Alireza Khastan; Elham Eskandari

Fuzzy clustering of fuzzy data considering the shape of the membership functions using a novel representation learning technique

Kybernetika, Vol. 61 (2025), No. 2, 168–184

Persistent URL: http://dml.cz/dmlcz/152986

KYBERNETIKA — VOLUME *61* (2025), NUMBER 2, PAGES 168–184

FUZZY CLUSTERING OF FUZZY DATA CONSIDERING THE SHAPE OF THE MEMBERSHIP FUNCTIONS USING A NOVEL REPRESENTATION LEARNING TECHNIQUE

Alireza Khastan and Elham Eskandari

Most existing distance measures for fuzzy data do not capture differences in the shapes of the left and right tails of membership functions. As a result, they may calculate a distance of zero between fuzzy data even when these differences exist. Additionally, some distance measures cannot compute distances between fuzzy data when their membership functions differ in type.

In this paper, inspired by human visual perception, we propose a fuzzy clustering method for fuzzy data using a novel representation technique that is capable of detecting small differences in the shapes of the left and right tails of membership functions. Moreover, it effectively clusters fuzzy data even when their membership functions differ in type. By utilizing the pre-trained ResNet50 network as a feature extractor and applying the FCM clustering method to the output from the last convolutional layer, our approach achieves high accuracy in clustering both synthetic and real data sets. Experimental results demonstrate that our method achieved a Rand Index of 0.9965, outperforming state-of-the-art methods, making it particularly suitable for applications that require high clustering accuracy.

Keywords: fuzzy data, resnet, representation learning, fuzzy clustering, convolutional neural networks

Classification: 62H30, 03E72, 68T10

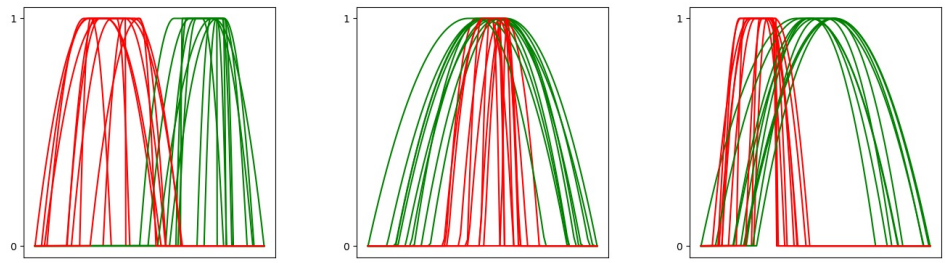
1. INTRODUCTION

Clustering is one of the fundamental tasks in Machine Learning [4]. It is the task of grouping data into clusters of similar samples, and fuzzy clustering generalizes this concept by allowing a sample to belong to multiple clusters according to a certain degree, known as the membership degree, which ranges in the unit interval [1]. Fuzzy clustering has many applications, including market segmentation, forecasting, and image segmentation [8, 18, 20].

Fuzzy numbers can model epistemic uncertainty and its propagation through calculations [16, 22, 24, 25]. The complex structure that characterizes fuzzy data implies three possible clustering scenarios: Fuzzy data can be clustered based on their centers. Here, dissimilarity arises from differences in the location (see Figure 1a). Another type of dissimilarity is related to the size, where fuzzy data are clustered with respect to their

DOI: 10.14736/kyb-2025-2-0168

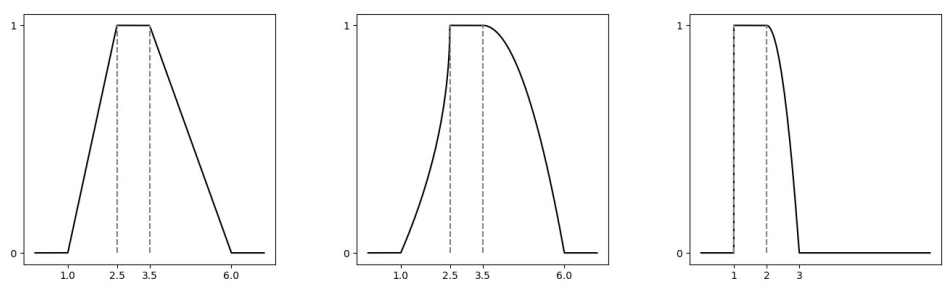
spreads (see Figure 1b). Finally, fuzzy data can be clustered based on both location and size (see Figure 1c).



(a) (b) (c)

Fig. 1: Fuzzy data which are clustered w.r.t. their (a) centers, (b) spreads, and (c) centers and spreads.

Figure 2 shows three samples of one-dimensional fuzzy numbers. Without needing additional information, a visual inspection can show that samples 2a and 2b are more alike compared to sample 2c. Regardless of whether a representation technique or a distance measure is used, many clustering methods in the literature are sufficiently robust to detect significant shape differences.



(a) (b) (c)

Fig. 2: Three samples of one-dimensional fuzzy numbers.

In *fuzzy data* clustering, it is essential to apply methods that are capable of detecting small shape differences. In this paper, for clustering one-dimensional fuzzy data, we imitate the concepts of seeing, describing, and understanding. Inspired by human visual perception, we first convert one-dimensional fuzzy data into two-dimensional image matrices by plotting them (seeing). Next, we derive feature tensors from pre-trained CNNs in an unsupervised manner. Each feature tensor consists of several feature maps, with each feature map detecting specific visual components of the fuzzy datum image

(describing). Finally, we flatten the feature tensors to obtain high-dimensional representations of the fuzzy data (understanding).

The structure of the rest of the paper is as follows. In Section 2, we briefly summarize some preliminary concepts about fuzzy data. Section 3 reviews the related works on fuzzy clustering of fuzzy data. Section 4 describes the proposed method. Section 5 contains a set of numerical experiments. Finally, Section 6 offers concluding remarks.

2. FUZZY DATA

Fuzzy numbers are capable of modeling epistemic uncertainty and its propagation through calculations.

Definition 2.1. (Bede [2]) Let $L, R : [0,1] \to [0,1]$ be two continuous, decreasing functions fulfilling $L(0) = R(0) = 1$, $L(1) = R(1) = 0$. The fuzzy set $x_i : \mathbb{R} \to [0,1], i = 1, \ldots, n$ is a one-dimensional L-R fuzzy number if

$$x_i(u) = \begin{cases} 0, & u \le c_{1i} - l_i, \\ L\left(\dfrac{c_{1i} - u}{l_i}\right), & c_{1i} - l_i \le u \le c_{1i}, \\ 1, & c_{1i} \le u \le c_{2i}, \\ R\left(\dfrac{u - c_{2i}}{r_i}\right), & c_{2i} \le u \le c_{2i} + r_i, \\ 0, & u \ge c_{2i} + r_i. \end{cases}$$

We write $x_i = (c_{1i}, c_{2i}, l_i, r_i)_{L,R}$, where $L(x)$ and $R(x)$ define the tail behavior of fuzzy number, c_{1i} and c_{2i} ($c_{1i} \le c_{2i}$) are called the left and the right center, respectively, l_i and r_i are called the left and the right spread, respectively, and $[c_{1i}, c_{2i}]$ is the core of x_i. There are essentially two types of fuzzy numbers: Singleton–core and interval–core [17]. Singleton-core fuzzy numbers have a single crisp point as their core, while interval-core fuzzy numbers have a continuous range (interval) as their core.

In a particular case, we obtain trapezoidal fuzzy numbers (see Figure 3a) when $L(x) = R(x) = 1 - x$. A trapezoidal fuzzy number x_i can be represented by the quadruple $(c_{1i}, c_{2i}, l_i, r_i) \in \mathbb{R}^4$. If we have $c_{1i} = c_{2i}$, the fuzzy number is called a triangular fuzzy number. If we have $l_i = r_i$, the fuzzy number is called a symmetric fuzzy number. We obtain parabolic fuzzy numbers (see Figure 3b) when $L(x) = R(x) = 1 - x^2$, square root fuzzy numbers (see Figure 3c) when $L(x) = R(x) = 1 - \sqrt{x}$, and Gaussian fuzzy numbers (see Figure 3d) when $L(x) = R(x) = \exp(-x^2)$.

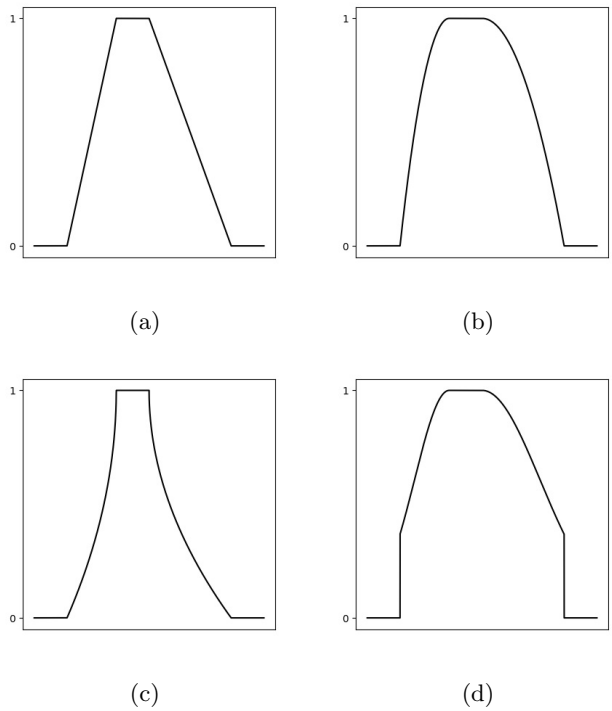


Fig. 3: The (a) trapezoidal, (b) parabolic, (c) square root, and (d) Gaussian fuzzy numbers.

3. RELATED WORKS

In the literature on fuzzy clustering, the Fuzzy C-Means (FCM) clustering model extended by Bezdek in [3] is the most widely applied model [15]. It is formalized as follows

$$\min : \sum_{i=1}^{n}\sum_{g=1}^{k}(u_{ig})^m d_{\mathrm{E}}^2(\mathbf{x}_i, \mathbf{h}_g),$$
$$\sum_{g=1}^{k} u_{ig} = 1, u_{ig} \geq 0, \tag{1}$$

where u_{ig} is the membership degree of the ith sample in the gth cluster, $m \in [1, \infty)$ denotes the fuzziness parameter, and $d_{\mathrm{E}}^2(\mathbf{x}_i, \mathbf{h}_g) = ||\mathbf{x}_i - \mathbf{h}_g||^2$ is the (squared) Euclidean distance measure between the ith sample and the centroid which characterizes the gth cluster.

In the literature, several fuzzy clustering models have been developed to handle fuzzy data. The majority of these proposals are based on the FCM clustering model and primarily focus on introducing suitable distance measures for fuzzy data.

However, most existing distance measures for fuzzy data do not detect differences in the shapes of the left (L function) and right (R function) tails of the membership functions (MFs). As a result, they may calculate a distance of zero between fuzzy data even when these differences exist. Moreover, some distance measures cannot compute distances between fuzzy data when their membership functions differ in type.

Yang and Ko [26] proposed the following distance measure for each pair of one-dimensional singleton-core L-R fuzzy numbers, denoted as x_i and $x_{i'}$:

$$d^2_{\mathrm{YK}}(x_i, x_{i'}) = (c_i - c_{i'})^2 + ((c_i - \lambda l_i) - (c_{i'} - \lambda l_{i'}))^2 + ((c_i + \rho r_i) - (c_{i'} + \rho r_{i'}))^2, \quad (2)$$

where $\lambda = \int_0^1 L^{-1}(\omega)d\omega$ and $\rho = \int_0^1 R^{-1}(\omega)d\omega$ are parameters that summarize the shapes of the left and right tails of MFs. For each value of λ and ρ, a particular MF is determined. For example, $\lambda = \rho = \frac{1}{2}$ for trapezoidal (see Figure 3a), $\lambda = \rho = \frac{2}{3}$ for parabolic (see Figure 3b), $\lambda = \rho = \frac{1}{3}$ for square root (see Figure 3c), and $\lambda = \rho = \frac{\sqrt{\pi}}{2}$ for Gaussian (see Figure 3d).

Example 3.1. The distance between the samples shown in Figures 2a and 2b cannot be computed using the distance measure (2) because their MFs differ in type—specifically, the membership function in Figure 2a is trapezoidal ($\lambda = \rho = \frac{1}{2}$), while that in Figure 2b is square root-parabolic ($\lambda = \frac{1}{3}$ and $\rho = \frac{2}{3}$).

Hung and Yang [21] modified the distance measure (2) and suggested the following new robust distance measure for each pair of one-dimensional singleton-core L-R fuzzy numbers, denoted as x_i and $x_{i'}$:

$$d^2_{\mathrm{HY}}(x_i, x_{i'}) = 1 - \exp\left[-\frac{1}{3}bd^2_{\mathrm{YK}}(x_i, x_{i'})\right], \quad (3)$$

where b is a positive constant.

D'Urso and Giordani [10] proposed the following weighted distance measure for each pair of multi-dimensional symmetric singleton-core L-R fuzzy numbers, denoted as $\mathbf{x}_i$ and $\mathbf{x}_{i'}$:

$$d^2_{\mathrm{DGior}}(\mathbf{x}_i, \mathbf{x}_{i'}) = w_c^2||\mathbf{c}_i - \mathbf{c}_{i'}||^2 + w_s^2||\mathbf{s}_i - \mathbf{s}_{i'}||^2, \quad (4)$$

where w_c and w_s are weights for the center and spread, respectively. Additionally, the following assumptions are provided: $w_c + w_s = 1$ and $w_c \geq w_s \geq 0$. Eskandari et al. [12] demonstrated that the usability of the weights is confined to cases where dissimilarity between fuzzy data arises from differences in location (see Figure 1a).

Coppi et al. [6] proposed the following weighted distance measure for each pair of multi-dimensional L-R fuzzy numbers, denoted as $\mathbf{x}_i$ and $\mathbf{x}_{i'}$:

$$d^2_{\mathrm{CDG}}(\mathbf{x}_i, \mathbf{x}_{i'}) = w_c^2\left[||\mathbf{c}_{1i} - \mathbf{c}_{1i'}||^2 + ||\mathbf{c}_{2i} - \mathbf{c}_{2i'}||^2\right] + w_s^2\left[||\mathbf{l}_i - \mathbf{l}_{i'}||^2 + ||\mathbf{r}_i - \mathbf{r}_{i'}||^2\right], \quad (5)$$

where w_c and w_s are weights for the center and spread, respectively. Additionally, the following assumptions are provided: $w_c + w_s = 1$ and $w_c \geq w_s \geq 0$. This distance measure may suffer from the risk of obtaining coincident clusters—clusters characterized by identical centroids. Moreover, it may calculate a distance of zero between fuzzy data even when their MFs differ in the shapes of the left and right tails.

Example 3.2. The distance measure (5) calculates a distance of zero between the samples shown in Figsures 2a and 2b, even though their MFs differ in the shapes of the left and right tails:

$$w_c^2 \left[||2.5 - 2.5||^2 + ||3.5 - 3.5||^2\right] + w_s^2 \left[||1.5 - 1.5||^2 + ||2.5 - 2.5||^2\right] = 0. \tag{6}$$

D'Urso and De Giovanni [9] by considering the distance measure (5), introduced the following robust distance measure for each pair of multi-dimensional L-R fuzzy numbers, denoted as $\mathbf{x}_i$ and $\mathbf{x}_{i'}$:

$$\begin{aligned} d^2_{\mathrm{DGiov}}(\mathbf{x}_i, \mathbf{x}_{i'}) &= 1 - \exp\{-\beta d^2_{CDG}(\mathbf{x}_i, \mathbf{x}_{i'})\} \\ &= 1 - \exp\{-\beta[(1-\nu)^2[||\mathbf{c}_{1i} - \mathbf{c}_{1i'}||^2 + ||\mathbf{c}_{2i} - \mathbf{c}_{2i'}||^2] \\ &+ \nu^2[||\mathbf{l}_i - \mathbf{l}_{i'}||^2 + ||\mathbf{r}_i - \mathbf{r}_{i'}||^2]]\}, \end{aligned} \tag{7}$$

where $\nu \leq 0.5$ is the weight for the spread and β is a positive constant determined based on the variability of the data. The clustering outcome can still be disrupted by a single outlier datum.

D'Urso and Leski [11] extended the distance measure (2) and introduced the following robust distance measure for each pair of multi-dimensional L-R fuzzy numbers, denoted as $\mathbf{x}_i$ and $\mathbf{x}_{i'}$:

$$\begin{aligned} \mathcal{D}(\mathbf{x}_i, \mathbf{h}_g) =& \mathcal{L}(\mathbf{c}_{1i} - \mathbf{c}_{1g}) + \mathcal{L}(\mathbf{c}_{2i} - c_{2g}) + \\ & \mathcal{L}[(\mathbf{c}_{1i} - \lambda\mathbf{l}_i) - (\mathbf{c}_{1g} - \lambda\mathbf{l}_g)] + \\ & \mathcal{L}[(\mathbf{c}_{2i} + \rho\mathbf{r}_i) - (\mathbf{c}_{2g} + \rho\mathbf{r}_g)], \end{aligned} \tag{8}$$

where $\mathcal{L}$ is the Squared (SQR), Linear (LIN), Huber (HUB) with parameter $\delta > 0$, Sigmoidal (SIG) with parameters $\alpha, \beta > 0$, or Logarithmic (LOG) loss function:

- $\mathcal{L}_{\mathrm{SQR}}(e) = e^2$
- $\mathcal{L}_{\mathrm{LIN}}(e) = |e|$
- $\mathcal{L}_{\mathrm{HUB}}(e) = \begin{cases} \dfrac{e^2}{\delta^2}, & |e| \leq \delta, \\ \dfrac{|e|}{\delta}, & |e| > \delta, \end{cases}$
- $\mathcal{L}_{\mathrm{SIG}}(e) = \dfrac{1}{1 + \exp(-\alpha(|e| - \beta))}$
- $\mathcal{L}_{\mathrm{LOG}}(e) = \log(1 + e^2)$.

Refer to Figure 4 for plots of the loss functions. For a vector argument $\mathbf{e} = [e_1, e_2, \ldots, e_m]$, the loss function takes the form $\mathcal{L}(\mathbf{e}) = \sum_{j=1}^{m} \mathcal{L}(e_j)$. This distance measure has a relatively high computational complexity. If $\mathcal{L}$ is the SQR loss function, then the distance measure proposed by D'Urso and Leski (8) is the same as that proposed by Yang and Ko (2).

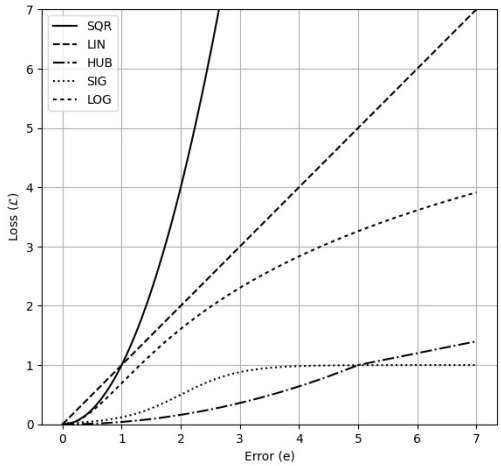


Fig. 4: The graphs of SQR, LIN, HUB with $\delta > 0$, SIG with $\alpha, \beta > 0$, and LOG loss functions [13].

Example 3.3. The distance between the samples shown in Figures 2a and 2b cannot be computed using the distance measure (8) because their MFs differ in type.

Eskandari and Khastan [12] proposed the following distance measure between each pair of multi-dimensional L-R fuzzy numbers, denoted as $\mathbf{x}_i$ and $\mathbf{x}_{i'}$:

$$d^2_{\mathrm{EKh}}(\mathbf{x}_i, \mathbf{x}_{i'}) = \sum_{j=1}^{m} \lambda_{gj}\{(1-\nu)^2[|c_{1ij}-c_{1i'j}|^2+|c_{2ij}-c_{2i'j}|^2]+\nu^2[|l_{ij}-l_{i'j}|^2+|r_{ij}-r_{i'j}|^2]\}, \tag{9}$$

where $\nu = 0.5$ and λ_{gj} is a suitable parameter measuring the importance of the jth feature with respect to the gth cluster. Additionally, the following assumptions are provided: $\lambda_{gj} > 0$ and $\sum_{j=1}^{m} \lambda_{gj} = 1, g = 1, 2, \ldots, k$.

4. PROPOSED METHOD

In this paper, we propose a fuzzy clustering method for fuzzy data using a novel representation technique called ResFClustFD, which is capable of detecting small differences in the shapes of the left and right tails of MFs. Moreover, it effectively clusters fuzzy data even when their MFs differ in type, while none of the existing methods are capable of doing so.

We begin by plotting one-dimensional fuzzy data. Each plot is sized at 224×224, with a white background and black foreground, for all fuzzy data in a data set. Higher resolutions may capture more details but could increase computational costs. The ranges for the x-axis (width) and y-axis (height) are set based on the min and max values from the complete set of n fuzzy data. These gray-scale images of fuzzy data are now treated as two-dimensional matrices.

In this work, we utilize the ResNet50 convolutional neural network [19], pre-trained on the ImageNet data set [7], which contains millions of labeled images, as a feature extractor. Its pre-trained weights allow for feature extraction without the need for domain-specific training data. The objective is to learn a robust feature representation from the fuzzy data. This differs from the purpose of the pre-trained ResNet network, which involves mapping images to thousand classes. Thus, we discard the avgpool and fc layers of the network and only keep the network up to the last convolutional layer. This layer captures the information about the shape of the fuzzy numbers.

Since the network's depth is 2,048, this results in 2,048 feature maps for each fuzzy number image. Finally, we apply the FCM clustering method to the output from the last convolutional layer. Figure 5 shows the overview of the proposed method.

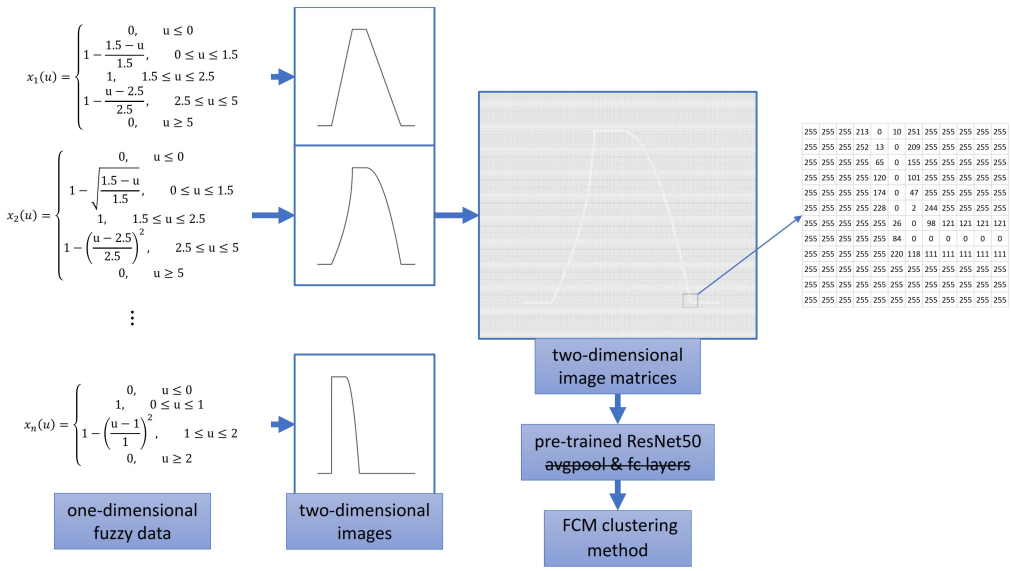


Fig. 5: Overview of the proposed method.

A step-by-step explanation of the ResFClustFD method is as below:

1. Conversion of Fuzzy Data: One-dimensional fuzzy data are transformed into two-dimensional image matrices by plotting their membership functions.

2. Feature Extraction: The pre-trained ResNet50 network is employed for feature extraction.

3. Clustering: The extracted features are input to the Fuzzy C-Means (FCM) algorithm.

To sum up, the key contributions of our work are the conversion of fuzzy data into two-dimensional image matrices and the use of pre-trained ResNet50 for feature extraction.

Scheme	MF	Size	$c_{1i} = c_{2i}$	l_i, r_i
Centers	Triangular	Class 1 $\left(i = 1, \ldots, \frac{n}{2}\right)$	$\mathcal{U}[0, 1]$	$\mathcal{U}[0, 1]$
		Class 2 $\left(i = \frac{n}{2} + 1, \ldots, n\right)$	$\mathcal{U}[1.5, 2.5]$	$\mathcal{U}[0, 1]$
		Outlier data ($0.1n$, $0.2n$, and $0.3n$)	$\mathcal{N}(4.5, 2)$	$\mathcal{U}[0, 1]$
Spreads	Parabolic	Class 1 $\left(i = 1, \ldots, \frac{n}{2}\right)$	$\mathcal{U}[0, 1]$	$\mathcal{U}[0, 1]$
		Class 2 $\left(i = \frac{n}{2} + 1, \ldots, n\right)$	$\mathcal{U}[0, 1]$	$\mathcal{U}[1.5, 2.5]$
		Outlier data ($0.1n$, $0.2n$, and $0.3n$)	$\mathcal{U}[0, 1]$	$\mathcal{N}(4.5, 2)$
Centers and spreads	Square root	Class 1 $\left(i = 1, \ldots, \frac{n}{2}\right)$	$\mathcal{U}[0, 1]$	$\mathcal{U}[0, 1]$
		Class 2 $\left(i = \frac{n}{2} + 1, \ldots, n\right)$	$\mathcal{U}[1.5, 2.5]$	$\mathcal{U}[1.5, 2.5]$
		Outlier data ($0.1n$, $0.2n$, and $0.3n$)	$\mathcal{N}(4.5, 2)$	$\mathcal{N}(4.5, 2)$

Tab. 1: The D'Urso and De Giovanni data sets.

5. NUMERICAL EXPERIMENTS

Among the various distance measures, only those proposed in [11, 26] can compute the distance for all types of L-R fuzzy data. Therefore, our proposed method will be compared with these approaches.

5.1. Data sets

In the data design, both singleton-core and interval-core fuzzy numbers are utilized. Additionally, five types of MFs are employed: trapezoidal, triangular, parabolic, square root, and Gaussian (normal). The number of classes is either 2 or 3, and the ratio of outlier data to non-outlier data ranges from 0 to 0.3. Consequently, the data sets exhibit a suitable level of diversity.

5.1.1. Synthetic data sets

D'Urso and De Giovanni data sets [9] Forty samples ($n = 40$) of singleton-core L-R fuzzy numbers were generated, with $0.1n$, $0.2n$, and $0.3n$ additional outlier data. Three data generation schemes were considered: centers (see top of Figure 6), spreads (see middle of Figure 6), and centers and spreads (see bottom of Figure 6). Refer to Table 1 for details.

Yang and Ko data set [26] The 30 trapezoidal fuzzy numbers data set is shown in Table 2.

5.1.2. Real data set

The meteorological data set [11] is presented in Table 3. We encode the data by employing an asymmetric Gaussian MF with the center set to the average temperature, the left spread equal to the average temperature minus the min temperature, and the right spread equal to the max temperature minus the average temperature. In the

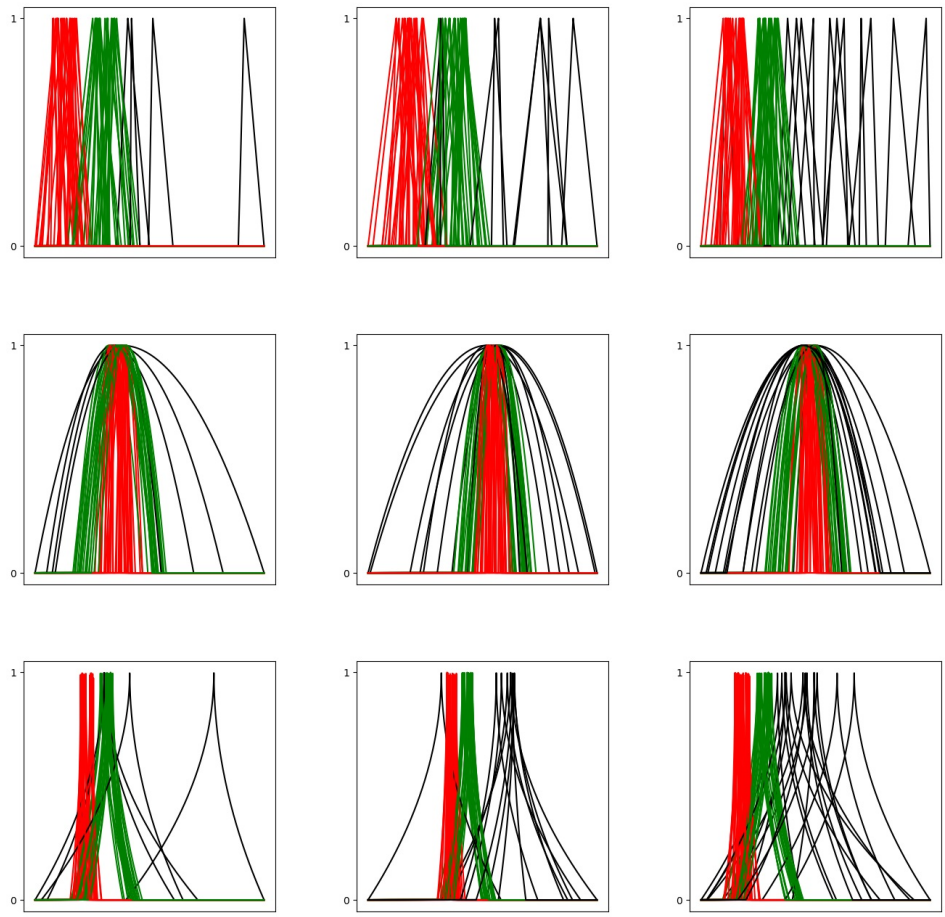


Fig. 6: The D'Urso and De Giovanni data sets: centers scheme (top), spreads scheme (middle), and centers and spreads scheme (bottom). The number of outlier data from left to right: $0.1n$, $0.2n$, and $0.3n$.

meteorological data set, there is a clearly defined group of stations with high temperatures ($i = 2, \ldots, 41$) and a clearly defined group of stations with low temperatures ($i = 43, \ldots, 82$). Additionally, there is a station with an extremely high temperature, a station with a moderate temperature, and a station with an extremely low temperature ($i = 1, 42, 83$, outlier data). Refer to Figure 7 for more details.

5.2. Experimental setup

All experiments were performed on the same machine (Operating System: macOS, Memory: 8GB, Processor: 1.1GHz dual-core Intel Core i3). We implemented the five pro-

i	c_{1i}	c_{2i}	l_i	r_i	Class	i	c_{1i}	c_{2i}	l_i	r_i	Class	i	c_{1i}	c_{2i}	l_i	r_i	Class
1	32.29	35.01	1.66	1.64	1	11	45.77	47.56	1.71	0.79	1	21	3.34	5.34	1.46	1.3	3
2	32.77	34.67	0.63	0.47	1	12	19.78	22.38	1.47	0.42	2	22	9.56	11.36	0.27	1	3
3	34.88	36.89	1.08	0.66	1	13	20.67	23.57	1.34	1.1	2	23	10.56	13.79	1.95	1.93	3
4	35.45	37.87	1.48	1.26	1	14	21.45	23.67	0.92	1.6	2	24	10.89	13.24	0.56	1.17	3
5	35.88	37.89	1.79	0.16	1	15	22.34	24.57	0.04	1.58	2	25	13.89	15.25	0.89	0.88	3
6	38.88	40.56	0.66	0.64	1	16	23.47	25.47	0.81	0.51	2	26	14.78	16.34	0.12	1.21	3
7	40.25	41.78	0.52	1.71	1	17	24.67	25.25	0.14	1.09	2	27	14.9	16.89	1.19	0.41	3
8	40.47	42.35	1.95	0.15	1	18	25.78	27.88	0.39	1.51	2	28	15.67	17.02	1.82	0.9	3
9	43.56	45.79	0.92	0.63	1	19	26.45	28.34	1.61	0.92	2	29	16.87	17.54	1.9	1.85	3
10	43.98	45.67	1.74	1.69	1	20	28.34	30.56	1.95	0.12	2	30	17.45	18.14	1.79	1.95	3

Tab. 2: The Yang and Ko data set.

i	aver.	max	min	i	aver.	max	min	i	aver.	max	min
1	43.1	50.5	35.2	31	19.3	24	15	61	0.7	2.5	-0.8
2	18.1	25.2	12.5	32	18	21.5	17	62	0.1	3	-1
3	22.6	30.6	14.4	33	19.8	24.9	15.7	63	3.9	18.3	-3.5
4	20.8	26.9	13.9	34	20.1	26.1	14.2	64	4.5	5.5	3.6
5	19.1	22.5	17.1	35	18.6	25.9	15.7	65	0.8	2.1	-0.2
6	18.5	26.5	10.6	36	18.6	22.3	15.1	66	2.9	5	-0.4
7	22.1	28.1	16.1	37	20.7	28.6	12	67	5	9.8	1.3
8	20.7	27.4	12.3	38	23	31	15.1	68	4.9	8.5	2.2
9	21.6	26.5	16.8	39	20.5	26.1	13.7	69	3.8	4.5	3
10	20.9	27.3	12.5	40	19.8	27.6	11.6	70	4.7	16.8	-3.4
11	18.1	23.9	11.2	41	21.6	29.5	16.5	71	3.3	6.2	1.3
12	21.2	26.7	18.6	42	11.7	15.5	8.8	72	4.9	8.1	0.3
13	22.9	29	15.4	43	4.2	10.4	-4.4	73	2.6	3.2	1.4
14	20.1	25.1	15.1	44	2.8	5.2	1	74	4.8	8	2
15	20.7	25.9	15.6	45	0.2	0.6	0	75	2.8	4.9	0.9
16	22.3	29.4	14	46	4.3	10.6	0.6	76	2.7	9.3	0.8
17	21.4	30.6	12.8	47	4	4.6	3.5	77	1.1	2.4	-0.3
18	22.1	28.9	15.4	48	0.4	4.7	-3.2	78	3.6	5	2.6
19	19.6	26.3	11.7	49	4.6	17.6	-5.6	79	1.7	3.6	1.1
20	18.3	27	13.5	50	4.6	14.6	-1.5	80	4.3	7.5	3
21	20.2	24.5	16.9	51	0.5	2.8	-4.4	81	1.6	5.7	-2.9
22	21.4	25.8	17.6	52	1.6	5	0.1	82	1.2	2	0
23	22.6	29	17.2	53	5	14.6	0	83	-57.9	-56.3	-60.9
24	19.1	24	10.8	54	4.7	6.4	3.8				
25	22.6	28.4	15.3	55	3.3	5.4	1.4				
26	20	24.8	15.9	56	1.9	5.7	-1.7				
27	20.4	28.7	15	57	4.4	4.8	3.7				
28	22.1	28.9	14.7	58	4.5	6.5	2.7				
29	20.1	27	15	59	2	3.9	0.7				
30	19.6	27	11.6	60	0.1	2.9	-2.8				

Tab. 3: The meteorological data set.

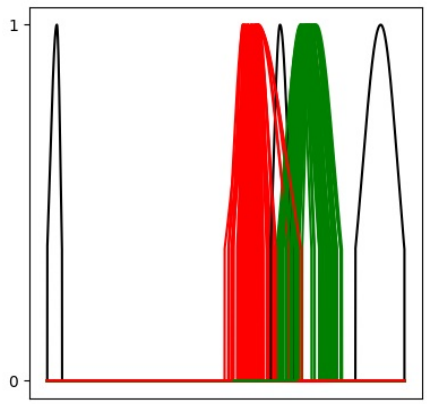


Fig. 7: The graph of meteorological data set.

posed clustering models in [11] in Google Colab, namely FcOMdC-FD SQR, FcOMdC-FD LIN, FcOMdC-FD HUB, FcOMdC-FD SIG, and FcOMdC-FD LOG, based on the original paper, along with our proposed clustering model.

The fuzziness parameter m plays a crucial role in fuzzy clustering. Values too close to 1 result in a partition with memberships close to 0 or 1, while excessively large values cause disproportionate overlap, with memberships close to $\frac{1}{c}$ (where c is the number of clusters). Chen et al. [5] proposed an improved fuzzy c-means clustering by varying the fuzziness parameter, to overcome the issue of tuning this parameter. In practice, $m = 1.5, 2$ are the most popular choices in fuzzy clustering, which we also follow in our experiments.

The effectiveness of the competing models is measured using the Rand Index (RI) [23]. The closer the RI values are to 1, the better the model's performance. The mean RI, its confidence interval, and the mean runtime are reported for 1,000 iterations of the experiment.

5.3. Results and discussion

The following section presents a comprehensive analysis of the effectiveness of different methods across various data sets and outlier levels. Tables 4, 5, and 6 summarize the results for each method. Below is a detailed discussion of the key findings:

- In the centers scheme with 30% outlier data, both the FcOMdC-FD, SQR and FcOMdC-FD, HUB methods resulted in a mean RI of 0.4872 across 1,000 iterations. Additionally, all RI values were identical at 0.4872, indicating zero variance. As shown in Table 7, all fuzzy data in either class 1 or class 2 were clustered into cluster 1, which can be explained by the fact that the quadratic component of the Huber loss, similar to squared loss, penalizes outlier data. For this data set, the

Scheme	Outlier data	Method	Mean RI $[0,1]$	95% CI for RI	Mean runtime (s)
Centers	10%	FcOMdC-FD, SQR [11, 26]	0.9574	(0.9487,0.9662)	**0.5643**
		FcOMdC-FD, LIN [11]	0.9738	(0.9668,0.9808)	1.0740
		FcOMdC-FD, HUB [11] ($\delta = 5$)	0.9579	(0.9493,0.9665)	0.7111
		FcOMdC-FD, SIG [11] ($\alpha = \beta = 2$)	0.9477	(0.9381,0.9573)	0.7020
		FcOMdC-FD, LOG [11]	0.9703	(0.9628,0.9777)	1.2104
		ResFClustFD	**0.9789**	(0.9744,0.9835)	10.0993
	20%	FcOMdC-FD, SQR [11, 26]	0.8349	(0.8230,0.8467)	1.1685
		FcOMdC-FD, LIN [11]	0.8800	(0.8665,0.8935)	1.7194
		FcOMdC-FD, HUB [11] ($\delta = 5$)	0.8333	(0.8213,0.8454)	1.8639
		FcOMdC-FD, SIG [11] ($\alpha = \beta = 2$)	0.8290	(0.8167,0.8413)	0.8237
		FcOMdC-FD, LOG [11]	0.9062	(0.8938,0.9185)	**0.7593**
		ResFClustFD	**0.9545**	(0.9499,0.9591)	12.8348
	30%	FcOMdC-FD, SQR [11, 26]	0.4872	Zero Variance	0.9199
		FcOMdC-FD, LIN [11]	0.8085	(0.7932,0.8238)	2.1354
		FcOMdC-FD, HUB [11] ($\delta = 5$)	0.4872	Zero Variance	2.4856
		FcOMdC-FD, SIG [11] ($\alpha = \beta = 2$)	0.7686	(0.7543,0.7828)	**0.8191**
		FcOMdC-FD, LOG [11]	0.8493	(0.8348,0.8637)	1.0650
		ResFClustFD	**0.8797**	(0.8710,0.8884)	12.1248
Spreads	10%	FcOMdC-FD, SQR [11, 26]	0.7613	(0.7469,0.7757)	**0.2974**
		FcOMdC-FD, LIN [11]	0.7334	(0.7211,0.7457)	1.0637
		FcOMdC-FD, HUB [11] ($\delta = 5$)	0.7697	(0.7554,0.7840)	0.4631
		FcOMdC-FD, SIG [11] ($\alpha = \beta = 2$)	0.7920	(0.7764,0.8076)	0.4087
		FcOMdC-FD, LOG [11]	0.8143	(0.8015,0.8272)	0.3696
		ResFClustFD	**0.9965**	(0.9944,0.9986)	10.5661
	20%	FcOMdC-FD, SQR [11, 26]	0.6295	(0.6188,0.6402)	**0.3083**
		FcOMdC-FD, LIN [11]	0.6677	(0.6579,0.6776)	0.8904
		FcOMdC-FD, HUB [11] ($\delta = 5$)	0.6381	(0.6269,0.6493)	0.4883
		FcOMdC-FD, SIG [11] ($\alpha = \beta = 2$)	0.7200	(0.7065,0.7335)	0.4708
		FcOMdC-FD, LOG [11]	0.7518	(0.7404,0.7632)	1.0588
		ResFClustFD	**0.9632**	(0.9565,0.9699)	13.5412
	30%	FcOMdC-FD, SQR [11, 26]	0.5851	(0.5783,0.5919)	0.7298
		FcOMdC-FD, LIN [11]	0.6169	(0.6060,0.6279)	**0.4958**
		FcOMdC-FD, HUB [11] ($\delta = 5$)	0.5860	(0.5792,0.5929)	1.1603
		FcOMdC-FD, SIG [11] ($\alpha = \beta = 2$)	0.5977	(0.5902,0.6053)	0.5846
		FcOMdC-FD, LOG [11]	0.6691	(0.6565,0.6817)	0.5015
		ResFClustFD	**0.7419**	(0.7315,0.7524)	13.7148
Centers and spreads	10%	FcOMdC-FD, SQR [11, 26]	0.9358	(0.9256,0.9460)	**0.3635**
		FcOMdC-FD, LIN [11]	0.9492	(0.9397,0.9587)	1.2931
		FcOMdC-FD, HUB [11] ($\delta = 5$)	0.8944	(0.8815,0.9072)	0.5650
		FcOMdC-FD, SIG [11] ($\alpha = \beta = 2$)	0.9385	(0.9281,0.9488)	2.2938
		FcOMdC-FD, LOG [11]	0.9728	(0.9657,0.9799)	1.0480
		ResFClustFD	**0.9954**	(0.9930,0.9978)	10.8193
	20%	FcOMdC-FD, SQR [11, 26]	0.7565	(0.7463,0.7668)	0.8064
		FcOMdC-FD, LIN [11]	0.8456	(0.8310,0.8602)	1.3559
		FcOMdC-FD, HUB [11] ($\delta = 5$)	0.8282	(0.8132,0.8432)	**0.7836**
		FcOMdC-FD, SIG [11] ($\alpha = \beta = 2$)	0.8056	(0.7902,0.8211)	2.5717
		FcOMdC-FD, LOG [11]	0.8349	(0.8200,0.8497)	1.4341
		ResFClustFD	**0.9642**	(0.9573,0.9712)	13.2382
	30%	FcOMdC-FD, SQR [11, 26]	0.5041	(0.5027,0.5055)	0.7245
		FcOMdC-FD, LIN [11]	0.7942	(0.7793,0.8090)	2.4090
		FcOMdC-FD, HUB [11] ($\delta = 5$)	0.5980	(0.5866,0.6095)	1.7847
		FcOMdC-FD, SIG [11] ($\alpha = \beta = 2$)	0.7959	(0.7811,0.8108)	**0.6987**
		FcOMdC-FD, LOG [11]	0.8411	(0.8264,0.8558)	0.7174
		ResFClustFD	**0.9646**	(0.9579,0.9714)	14.8094

The best result is marked in bold.

Tab. 4: Clustering results for the D'Urso and De Giovanni data sets, with $m = 2$.

Method	Mean RI $[0,1]$	95% CI for RI	Mean runtime (s)
FcOMdC-FD, SQR [11, 26]	0.9154	(0.9080,0.9228)	0.3481
FcOMdC-FD, LIN [11]	0.9107	(0.9026,0.9188)	0.4446
FcOMdC-FD, HUB [11] ($\delta = 5$)	0.9064	(0.8986,0.9141)	0.5889
FcOMdC-FD, SIG [11] ($\alpha = \beta = 2$)	0.8183	(0.8112,0.8253)	**0.2932**
FcOMdC-FD, LOG [11]	0.8339	(0.8264,0.8414)	0.3421
ResFClustFD	**0.9581**	(0.9577,0.9586)	10.1038

The best result is marked in bold.

Tab. 5: Clustering results for the Yang and Ko data set, with $m = 1.5$.

Method	Mean RI $[0,1]$	95% CI for RI	Mean runtime (s)
FcOMdC-FD, SQR [11, 26]	0.8441	(0.8295,0.8586)	**1.1227**
FcOMdC-FD, LIN [11]	0.9565	(0.9476,0.9653)	5.3534
FcOMdC-FD, HUB [11] ($\delta = 5$)	0.9797	(0.9736,0.9859)	5.1217
FcOMdC-FD, SIG [11] ($\alpha = \beta = 2$)	0.8763	(0.8637,0.8889)	5.6343
FcOMdC-FD, LOG [11]	0.8918	(0.8797,0.9038)	6.0408
ResFClustFD	**0.9938**	(0.9915,0.9961)	22.7968

The best result is marked in bold.

Tab. 6: Clustering results for the meteorological data set, with $m = 2$.

values were TP=380, TN=0, FP=400, and FN=0 (with outlier data excluded), resulting in an RI=0.4872.

- ResFClustFD consistently achieved the highest RI across all data sets and outlier levels, indicating superior clustering performance and making it the best choice for scenarios where accurate clustering is critical.
- FcOMdC-FD methods generally performed faster, with the SQR loss often being the quickest, suggesting they are more computationally efficient.
- While ResFClustFD provides the best clustering results, its runtime is significantly longer, which might be a limitation in real-time applications or large-scale data analysis.
- As the percentage of outliers increases, the performance (RI) of all methods generally decreases. This trend is more pronounced in the FcOMdC-FD methods, indicating that these methods might be less robust to higher outlier percentages compared to ResFClustFD.
- The reduction in RI with increasing outlier percentage is less steep for ResFClustFD, highlighting its robustness in handling outlier data.
- Among FcOMdC-FD variants, the LOG variant often shows better RI compared to others, but it is not always the fastest.

i	Class	Cluster	i	Class	Cluster	i	Class	Cluster	i	Class	Cluster
1	-	1	14	2	1	27	2	1	40	-	1
2	2	1	15	1	1	28	2	1	41	1	1
3	2	1	16	2	1	29	2	1	42	2	1
4	1	1	17	1	1	30	2	1	43	1	1
5	1	1	18	1	1	31	1	1	44	1	1
6	-	2	19	1	1	32	-	2	45	1	1
7	-	1	20	2	1	33	-	2	46	1	1
8	-	1	21	-	2	34	2	1	47	1	1
9	2	1	22	-	2	35	2	1	48	2	1
10	2	1	23	1	1	36	-	2	49	-	2
11	1	1	24	2	1	37	1	1	50	2	1
12	2	1	25	-	2	38	1	1	51	2	1
13	1	1	26	1	1	39	1	1	52	2	1

Tab. 7: Clustering results for the centers scheme with 30% outlier data, obtained using the FcOMdC-FD, SQR and FcOMdC-FD, HUB methods.

- The performance of different methods varies with the nature of the data set. For instance, in data sets like the 'centers scheme' and 'Yang and Ko', where the data might be less complex and the shape of the MF is linear, the differences in RI between methods are smaller, making runtime a more crucial factor. In contrast, in data sets such as the 'spreads scheme', 'centers and spreads scheme', and 'meteorological', where the data is more complex and the shape of the MF is non-linear, ResFClustFD's RI is significantly higher than that of other methods, emphasizing its superior effectiveness.

6. CONCLUSION

In this paper, we proposed a fuzzy clustering method called ResFClustFD, which effectively detects small differences in the shapes of the left and right tails of membership functions. We first converted one-dimensional fuzzy data into two-dimensional image matrices and then utilized the pre-trained ResNet50 network as a feature extractor. The FCM clustering method was applied to the output from the last convolutional layer of ResNet50. Our experiments demonstrate that ResFClustFD achieves high accuracy in clustering both synthetic and real data sets. It consistently produced the highest Rand Index across various scenarios, indicating superior clustering performance, particularly where the data is more complex and the shape of the membership function is non-linear. For applications requiring high clustering accuracy, ResFClustFD is recommended despite its longer runtime. Future research could explore alternative architectures like vgg or densenet models for enhanced feature extraction. Additionally, extending the method to multi-dimensional fuzzy data presents an exciting opportunity.

ACKNOWLEDGMENTS

The authors are grateful to the editor and the anonymous referees for their thorough revision of the manuscript and their valuable suggestions.

FUNDING

This work is based upon research funded by Iran National Science Foundation (INSF) and Institute for Advanced Studies in Basic Sciences (IASBS) under Project No. 4024435.

(Received October 26, 2024)

REFERENCES

[1] R. M. Alguliyev, R. M. Aliguliyev, and R. G. Alakbarov: Constrained k-means algorithm for resource allocation in mobile cloudlets. Kybernetika *59* (2023), 1, 88–109.

[2] B. Bede: Mathematics of Fuzzy Sets and Fuzzy Logic. Springer, Berlin, Heidelberg 2013.

[3] J. C. Bezdek: Pattern Recognition with Fuzzy Objective Function Algorithms. Springer, New York 1981.

[4] G. Castellano and G. Vessio: A deep learning approach to clustering visual arts. Int. J. Computer Vision *130* (2022), 11, 2590–2605. DOI:10.1007/s11263-022-01664-y

[5] Y. Chen, S. Zhou, X. Zhang, D. Liu, and C. Fu: Improved fuzzy c-means clustering by varying the fuzziness parameter. Pattern Recogn. Lett. *157* (2022), 60–66. DOI:10.1016/j.patrec.2022.03.017

[6] R. Coppi, P. D'Urso, and P. Giordani: A note on the algorithms for determining the model structureFuzzy and possibilistic clustering for fuzzy data. Comput. Statist. Data Analysis *56* (2012), 4, 915–927. DOI:10.1016/j.csda.2010.09.013

[7] J. Deng, W. Dong, R. Socher, L. J. Li, K. Li, and L. Fei-Fei: Imagenet: A largescale hierarchical image database. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2009, pp. 248–255.

[8] W. Ding, M. Abdel-Basset, H. Hawash, and W. Pedrycz: Multimodal infant brain segmentation by fuzzy-informed deep learning. IEEE Trans. Fuzzy Systems *30* (2021), 4, 1088–1101. DOI:10.1109/TFUZZ.2021.3052461

[9] P. D'Urso and L. De Giovanni: Robust clustering of imprecise data. Chemometr. Intell. Labor. Systems *136* (2014), 58–80. DOI:10.1016/j.chemolab.2014.05.004

[10] P. D'Urso and P. Giordani: A weighted fuzzy c-means clustering model for fuzzy data. Comput. Statist. Data Analysis *50* (2006), 6, 1496–1523. DOI:10.1016/j.csda.2004.12.002

[11] P. D'Urso and J. M. Leski: Fuzzy clustering of fuzzy data based on robust loss functions and ordered weighted averaging. Fuzzy Sets Systems *389* (2020), 1–28. DOI:10.1016/j.fss.2019.03.017

[12] E. Eskandari and A. Khastan: A robust fuzzy clustering model for fuzzy data based on an adaptive weighted L1 norm. Iranian J. Fuzzy Systems *20* (2023), 6, 1–20.

[13] E. Eskandari and A. Khastan: Investigating the effect of using different loss functions on the performance of the fuzzy clustering model for fuzzy data in the presence of outlier data. Fuzzy Systems Appl. *7* (2024), 1, 109–123.

[14] E. Eskandari, A. Khastan, and S. Tomasiello: Improved determination of the weights in a clustering approach based on a weighted dissimilarity measure between fuzzy data. In: IEEE International Conference on Fuzzy Systems (FUZZ-IEEE) 2022, pp. 1–6.

[15] Q. Feng, L. Chen, C. L. P. Chen, and L. Guo: Deep fuzzy clustering – a representation learning approach. IEEE Trans. Fuzzy Systems *28* (2020), 7, 1420–1433. DOI:10.1109/TFUZZ.2020.2998378

[16] M. B. Ferraro and P. Giordani: Possibilistic and fuzzy clustering methods for robust analysis of non-precise data. Int. J. Approx. Reasoning *88* (2017), 23–38. DOI:10.1016/j.ijar.2017.05.002

[17] J. C. Figueroa-Garcia, C. A. Varón-Gaviria, and J. L. Barbosa-Fontecha: Fuzzy random variable generation using α-cuts. IEEE Trans. Fuzzy Systems *29* (2019), 3, 539–548.

[18] Y. Gao, W. Wang, J. Xie, and J. Pan: A new robust fuzzy c-means clustering method based on adaptive elastic distance. Knowledge Based Systems *237* (2022), 107769. DOI:10.1016/j.knosys.2021.107769

[19] K. He, X. Zhang, S. Ren, and J. Sun: Deep residual learning for image recognition. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2016, pp. 770–778.

[20] K. Huang, Y. Zhang, H. D. Cheng, P. Xing, and B. Zhang: Semantic segmentation of breast ultrasound image with fuzzy deep learning network and breast anatomy constraints. Neurocomputing *450* (2021), 319–335. DOI:10.1016/j.neucom.2021.04.012

[21] W. Hung and M. Yang: Fuzzy clustering on LR-type fuzzy numbers with an application in Taiwanese tea evaluation. Fuzzy Sets Systems *150* (2005), 3, 561–577. DOI:10.1016/j.fss.2004.04.007

[22] N. Kartli, E. Bostanci, and M. S. Guzel: Heuristic algorithm for an optimal solution of fully fuzzy transportation problem. Computing *106* (2024), 10, 3195–3227. DOI:10.1007/s00607-024-01319-5

[23] W. M. Rand: Objective criteria for the evaluation of clustering methods. J. American Statist. Assoc. *66* (1971), 336, 846–850. DOI:10.1080/01621459.1971.10482356

[24] L. Su and X. Cao: Fuzzy autoencoder for multiple change detection in remote sensing images. J. Appl. Remote Sensing *12* (2018), 3, 035014. DOI:10.1117/1.jrs.12.035014

[25] X. Wei, D. Lu, X. Cao, L. Su, L. Wang, H. Guo, Y. Hou, and X. He: A fuzzy artificial neural network-based method for Cerenkov luminescence tomography. AIP Advances *9* (2019), 6, 065105.

[26] M. Yang and C. Ko: On a class of fuzzy c-numbers clustering procedures for fuzzy data. Fuzzy Sets Systems *84* (1996), 1, 49–60. DOI:10.1016/0165-0114(95)00308-8

Alireza Khastan, Corresponding author. Department of Mathematics, Institute for Advanced Studies in Basic Sciences, 444 Prof. Yousef Sobouti Blvd., Zanjan 45137-66731. Iran.

e-mail: khastan@iasbs.ac.ir

Elham Eskandari, Department of Mathematics, Institute for Advanced Studies in Basic Sciences, 444 Prof. Yousef Sobouti Blvd., Zanjan 45137-66731. Iran.

e-mail: eskandari.e69@gmail.com