the studied methods are significantly different. Now that the null hypothesis is rejected, we can proceed with a post-hoc test. The Nemenyi test [54] can be used when all classifiers are compared with each other [57].

The critical value in our experiments with $k = 7$ and $\alpha = 0.05$ is 2.1228. As a result, the accuracy, precision, recall, macro F-score, and micro F-score of FPFS-EC are significantly different from Fuzzy kNN, FSSC, FussCyier, HDFSSC, and BM-Fuzzy kNN methods, but its running time is not significantly different from that of Fuzzy kNN. Fig. 3 presents the critical diagrams generated by the Nemenyi post-hoc test for the five evaluation measures and running time.

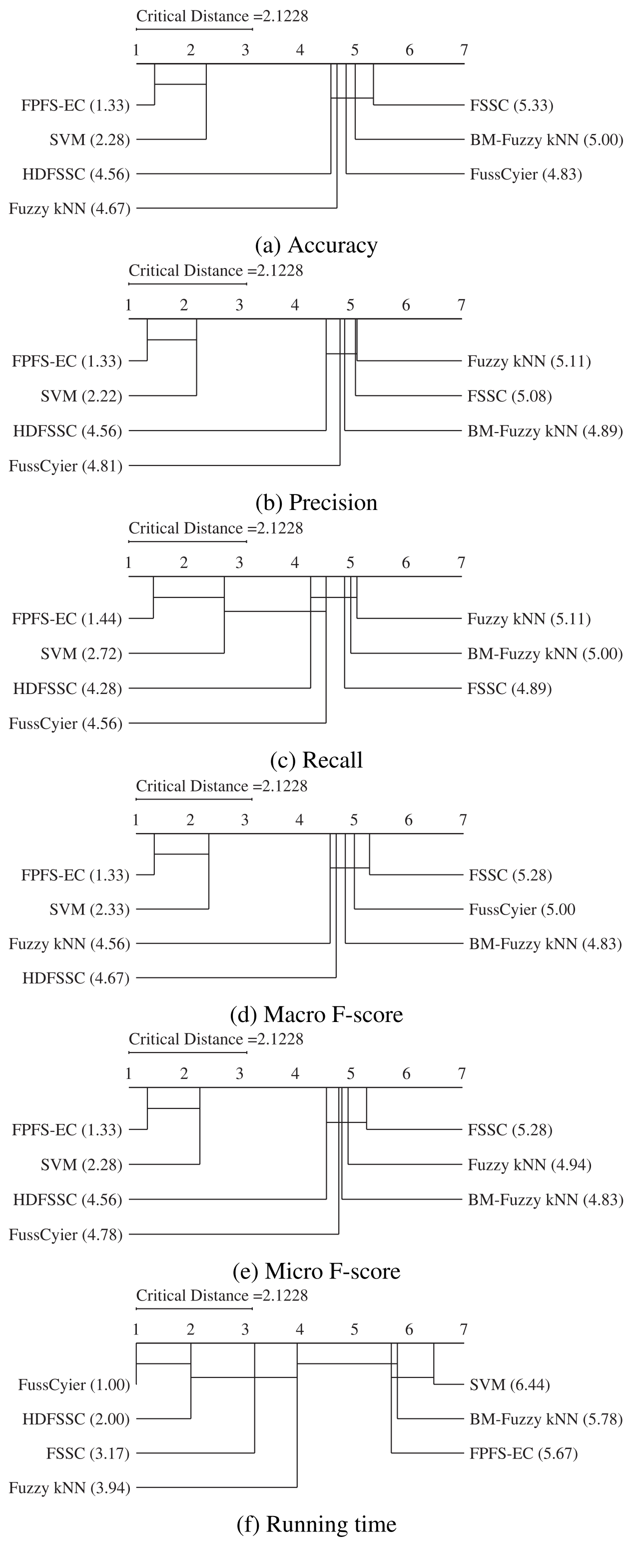

FIGURE 3. The critical diagrams for the five evaluation measures and running time: The results from the Nemenyi post-hoc test at 0.05 significance level and average rank scores from the friedman test.

Fig. 3 shows that the differences between the average ranks of FPFS-EC and those of the others except for SVM are higher than the critical distance of 2.1228 in terms

TABLE 5. Pairwise performance comparison of the classifiers via the friedman test.

	SVM	Fuzzy kNN	FSSC	FussCyier	HDFSSC	BM-Fuzzy kNN	FPFS-EC
SVM	-	+	+	-	-	+	-
Fuzzy kNN	+	-	-	-	-	-	+
FSSC	+	-	-	-	-	-	+
FussCyier	-	-	-	-	-	-	+
HDFSSC	-	-	-	-	-	-	+
BM-Fuzzy kNN	+	-	-	-	-	-	+
FPFS-EC	-	+	+	+	+	+	-

- represents compared classifiers' performances are not significantly different, whereas + stands for they are.

TABLE 6. Computational complexities of the classifiers.

Classifier	Computational Complexity
SVM with kernel	$O(m^3)$
Fuzzy kNN	$O(n^2 \log k)$
FSSC	$O(ml)$
FussCyier	$O(ml)$
HDFSSC	$O(ml)$
BM-Fuzzy kNN	$O(ln^3 \log k)$
FPFS-EC	$O(mn)$

k is number of nearest neighbor, m is the sample number of the training data, n is the parameter number of the training data , and l is the class number of the data.

of all the performance metrics, in contrast to the running time ranks. Besides, Table 5 offers the pairwise comparison between the classifiers obtained via the critical distances in the Friedman test. Fig. 3 and Table 5 manifest that FPFS-EC remarkably outperforms the others in terms of five performance measures.

VII. EVALUATION OF COMPUTATIONAL COMPLEXITY

This section compares the classifiers' computational complexity by utilizing big O notation besides their running time results obtained in 30 runs for the 18 UCI datasets. As can be observed in Table 3, FPFS-EC in general seems to operate faster than SVM and BM-Fuzzy kNN and slightly slower than Fuzzy kNN, FSSC, FussCyier, and HDFSSC. The underlying cause of its slightly slower running than the others is that, in the pre-processing step, FPFS-EC employs all of the training samples while FSSC, FussCyier, and HDFSSC utilize a class-based mean of the training samples. Additionally, FPFS-EC's running time occurs under 1 s for 17 of the 18 datasets (except for "Semeion"). Thanks to its low running time, the proposed classifier can be employed in real-time applications. From the pseudocode of FPFS-EC, the computational complexity is $O(mn)$ for each test sample. Here, m and n are the numbers of the training samples and attributes, respectively. The computational complexities of the compared classifiers are provided in Table 6.

VIII. DISCUSSION

In this section, we discuss FPFS-EC and its classification performance. The subsections Simulation Results and Statistical Evaluation corroborate that FPFS-EC has a classification advantage in the considered datasets over SVM, Fuzzy kNN, FSSC, FussCyier, HDFSSC, and BM-Fuzzy kNN. FPFS-EC's success majorly results from the use of a pseudo-similarity of *fpfs*-matrices – i.e., Euclidean pseudo-similarity – based on parameters' impact. Euclidean pseudo-similarity produces a similarity coefficient utilizing the Pearson correlation between parameters and class labels. This process provides that more significant parameters affect the classification phase more profoundly, whereas less significant parameters exert less effect. The second is that FPFS-EC processes training samples separately. On the other hand, FSSC, FussCyier, and HDFSSC classify the considered test sample employing the averages of the training samples, which causes data loss.

IX. CONCLUSION

This paper defined eight pseudo-metrics of *fpfs*-matrices and eight pseudo-similarities of *fpfs*-matrices based on these pseudo-metrics. Contrary to most of the studies in the literature working on a fictitious problem, we applied the similarity measures of *fpfs*-matrices to actual numerical data classification. In other words, we developed FPFS-EC based on the pseudo-similarity of *fpfs*-matrices for numerical data classification and compared FPFS-EC with SVM [49], FSSC [44], FussCyier [45], HDFSSC [46], Fuzzy kNN [47], and BM-Fuzzy kNN [48]. The results show that FPFS-EC outperforms the other methods and *fpfs*-matrices are more efficacious than fuzzy soft sets for the 18 data sets used herein. This study is believed to inspire new research on constructing *fpfs*-matrices for real-life problems, such as data classification..

However, since *fpfs*-matrices can effectively model classification problems containing uncertainty, further research should be conducted to focus on them. We foresee that one way of improving FPFS-EC is to use different similarity measures of *fpfs*-matrices or define similarity measures of intuitionistic fuzzy parameterized intuitionistic fuzzy soft matrices [58]. Another is to employ different soft decision-making methods constructed by *fpfs*-matrices, such as in [9]–[11], [15]–[21], and [59]. The other is to decrease the negative effects of the unstable data in the datasets herein on classification success.

Finally, it should be stated that when the success of a method is below 90%, the margin of error is unacceptable, particularly in medical decision-making. To overcome this problem and perform a highly reliable diagnosis, considered methods should be customized according to the subject.

AUTHOR CONTRIBUTIONS

Samet Memiş devised the main conceptual ideas and developed the theoretical framework. Uğur Erkan carried out the simulations and statistical analyses. Serdar Enginoğlu encouraged Samet Memiş to investigate distance and similarity measures of *fpfs*-matrices and supervised this work's findings. Samet Memiş and Serdar Enginoğlu wrote the manuscript in consultation with Uğur Erkan. All authors discussed the results and contributed to the final manuscript.

CONFLICT OF INTEREST

The authors declare that they have no conflict of interest.

APPENDIX

Proof [Proposition 12]: For all $[a_{ij}],[b_{ij}],[c_{ij}] \in FPFS_E[U]$,

i. $d_1([a_{ij}],[a_{ij}]) = \sum_{i=1}^{m-1}\sum_{j=1}^{n}|a_{0j}a_{ij}-a_{0j}a_{ij}| = \sum_{i=1}^{m-1}\sum_{j=1}^{n}0=0$

ii. $d_1([a_{ij}],[b_{ij}]) = \sum_{i=1}^{m-1}\sum_{j=1}^{n}|a_{0j}a_{ij}-b_{0j}b_{ij}| = \sum_{i=1}^{m-1}\sum_{j=1}^{n}|b_{0j}b_{ij}-a_{0j}a_{ij}| = d([b_{ij}],[a_{ij}])$

iii.
$$\begin{aligned} d_1([a_{ij}],[b_{ij}]) &= \sum_{i=1}^{m-1}\sum_{j=1}^{n}|a_{0j}a_{ij}-b_{0j}b_{ij}| \\ &= \sum_{i=1}^{m-1}\sum_{j=1}^{n}|a_{0j}a_{ij}-c_{0j}c_{ij} \\ &\quad +c_{0j}c_{ij}-b_{0j}b_{ij}| \\ &\le \sum_{i=1}^{m-1}\sum_{j=1}^{n}|a_{0j}a_{ij}-c_{0j}c_{ij}| \\ &\quad + \sum_{i=1}^{m-1}\sum_{j=1}^{n}|c_{0j}c_{ij}-b_{0j}b_{ij}| \\ &= d_1([a_{ij}],[c_{ij}])+d_1([c_{ij}],[b_{ij}]) \end{aligned}$$
□

Proof [Proposition 13]: For all $[a_{ij}],[b_{ij}],[c_{ij}] \in FPFS_E[U]$,

i. $d_2([a_{ij}],[a_{ij}]) = \max_{i\in I_{m-1}}\left\{\max_{j\in I_n}\{|a_{0j}a_{ij}-a_{0j}a_{ij}|\}\right\} = \max_{i\in I_{m-1}}\left\{\max_{j\in I_n}\{0\}\right\}=0$

ii. $d_2([a_{ij}],[b_{ij}]) = \max_{i\in I_{m-1}}\left\{\max_{j\in I_n}\{|a_{0j}a_{ij}-b_{0j}b_{ij}|\}\right\} = \max_{i\in I_{m-1}}\left\{\max_{j\in I_n}\{|b_{0j}b_{ij}-a_{0j}a_{ij}|\}\right\} = d_2([b_{ij}],[a_{ij}])$

iii.
$$\begin{aligned} d_2([a_{ij}],[b_{ij}]) &= \max_{i\in I_{m-1}}\left\{\max_{j\in I_n}\{|a_{0j}a_{ij}-b_{0j}b_{ij}|\}\right\} \\ &= \max_{i\in I_{m-1}}\left\{\max_{j\in I_n}\{|a_{0j}a_{ij}-c_{0j}c_{ij} \right. \\ &\quad \left. +c_{0j}c_{ij}-b_{0j}b_{ij}|\}\right\} \\ &\le \max_{i\in I_{m-1}}\left\{\max_{j\in I_n}\{|a_{0j}a_{ij}-c_{0j}c_{ij}| \right. \\ &\quad \left. +|c_{0j}c_{ij}-b_{0j}b_{ij}|\}\right\} \\ &\le \max_{i\in I_{m-1}}\left\{\max_{j\in I_n}\{|a_{0j}a_{ij}-c_{0j}c_{ij}|\} \right. \\ &\quad \left. +\max_{j\in I_n}\{|c_{0j}c_{ij}-b_{0j}b_{ij}|\}\right\} \\ &\le \max_{i\in I_{m-1}}\left\{\max_{j\in I_n}\{|a_{0j}a_{ij}-c_{0j}c_{ij}|\}\right\} \\ &\quad +\max_{i\in I_{m-1}}\left\{\max_{j\in I_n}\{|c_{0j}c_{ij}-b_{0j}b_{ij}|\}\right\} \\ &= d_2([a_{ij}],[c_{ij}])+d_2([c_{ij}],[b_{ij}]) \end{aligned}$$
□

Proof [Proposition 14]: For all $[a_{ij}],[b_{ij}],[c_{ij}] \in FPFS_E[U]$,

i. $d_3([a_{ij}],[a_{ij}]) = \left(\sum_{i=1}^{m-1}\sum_{j=1}^{n}|a_{0j}a_{ij}-a_{0j}a_{ij}|^2\right)^{\frac{1}{2}} = \left(\sum_{i=1}^{m-1}\sum_{j=1}^{n}0\right)^{\frac{1}{2}}=0$

ii. $d_3([a_{ij}],[b_{ij}]) = \left(\sum_{i=1}^{m-1}\sum_{j=1}^{n}|a_{0j}a_{ij}-b_{0j}b_{ij}|^2\right)^{\frac{1}{2}} = \left(\sum_{i=1}^{m-1}\sum_{j=1}^{n}|b_{0j}b_{ij}-a_{0j}a_{ij}|^2\right)^{\frac{1}{2}} = d_3([b_{ij}],[a_{ij}])$

iii.
$$\begin{aligned} d_3([a_{ij}],[b_{ij}]) &= \left(\sum_{i=1}^{m-1}\sum_{j=1}^{n}|a_{0j}a_{ij}-b_{0j}b_{ij}|^2\right)^{\frac{1}{2}} \\ &= \left(\sum_{i=1}^{m-1}\sum_{j=1}^{n}|a_{0j}a_{ij}-c_{0j}c_{ij} \right. \\ &\quad \left. +c_{0j}c_{ij}-b_{0j}b_{ij}|^2\right)^{\frac{1}{2}} \\ &\le \left(\sum_{i=1}^{m-1}\sum_{j=1}^{n}\left(|a_{0j}a_{ij}-c_{0j}c_{ij}| \right.\right. \\ &\quad \left.\left. +|c_{0j}c_{ij}-b_{0j}b_{ij}|\right)^2\right)^{\frac{1}{2}} \\ &\le \left(\sum_{i=1}^{m-1}\sum_{j=1}^{n}|a_{0j}a_{ij}-c_{0j}c_{ij}|^2\right)^{\frac{1}{2}} \\ &\quad +\left(\sum_{i=1}^{m-1}\sum_{j=1}^{n}|c_{0j}c_{ij}-b_{0j}b_{ij}|^2\right)^{\frac{1}{2}} \\ &= d_3([a_{ij}],[c_{ij}])+d_3([c_{ij}],[b_{ij}]) \end{aligned}$$
□

Proof [Proposition 15]: For all $[a_{ij}],[b_{ij}],[c_{ij}] \in FPFS_E[U]$,

i. $d_4([a_{ij}],[a_{ij}]) = \sum_{i=1}^{m-1}\left(\sum_{j=1}^{n}|a_{0j}a_{ij}-a_{0j}a_{ij}|^2\right)^{\frac{1}{2}} = \sum_{i=1}^{m-1}\left(\sum_{j=1}^{n}0\right)^{\frac{1}{2}}=0$

ii. $d_4([a_{ij}],[b_{ij}]) = \sum_{i=1}^{m-1}\left(\sum_{j=1}^{n}|a_{0j}a_{ij}-b_{0j}b_{ij}|^2\right)^{\frac{1}{2}} = \sum_{i=1}^{m-1}\left(\sum_{j=1}^{n}|b_{0j}b_{ij}-a_{0j}a_{ij}|^2\right)^{\frac{1}{2}} = d_4([b_{ij}],[a_{ij}])$

iii.
$$\begin{aligned} d_4([a_{ij}],[b_{ij}]) &= \sum_{i=1}^{m-1}\left(\sum_{j=1}^{n}|a_{0j}a_{ij}-b_{0j}b_{ij}|^2\right)^{\frac{1}{2}} \\ &= \sum_{i=1}^{m-1}\left(\sum_{j=1}^{n}|a_{0j}a_{ij}-c_{0j}c_{ij} + c_{0j}c_{ij}-b_{0j}b_{ij}|^2\right)^{\frac{1}{2}} \\ &\leq \sum_{i=1}^{m-1}\left(\sum_{j=1}^{n}\left(|a_{0j}a_{ij}-c_{0j}c_{ij}| + |c_{0j}c_{ij}-b_{0j}b_{ij}|\right)^2\right)^{\frac{1}{2}} \\ &\leq \sum_{i=1}^{m-1}\left[\left(\sum_{j=1}^{n}|a_{0j}a_{ij}-c_{0j}c_{ij}|^2\right)^{\frac{1}{2}} + \left(\sum_{j=1}^{n}|c_{0j}c_{ij}-b_{0j}b_{ij}|^2\right)^{\frac{1}{2}}\right] \\ &= \sum_{i=1}^{m-1}\left(\sum_{j=1}^{n}|a_{0j}a_{ij}-c_{0j}c_{ij}|^2\right)^{\frac{1}{2}} + \sum_{i=1}^{m-1}\left(\sum_{j=1}^{n}|c_{0j}c_{ij}-b_{0j}b_{ij}|^2\right)^{\frac{1}{2}} \\ &= d_4([a_{ij}],[c_{ij}]) + d_4([c_{ij}],[b_{ij}]) \end{aligned}$$

□

Proof [Proposition 16]: For all $[a_{ij}],[b_{ij}],[c_{ij}] \in FPFS_E[U]$,

i. $d_5([a_{ij}],[a_{ij}]) = \sum_{i=1}^{m-1}\max_{j\in I_n}\{|a_{0j}a_{ij}-a_{0j}a_{ij}|\} = \sum_{i=1}^{m-1}\max_{j\in I_n}\{0\} = 0$

ii. $d_5([a_{ij}],[b_{ij}]) = \sum_{i=1}^{m-1}\max_{j\in I_n}\{|a_{0j}a_{ij}-b_{0j}b_{ij}|\} = \sum_{i=1}^{m-1}\max_{j\in I_n}\{|b_{0j}b_{ij}-a_{0j}a_{ij}|\} = d_5([b_{ij}],[a_{ij}])$

iii.
$$\begin{aligned} d_5([a_{ij}],[b_{ij}]) &= \sum_{i=1}^{m-1}\max_{j\in I_n}\{|a_{0j}a_{ij}-b_{0j}b_{ij}|\} \\ &= \sum_{i=1}^{m-1}\max_{j\in I_n}\{|a_{0j}a_{ij}-c_{0j}c_{ij} + c_{0j}c_{ij}-b_{0j}b_{ij}|\} \\ &\leq \sum_{i=1}^{m-1}\max_{j\in I_n}\{|a_{0j}a_{ij}-c_{0j}c_{ij}| + |c_{0j}c_{ij}-b_{0j}b_{ij}|\} \\ &\leq \sum_{i=1}^{m-1}\left[\max_{j\in I_n}\{|a_{0j}a_{ij}-c_{0j}c_{ij}|\} + \max_{j\in I_n}\{|c_{0j}c_{ij}-b_{0j}b_{ij}|\}\right] \\ &= \sum_{i=1}^{m-1}\max_{j\in I_n}\{|a_{0j}a_{ij}-c_{0j}c_{ij}|\} + \sum_{i=1}^{m-1}\max_{j\in I_n}\{|c_{0j}c_{ij}-b_{0j}b_{ij}|\} \\ &= d_5([a_{ij}],[c_{ij}]) + d_5([c_{ij}],[b_{ij}]) \end{aligned}$$

□

Proof [Proposition 17]: For all $[a_{ij}],[b_{ij}],[c_{ij}] \in FPFS_E[U]$ and $p \in \mathbb{N}^+$,

i. $d_6^p([a_{ij}],[a_{ij}]) = \sum_{i=1}^{m-1}\left(\sum_{j=1}^{n}|a_{0j}a_{ij}-a_{0j}a_{ij}|^p\right)^{\frac{1}{p}} = \sum_{i=1}^{m-1}\left(\sum_{j=1}^{n}0\right)^{\frac{1}{p}} = 0$

ii. $d_6^p([a_{ij}],[b_{ij}]) = \sum_{i=1}^{m-1}\left(\sum_{j=1}^{n}|a_{0j}a_{ij}-b_{0j}b_{ij}|^p\right)^{\frac{1}{p}} = \sum_{i=1}^{m-1}\left(\sum_{j=1}^{n}|b_{0j}b_{ij}-a_{0j}a_{ij}|^p\right)^{\frac{1}{p}} = d_6^p([b_{ij}],[a_{ij}])$

iii.
$$\begin{aligned} d_6^p([a_{ij}],[b_{ij}]) &= \left(\sum_{i=1}^{m-1}\sum_{j=1}^{n}|a_{0j}a_{ij}-b_{0j}b_{ij}|^p\right)^{\frac{1}{p}} \\ &= \left(\sum_{i=1}^{m-1}\sum_{j=1}^{n}|a_{0j}a_{ij}-c_{0j}c_{ij} + c_{0j}c_{ij}-b_{0j}b_{ij}|^p\right)^{\frac{1}{p}} \\ &\leq \left(\sum_{i=1}^{m-1}\sum_{j=1}^{n}\left(|a_{0j}a_{ij}-c_{0j}c_{ij}| + |c_{0j}c_{ij}-b_{0j}b_{ij}|\right)^p\right)^{\frac{1}{p}} \\ &\leq \left(\sum_{i=1}^{m-1}\sum_{j=1}^{n}|a_{0j}a_{ij}-c_{0j}c_{ij}|^p\right)^{\frac{1}{p}} + \left(\sum_{i=1}^{m-1}\sum_{j=1}^{n}|c_{0j}c_{ij}-b_{0j}b_{ij}|^p\right)^{\frac{1}{p}} \\ &= d_6^p([a_{ij}],[c_{ij}]) + d_6^p([c_{ij}],[b_{ij}]) \end{aligned}$$

□

Proof [Proposition 18]: For all $[a_{ij}],[b_{ij}],[c_{ij}] \in FPFS_E[U]$ and $p \in \mathbb{N}^+$,

i. $d_7^p([a_{ij}],[a_{ij}]) = \sum_{i=1}^{m-1}\left(\sum_{j=1}^{n}|a_{0j}a_{ij}-a_{0j}a_{ij}|^p\right)^{\frac{1}{p}} = \sum_{i=1}^{m-1}\left(\sum_{j=1}^{n}0\right)^{\frac{1}{p}} = 0$

ii. $d_7^p([a_{ij}],[b_{ij}]) = \sum_{i=1}^{m-1}\left(\sum_{j=1}^{n}|a_{0j}a_{ij}-b_{0j}b_{ij}|^p\right)^{\frac{1}{p}} = \sum_{i=1}^{m-1}\left(\sum_{j=1}^{n}|b_{0j}b_{ij}-a_{0j}a_{ij}|^p\right)^{\frac{1}{p}} = d_7^p([b_{ij}],[a_{ij}])$

iii.
$$\begin{aligned} d_7^p([a_{ij}],[b_{ij}]) &= \sum_{i=1}^{m-1}\left(\sum_{j=1}^{n}|a_{0j}a_{ij}-b_{0j}b_{ij}|^p\right)^{\frac{1}{p}} \\ &= \sum_{i=1}^{m-1}\left(\sum_{j=1}^{n}|a_{0j}a_{ij}-c_{0j}c_{ij} + c_{0j}c_{ij}-b_{0j}b_{ij}|^p\right)^{\frac{1}{p}} \\ &\leq \sum_{i=1}^{m-1}\left(\sum_{j=1}^{n}\left(|a_{0j}a_{ij}-c_{0j}c_{ij}| + |c_{0j}c_{ij}-b_{0j}b_{ij}|\right)^p\right)^{\frac{1}{p}} \\ &\leq \sum_{i=1}^{m-1}\left[\left(\sum_{j=1}^{n}|a_{0j}a_{ij}-c_{0j}c_{ij}|^p\right)^{\frac{1}{p}} + \left(\sum_{j=1}^{n}|c_{0j}c_{ij}-b_{0j}b_{ij}|^p\right)^{\frac{1}{p}}\right] \\ &= \sum_{i=1}^{m-1}\left(\sum_{j=1}^{n}|a_{0j}a_{ij}-c_{0j}c_{ij}|^p\right)^{\frac{1}{p}} + \sum_{i=1}^{m-1}\left(\sum_{j=1}^{n}|c_{0j}c_{ij}-b_{0j}b_{ij}|^p\right)^{\frac{1}{p}} \\ &= d_7^p([a_{ij}],[c_{ij}]) + d_7^p([c_{ij}],[b_{ij}]) \end{aligned}$$

□

Proof [Proposition 19]: For all $[a_{ij}],[b_{ij}],[c_{ij}] \in FPFS_E[U]$ and $p \in \mathbb{N}^+$,

i. $d_8^p([a_{ij}],[a_{ij}]) = \left(\sum_{i=1}^{m-1}\max_{j\in I_n}\{|a_{0j}a_{ij}-a_{0j}a_{ij}|^p\}\right)^{\frac{1}{p}} = \left(\sum_{i=1}^{m-1}\max_{j\in I_n}\{0\}\right)^{\frac{1}{p}} = 0$

ii. $d_8^p([a_{ij}],[b_{ij}]) = \left(\sum_{i=1}^{m-1} \max_{j\in I_n}\{|a_{0j}a_{ij} - b_{0j}b_{ij}|^p\}\right)^{\frac{1}{p}}$

$$= \left(\sum_{i=1}^{m-1} \max_{j\in I_n}\{|b_{0j}b_{ij} - a_{0j}a_{ij}|^p\}\right)^{\frac{1}{p}} = d_8^p([b_{ij}],[a_{ij}])$$

iii. $d_8^p([a_{ij}],[b_{ij}]) = \left(\sum_{i=1}^{m-1} \max_{j\in I_n}\{|a_{0j}a_{ij} - b_{0j}b_{ij}|^p\}\right)^{\frac{1}{p}}$

$$= \left(\sum_{i=1}^{m-1} \max_{j\in I_n}\{|a_{0j}a_{ij} - c_{0j}c_{ij} + c_{0j}c_{ij} - b_{0j}b_{ij}|^p\}\right)^{\frac{1}{p}}$$

$$\leq \left(\sum_{i=1}^{m-1} \max_{j\in I_n}\left\{\left(|a_{0j}a_{ij} - c_{0j}c_{ij}| + |c_{0j}c_{ij} - b_{0j}b_{ij}|\right)^p\right\}\right)^{\frac{1}{p}}$$

$$= \left(\sum_{i=1}^{m-1} \max_{j\in I_n}\{|a_{0j}a_{ij} - c_{0j}c_{ij}| + |c_{0j}c_{ij} - b_{0j}b_{ij}|\}^p\right)^{\frac{1}{p}}$$

$$\leq \left(\sum_{i=1}^{m-1} \left(\max_{j\in I_n}\{|a_{0j}a_{ij} - c_{0j}c_{ij}|\} + \max_{j\in I_n}\{|c_{0j}c_{ij} - b_{0j}b_{ij}|\}\right)^p\right)^{\frac{1}{p}}$$

$$\leq \left(\sum_{i=1}^{m-1} \max_{j\in I_n}\{|a_{0j}a_{ij} - c_{0j}c_{ij}|\}^p + \max_{j\in I_n}\{|c_{0j}c_{ij} - b_{0j}b_{ij}|\}^p\right)^{\frac{1}{p}}$$

$$\leq \left(\sum_{i=1}^{m-1} \max_{j\in I_n}\{|a_{0j}a_{ij} - c_{0j}c_{ij}|\}^p\right)^{\frac{1}{p}} + \left(\sum_{i=1}^{m-1} \max_{j\in I_n}\{|c_{0j}c_{ij} - b_{0j}b_{ij}|\}^p\right)^{\frac{1}{p}}$$

$$= d_8^p([a_{ij}],[c_{ij}]) + d_8^p([c_{ij}],[b_{ij}])$$

□

Proof [Proposition 23]: Let $[a_{ij}], [b_{ij}] \in FPFS_E[U]$.

i. Since $[a_{ij}]\tilde{\subseteq}[b_{ij}]\tilde{\subseteq}[c_{ij}]$, for all $i \in I_m^*$ and $j \in I_n$, $a_{ij} \leq b_{ij} \leq c_{ij}$. Therefore, for all $i \in I_m$ and $j \in I_n$, $a_{0j}a_{ij} \leq b_{0j}b_{ij} \leq c_{0j}c_{ij}$ holds. Then,

$$b_{0j}b_{ij} - a_{0j}a_{ij} \leq c_{0j}c_{ij} - a_{0j}a_{ij} \quad \text{and}$$
$$c_{0j}c_{ij} - b_{0j}b_{ij} \leq c_{0j}c_{ij} - a_{0j}a_{ij}$$

Thus,

$$|b_{0j}b_{ij} - a_{0j}a_{ij}| \leq |c_{0j}c_{ij} - a_{0j}a_{ij}| \quad \text{and}$$
$$|c_{0j}c_{ij} - b_{0j}b_{ij}| \leq |c_{0j}c_{ij} - a_{0j}a_{ij}|$$

Thereafter,

$$\sum_{i=1}^{m-1}\sum_{j=1}^{n} |a_{0j}a_{ij} - b_{0j}b_{ij}| \leq \sum_{i=1}^{m-1}\sum_{j=1}^{n} |a_{0j}a_{ij} - c_{0j}c_{ij}| \quad \text{and}$$
$$\sum_{i=1}^{m-1}\sum_{j=1}^{n} |b_{0j}b_{ij} - c_{0j}c_{ij}| \leq \sum_{i=1}^{m-1}\sum_{j=1}^{n} |a_{0j}a_{ij} - c_{0j}c_{ij}|$$

Consequently,

$$d_1([a_{ij}],[b_{ij}]) \leq d_1([a_{ij}],[c_{ij}]) \quad \text{and}$$
$$d_1([b_{ij}],[c_{ij}]) \leq d_1([a_{ij}],[c_{ij}])$$

Others can be proved by similar way. □

REFERENCES

[1] D. Molodtsov, "Soft set theory-first results," *Comput. Math. Appl.*, vol. 37, nos. 4–5, pp. 19–31, 1999, doi: 10.1016/S0898-1221(99)00056-5.

[2] P. K. Maji, R. Biswas, and A. R. Roy, "Fuzzy soft sets," *J. Fuzzy Math.*, vol. 9, no. 3, pp. 589–602, 2001.

[3] N. Çağman, S. Enginoğlu, and F. Çıtak, "Fuzzy soft set theory and its applications," *Iranian J. Fuzzy Syst.*, vol. 8, no. 3, pp. 137–147, 2011. [Online]. Available: https://ijfs.usb.ac.ir/article_292_22928400ec0d727700fd251a4f63fa07.pdf

[4] N. Çağman, F. Çıtak, and S. Enginoğlu, "FP-soft set theory and its applications," *Ann. Fuzzy Math. Informat.*, vol. 2, no. 2, pp. 219–226, 2011. [Online]. Available: http://www.afmi.or.kr/papers/2011/Vol-02_No-02/AFMI-2-2(219-226)-J-110329R1.pdf

[5] N. Çağman, F. Çıtak, and S. Enginoğlu, "Fuzzy parameterized fuzzy soft set theory and its applications," *Turkish J. Fuzzy Syst.*, vol. 1, no. 1, pp. 21–35, 2010. [Online]. Available: https://pdfs.semanticscholar.org/55ce/c8656188fa125d335deb8d358239d2fcdaa9.pdf

[6] N. Çağman and S. Enginoğlu, "Soft matrix theory and its decision making," *Comput. Math. Appl.*, vol. 59, no. 10, pp. 3308–3314, May 2010, doi: 10.1016/j.camwa.2010.03.015.

[7] N. Cagman and S. Enginoglu, "Fuzzy soft matrix theory and its application in decision making," *Iranian J. Fuzzy Syst.*, vol. 9, no. 1, pp. 109–119, 2012. [Online]. Available: https://ijfs.usb.ac.ir/article_229_b6d292816ccde2e91d91920714cb6245.pdf

[8] S. Enginoğlu and N. Çağman, "Fuzzy parameterized fuzzy soft matrices and their application in decision-making," *TWMS J. Appl. Eng. Math.*, vol. 10, no. 4, pp. 1105–1115, 2020. [Online]. Available: https://jaem.isikun.edu.tr/web/images/articles/vol.10.no.4/25.pdf

[9] S. Enginoğlu and S. Memiş, "A configuration of some soft decision-making algorithms via fpfs-matrices," *Cumhuriyet Sci. J.*, vol. 39, no. 4, pp. 871–881, 2018, doi: 10.17776/csj.409915.

[10] S. Enginoğlu, S. Memiş, and T. Öngel, "Comment on soft set theory and uni-int decision-making [European Journal of Operational Research, (2010) 207, 848-855]," *J. New Results Sci.*, vol. 7, no. 3, pp. 28–43, 2018. [Online]. Available: https://dergipark.org.tr/download/article-file/596954

[11] S. Enginoğlu, S. Memiş, and B. Arslan, "Comment (2) on soft set theory and uni-int decision-making [European journal of operational research, (2010) 207, 848-855]," *J. New Theory*, vol. 25, no. 3, pp. 84–102, 2018. [Online]. Available: https://dergipark.org.tr/download/article-file/594503

[12] T. Aydın and S. Enginoğlu, "A configuration of five of the soft decision-making methods via fuzzy parameterized fuzzy soft matrices and their application to a performance-based value assignment problem," in *Proc. Int. Conf. Sci. Technol. Natural Sci. Technol. (ICONST-NST)*, M. Kılıç, K. Özkan, M. Karaboyacı, K. Taşdelen, H. Kandemir, and A. Beram, Eds. Prizren, Kosovo: Association of Kutbilge Academicians, 2019, pp. 56–67.

[13] S. Enginoğlu and T. Öngel, "Configurations of several soft decision-making methods to operate in fuzzy parameterized fuzzy soft matrices space," *Eskehir Tech. Univ. J. Sci. Technol. A-Appl. Sci. Eng.*, vol. 21, no. 1, pp. 58–71, Mar. 2020, doi: 10.18038/estubtda.562578.

[14] T. Aydın and S. Enginoğlu, "Configurations of SDM methods proposed between 1999 and 2012: A follow-up study," in *Proc. 4th Int. Conf. Math. Istanbul Meeting World Mathematicians*, K. Yıldırım, Ed. Istanbul, Turkey: Fatih Sultan Mehmet Vakıf University, 2020, pp. 192–211. [Online]. Available: https://www.researchgate.net/publication/346569120

[15] S. Enginoğlu and S. Memiş, "Comment on fuzzy soft sets [the journal of fuzzy mathematics 9 (3), 2001, 589–602]," *Int. J. Latest Eng. Res. Appl.*, vol. 3, no. 9, pp. 1–9, 2018. [Online]. Available: https://www.ijlera.com/papers/v3-i9/1.201809134.pdf

[16] S. Enginoğlu, S. Memiş, and T. Öngel, "A fast and simple soft decision-making algorithm: EMO18o," in *Proc. Int. Conf. Math. Stud. Appl.*, M. Akgül, İ. Yılmaz, and A. İpek, Eds. Karaman, Turkey: Karamanoğlu Mehmetbey University, 2018, pp. 179–186.

[17] S. Enginoğlu, S. Memiş, and B. Arslan, "A fast and simple soft decision-making algorithm: EMA18an," in *Proc. Int. Conf. Math. Stud. Appl.*, M. Akgül, İ. Yılmaz, and A. İpek, Eds. Karaman, Turkey: Karamanoğlu Mehmetbey University, 2018, pp. 428–436.

[18] S. Enginoğlu and S. Memiş, "A review on an application of fuzzy soft set in multicriteria decision making problem [P. K. Das, R. Borgohain, International Journal of Computer Applications 38 (2012) 33-37]," in *Proc. Int. Conf. Mathematical Stud. Appl.*, M. Akgül, İ. Yılmaz, and A. İpek, Eds., Karaman, Turkey, 2018, pp. 173–178.
[19] S. Enginoğlu and S. Memiş, "A review on some soft decision-making methods," in *Proc. Int. Conf. Mathematical Stud. Appl.*, M. Akgül, İ. Yılmaz, and A. İpek, Eds. Karaman, Turkey: Karamanoğlu Mehmetbey University, 2018, pp. 437–442.
[20] S. Enginoğlu, S. Memiş, and F. Karaaslan, "A new approach to group decision-making method based on topsis under fuzzy soft environment," *J. New Results Sci.*, vol. 8, no. 2, pp. 42–52, 2019. [Online]. Available: https://dergipark.org.tr/tr/download/article-file/904374
[21] S. Enginoğlu, S. Memiş, and N. Çağman, "A generalization of fuzzy soft max-min decision-making method and its application to a performance-based value assignment in image denoising," *El-Cezerî J. Sci. Eng.*, vol. 6, no. 3, pp. 466–481, Sep. 2019, doi: 10.31202/ecjse.551487.
[22] S. Enginoğlu and S. Memiş, "A generalization of fuzzy soft max-min decision-making method and its application to a performance-based value assignment in image denoising," *J. New Results Sci.*, vol. 9, no. 1, pp. 19–36, Sep. 2019. [Online]. Available: http://dergipark.org.tr/tr/pub/jnrs/issue/53974/709375
[23] T. Mahmood, Z. U. Rehman, and A. Sezgin, "Lattice ordered soft near rings," *Korean J. Math.*, vol. 26, no. 3, pp. 503–517, 2018, doi: 10.11568/kjm.2018.26.3.503.
[24] A. Sezgin, N. Çağman, and F. Çıtak, "α-inclusions applied to group theory via soft set and logic," *Commun. Fac. Sci. Univ. Ankara Ser. A1 Math. Statist.*, vol. 68, no. 1, pp. 334–352, 2019. [Online]. Available: https://dergipark.org.tr/download/article-file/467069
[25] C. S. Özlü and A. Sezgin, "Soft covered ideals in semigroups," *Acta Universitatis Sapientiae, Mathematica*, vol. 12, no. 2, pp. 317–346, 2020, doi: 10.2478/ausm-2020-0023.
[26] G. Şenel, J. G. Lee, and K. Hur, "Advanced soft relation and soft mapping," *Int. J. Comput. Intell. Syst.*, vol. 14, no. 1, pp. 461–470, 2021, doi: 10.2991/ijcis.d.201221.001.
[27] S. Enginoğlu, N. Çağman, S. Karataş, and T. Aydın, "On soft topology," *El-Cezerî J. Sci. Eng.*, vol. 2, no. 3, pp. 23–38, 2015, doi: 10.31202/ecjse.67135.
[28] I. Zorlutuna and S. Atmaca, "Fuzzy parametrized fuzzy soft topology," *New Trends Math. Sci.*, vol. 4, no. 1, pp. 142–152, 2016, doi: 10.20852/ntmsci.2016115658.
[29] M. Riaz and M. R. Hashmi, "Fuzzy parameterized fuzzy soft topology with applications," *Ann. FUZZY Math. Informat.*, vol. 13, no. 5, pp. 593–613, May 2017. [Online]. Available: https://www.kci.go.kr/kciportal/landing/article.kci?arti_id=ART002458073
[30] M. Riaz and M. R. Hashmi, "Fuzzy parameterized fuzzy soft compact spaces with decision-making," *Punjab Univ. J. Math.*, vol. 50, no. 2, pp. 131–145, 2018. [Online]. Available: https://pu.edu.pk/images/journal/maths/PDF/Paper-10_50_2_2018.pdf
[31] T. Aydın and S. Enginoğlu, "Some results on soft topological notions," *J. New Results Sci.*, vol. 10, no. 1, pp. 65–75, 2021. [Online]. Available: https://dergipark.org.tr/tr/pub/jnrs/issue/62194/910337
[32] N. Çağman and S. Enginoğlu, "Soft set theory and uni-int decision making," *Eur. J. Oper. Res.*, vol. 207, pp. 848–855, Dec. 2010, doi: 10.1016/j.ejor.2010.05.004.
[33] F. Karaaslan and I. Deli, "Soft neutrosophic classical sets and their applications in decision-making," *Palestine J. Math.*, vol. 9, no. 1, pp. 312–326, 2020.
[34] H. Garg and R. Arora, "Algorithms based on COPRAS and aggregation operators with new information measures for possibility intuitionistic fuzzy soft decision-making," *Math. Problems Eng.*, vol. 2020, Art. no. 1563768, Apr. 2020, doi: 10.1155/2020/1563768.
[35] M. Riaz, F. Çıtak, W. Nabeela, and A. Mushtaq, "Roughness and fuzziness associated with soft multi-sets and their application to madm," *J. New Theory*, vol. 31, pp. 1–19, Jun. 2020. [Online]. Available: https://dergipark.org.tr/en/download/article-file/1177455
[36] H. Garg, "Multi-attribute group decision-making process based on possibility degree and operators for intuitionistic multiplicative set," *Complex Intell. Syst.*, vol. 7, no. 2, pp. 1099–1121, 2021, doi: 10.1007/s40747-020-00256-y.
[37] P. Majumdar and S. K. Samanta, "Similarity measure of soft sets," *New Math. Natural Comput.*, vol. 4, no. 1, pp. 1–12, 2008, doi: 10.1142/S1793005708000908.
[38] A. Kharal, "Distance and similarity measures for soft sets," *New Math. Natural Comput.*, vol. 6, no. 3, pp. 321–334, Nov. 2010.
[39] P. Majumdar and S. K. Samanta, "On similarity measures of fuzzy soft sets," *Int. J. Adv. Soft Comput. Appl.*, vol. 3, no. 2, pp. 1–8, 2011. [Online]. Available: https://www.i-csrs.org/Volumes/ijasca/vol.3/vol.3.2.4.Jul. 11.pdf
[40] M. Riaz, K. Naeem, and D. Afzal, "A similarity measure under pythagorean fuzzy soft environment with applications," *Comput. Appl. Math.*, vol. 39, no. 4, pp. 1–17, Dec. 2020, doi: 10.1007/s40314-020-01321-5.
[41] Q. Feng and W. Zheng, "New similarity measures of fuzzy soft sets based on distance measures," *Ann. Fuzzy Math. Informat.*, vol. 7, no. 4, pp. 669–686, 2014. [Online]. Available: https://pdfs.semanticscholar.org/9921/20d1eb3dfc1286259a5bee2ff260e0e377cb.pdf
[42] F. Xiao, "A hybrid fuzzy soft sets decision making method in medical diagnosis," *IEEE Access*, vol. 6, pp. 25300–25312, 2018, doi: 10.1109/ACCESS.2018.2820099.
[43] M. M. Mushrif, S. Senqupta, and A. K. Ray, "Texture classification using a novel, soft-set theory based classification algorithm," in *Proc. 7th Asian Conf. Comput. Vis.*, P. J. Narayanan, S. K. Nayar, and H.-Y. Shum, Eds. Hyderabad, India: Springer, 2006, pp. 246–254. [Online]. Available: https://link.springer.com/chapter/10.1007/11612032_26
[44] B. Handaga, T. Herawan, and M. M. Deris, "FSSC: An algorithm for classifying numerical data using fuzzy soft set theory," *Int. J. Fuzzy Syst. Appl.*, vol. 2, no. 4, pp. 29–46, Oct. 2012, doi: 10.4018/ijfsa.2012100102.
[45] S. A. Lashari, R. Ibrahim, and N. Senan, "Medical data classification using similarity measure of fuzzy soft set based distance measure," *J. Telecommun., Electron. Comput. Eng.*, vol. 9, nos. 2–9, pp. 95–99, 2017. [Online]. Available: https://jtec.utem.edu.my/jtec/article/view/2681
[46] I. T. R. Yanto, R. R. Seadudin, S. A. Lashari, and Haviluddin, "A numerical classification technique based on fuzzy soft set using Hamming distance," in *Proc. 3rd Int. Conf. Soft Comput. Data Mining*, R. Ghazali, M. M. Deris, N. M. Nawi, and J. H. Abawajy, Eds., Johor, Malaysia, 2018, pp. 252–260. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-319-72550-5_25
[47] J. M. Keller, M. R. Gray, and J. A. Givens, "A fuzzy K-nearest neighbor algorithm," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-15, no. 4, pp. 580–585, Jul. 1985, doi: 10.1109/TSMC.1985.6313426.
[48] M. M. Kumbure, P. Luukka, and M. Collan, "A new fuzzy k-nearest neighbor classifier based on the Bonferroni mean," *Pattern Recognit. Lett.*, vol. 140, pp. 172–178, Dec. 2020, doi: 10.1016/j.patrec.2020.10.005.
[49] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, 1995. [Online]. Available: https://link.springer.com/article/10.1007/BF00994018
[50] D. Dua and C. Graff. (2019). *UCI Machine Learning Repository*. [Online]. Available: https://archive.ics.uci.edu/ml
[51] S. Memiş, S. Enginoğlu, and U. Erkan, "A data classification method in machine learning based on normalised Hamming pseudo-similarity of fuzzy parameterized fuzzy soft matrices," *Bilge Int. J. Sci. Technol. Res.*, vol. 3, pp. 1–8, Dec. 2019, doi: 10.30516/bilgesci.643821.
[52] S. Memiş and S. Enginoğlu, "An application of fuzzy parameterized fuzzy soft matrices in data classification," in *Proc. Int. Conf. Sci. Technol.; Natural Sci. Technol. (ICONST-NST)*, M. Kılıç, K. Özkan, M. Karaboyacı, K. Taşdelen, H. Kandemir, and A. Beram, Eds. Prizren, Kosovo: Association of Kutbilge Academicians, 2019, pp. 68–77. [Online]. Available: https://www.researchgate.net/publication/335524651
[53] M. Friedman, "A comparison of alternative tests of significance for the problem of m rankings," *Ann. Math. Statist.*, vol. 11, no. 1, pp. 86–92, 1940. [Online]. Available: https://www.jstor.org/stable/2235971
[54] P. B. Nemenyi. (1963). *Distribution-Free Multiple Comparisons*. [Online]. Available: https://books.google.com.tr/books?id=nhDMtgAACAAJ
[55] M. Stone, "Cross-validatory choice and assessment of statistical predictions," *J. Roy. Stat. Society. Ser. (Methodological)*, vol. 36, no. 2, pp. 111–147, 1974, doi: 10.1111/j.2517-6161.1974.tb00994.x.
[56] U. Erkan, "A precise and stable machine learning algorithm: Eigenvalue classification (EigenClass)," *Neural Comput. Appl.*, vol. 33, no. 10, pp. 5381–5392, May 2021, doi: 10.1007/s00521-020-05343-2.
[57] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *J. Mach. Learn. Res.*, vol. 7, pp. 1–30, Dec. 2006. [Online]. Available: https://www.jmlr.org/papers/volume7/demsar06a/demsar06a.pdf
[58] S. Enginoğlu and B. Arslan, "Intuitionistic fuzzy parameterized intuitionistic fuzzy soft matrices and their application in decision-making," *Comput. Appl. Math.*, vol. 39, no. 4, 2020, Art. no. 325. [Online]. Available: https://link.springer.com/article/10.1007%2Fs40314-020-01325-1

[59] S. Enginoğlu, T. Aydın, S. Memiş, and B. Arslan, "Operability-oriented configurations of the soft decision-making methods proposed between 2013 and 2016 and their comparisons," *J. New Theory*, vol. 34, pp. 82–114, 2021. [Online]. Available: https://dergipark.org.tr/tr/pub/jnt/issue/61070/896315

SAMET MEMİŞ was born in Turkey, in 1991. He received the B.Sc., M.Sc., and Ph.D. degrees from the Department of Mathematics, Çanakkale Onsekiz Mart University, in 2014, 2016, and 2021, respectively.

His current research interests are fuzzy sets, soft sets, soft matrices, *fpfs*-matrices, similarity measure, distance measure, decision-making, image denoising, and machine learning.

SERDAR ENGİNOĞLU received the bachelor's degree in mathematics from Atatürk University, Erzurum, Turkey, in 1998, and the master's and Ph.D. degrees in mathematics from Gaziosmanpaşa University, Tokat, Turkey, in 2009 and 2012, respectively.

He is currently an Associate Professor and the Head with the Department of Fundamentals of Mathematics and Mathematical Logic, Çanakkale Onsekiz Mart University, Çanakkale, Turkey. His current research interests include fuzzy sets, soft sets, soft matrices, soft decision-making, image processing, and machine learning.

UĞUR ERKAN received the B.Sc. degree in computer engineering from Selçuk University, Turkey, in 2001, the M.Sc. degree in mathematics and the Ph.D. degree in mechatronics engineering from Gaziosmanpaşa University, Turkey, in 2012 and 2017, respectively.

He is currently an Assistant Professor with the Department of Computer Engineering, Karamanoğlu Mehmetbey University, Karaman, Turkey. His research interests include image processing, image denoising, image restoration, image encryption, and machine learning.