

Robust and Sparse n-Pythagorean Fuzzy K-Means Clustering

Manas Singh Sakrwar* and A.S Ranadive

Department of Mathematics,

Guru Ghasidas Vishwavidyalaya, Bilaspur (C.G.), India

E-mail: manassingh0826@outlook.com ; asranadive04@yahoo.co.in

(Received July 18, 2025, Accepted August 31, 2025)

Abstract: In this paper, we propose a novel clustering approach termed the Robust and Sparse n-Pythagorean Fuzzy K-Means algorithm, which extends the standard n-Pythagorean fuzzy K-Means by addressing two key limitations: sensitivity to outliers and lack of membership sparsity. To enhance robustness, we replace the conventional squared distance term with a robust loss function, making the algorithm more resilient to noise and outliers. Furthermore, we introduce an additional sparsity-inducing regularization term that encourages each data point to be associated with a limited number of clusters, thereby improving interpretability and reducing overfitting. Experimental evaluations on several benchmark datasets demonstrate that the proposed method achieves superior clustering performance by effectively balancing robustness and sparsity in soft clustering scenarios.

Keywords: n-Pythagorean Fuzzy Sets, Fuzzy Clustering, Sparse Clustering.

1. Introduction

Clustering, a fundamental unsupervised learning technique, groups similar data points without prior class labels. Fuzzy C-Means (FCM), introduced by Dunn¹ and refined by Bezdek², enables soft clustering by assigning membership degrees to data points based on their proximity to cluster centroids. Despite its effectiveness in handling overlapping data, FCM is notably sensitive to noise, outliers and struggles with feature selection in high-dimensional settings.

To address these limitations, several extensions have been proposed. For instance, Sparse Fuzzy C-Means (SFCM) by Qiu *et al.*³, inspired by Witten and Tibshirani's Sparse K-Means (SPKM)⁴, incorporates sparsity-inducing regularization to enhance feature selection. Similarly, Robust and Sparse

Fuzzy K-Means (RSFKM) by Xu *et al.*⁵ integrates robust loss functions, such as the ℓ_1 and iterative re-weighted procedures⁶, to reduce the influence of outliers.

A primary drawback of many FCM variants, including those extended to handle sparse features, is their lack of robustness to outliers due to their reliance on Euclidean distance, which performs poorly in noisy environments. Efforts to mitigate this include replacing the Euclidean norm with kernel distances⁷. However, such methods often lack spatial context and can suffer from slow convergence. To improve robustness, several enhanced Fuzzy K-Means (FKM) algorithms have been proposed: Ji *et al.*⁸ introduced local spatial weights to suppress outliers and reduce ambiguity; Zhao *et al.*⁹ added non-local spatial constraints to better handle image noise; and Ji *et al.*¹⁰ leveraged embedded spatial information for further performance gains. Other approaches include incorporating a normed kernel function and center initialization for robustness¹¹, or proposing adaptive spatial information-theoretic fuzzy clustering to enhance reliability in noisy environments¹².

The second significant challenge in fuzzy clustering lies in selecting appropriate regularization. While regularization demonstrably improves clustering robustness¹³, various regularization-based clustering models have been developed¹⁴⁻¹⁸. The empirical and computationally expensive nature of selecting the regularization constant typically limits their practical application.

Furthermore, traditional fuzzy set-based clustering methods often fall short in accounting for uncertainty and hesitation in membership assignments. To overcome this, Atanassov¹⁹ introduced Intuitionistic Fuzzy Sets (IFSs), which provide an additional degree of freedom to model uncertainty by explicitly incorporating both membership and non-membership degrees, along with the possibility of hesitation. Xu²⁰ then extended FCM to the intuitionistic fuzzy domain, proposing the Intuitionistic Fuzzy C-Means (IFCM) algorithm.

Further advancing this concept, Bryniarska's²¹ n -Pythagorean fuzzy sets generalize IFSs, allowing the sum of the n^{th} powers of membership and non-membership to lie within the unit interval. Building on this framework, Xian *et al.* proposed the n -Pythagorean Fuzzy C-Means (nPyFCM)²². While nPyFCM extends FCM to a more flexible fuzzy domain, it unfortunately retains its core limitations: sensitivity to outliers due to reliance on Euclidean distance and lack of inherent membership sparsity.

To address these critical challenges comprehensively, we propose the Robust and Sparse n -Pythagorean Fuzzy K-Means (RS-nPyFKM) algorithm. RS-nPyFKM significantly enhances robustness by replacing the conventional

squared Euclidean distance with noise-tolerant loss functions using capped ℓ_1 , which effectively reduce the impact of outliers. Additionally, we introduce a sparsity-inducing penalty on the membership matrix, promoting selective cluster associations for each data point. This not only improves interpretability but also reduces overfitting. A dynamic strategy for tuning the regularization parameter γ is also proposed, which significantly enhances adaptability across diverse datasets. Extensive experiments on benchmark datasets demonstrate that RS-nPyFKM consistently outperforms existing methods, achieving robust and interpretable clustering results while effectively balancing these crucial aspects in soft clustering applications.

2. Preliminaries

The fuzzy set was introduced by Zadeh. Since in real life there is uncertainty in giving exact membership, and therefore Atanassov¹⁹ introduced intuitionistic fuzzy sets. Later, Yager introduced Pythagorean fuzzy sets. n -Pythagorean fuzzy set is a generalization of Pythagorean sets. Its definition is given below:

Definition 2.1.²¹ Let Ω be a universal set then n -Pythagorean fuzzy set A is a collection of elements of Ω with membership and non-membership values.

$$A = \{(\omega, \mu_A(\omega), \nu_A(\omega)) | \omega \in \Omega\},$$

where $\mu_A, \nu_A: \Omega \rightarrow [0,1]$ are membership function and non-membership function, respectively, with property $0 \leq \mu_A^n(\omega) + \nu_A^n(\omega) \leq 1$. Here, the hesitance degree is

$$\pi_A(\omega) = (1 - \mu_A^n(\omega) - \nu_A^n(\omega))^{1/n}.$$

An n -Pythagorean fuzzy generator (n -PFG) is a function that transforms a standard fuzzy set into an n -Pythagorean fuzzy set. This transformation assigns appropriate non-membership degrees to each element based on its membership degree while ensuring compliance with the formal definition of an n -Pythagorean fuzzy set.

We define the n -Pythagorean fuzzy generator as:

Definition 2.2. A function $f: [0,1] \rightarrow [0,1]$ is said to be an n -PFG if

1. $f(x) \leq (1 - x^n)^{\frac{1}{n}} \forall x \in [0,1]$.
2. $f(0) \leq 1$ and $f(1) = 0$.

For a fuzzy set $A = \{(\omega, \mu_A(\omega)) | \omega \in \Omega\}$, we can generate an n -Pythagorean fuzzy set (n -PFS) using an n -Pythagorean fuzzy generator (n -PFG) as

$$A = \{(\omega, \mu_A(\omega), f(\mu_A(\omega))) | \omega \in \Omega\}$$

Our algorithm incorporated hesitation into a general fuzzy framework using n -Pythagorean fuzzy sets, where the membership degree μ was bounded as

$$\mu \in [\mu, (\mu^n + \pi^n)^{1/n}].$$

We used the upper bound of μ in our updates.

Although alternative generators for the non-membership degree ν existed such we adopted the Yager-type function:

$$(2.1) \quad f_\alpha(\mu(\omega)) = (1 - (\mu(\omega))^\alpha)^{1/\alpha}, \quad 0 < \alpha \leq n.$$

3. Robust Sparse n -Pythagorean Fuzzy K Means Clustering

We have a dataset $\Omega = \{\omega_1, \omega_2, \dots, \omega_N\}$ where ω_i is a vector with dimension d and N represents the number of samples. Our goal is to distribute each data point into different clusters, where K represents the number of clusters. We determine these clusters by minimizing intra-cluster distance and maximizing inter-cluster distance. Each cluster has a prototype, referred to as its center. We assign the membership of each data point based on its distance from the centers. Then, based on the data points in the clusters, we calculate new centers. This process continues until the objective function reaches its minimum. Our output will be a matrix $\mathbf{U}_{N \times K}$, where the (i, j) -th entry is represented as u_{ij} and gives the membership value of the i^{th} data point in the j^{th} cluster. Additionally, we will obtain a matrix of cluster centers $\mathbf{C} = [\mathbf{c}_1, \dots, \mathbf{c}_K]^T$, where the j^{th} entry represents the cluster center of the j^{th} cluster.

The most common way to find the $\mathbf{U}$ and $\mathbf{C}$ to use the least square loss function.

$$(3.1) \quad \min_{\mathbf{U} \mathbf{1} = \mathbf{1}, \mathbf{U} \geq 0, \mathbf{C}} \sum_{i=1}^N \sum_{j=1}^K u_{ij} \|\omega_i - \mathbf{c}_j\|_2^2.$$

The least squares loss is sensitive to outliers and can be replaced by the capped ℓ_1 -norm for improved robustness. This norm assigns zero membership to points lying beyond a threshold ϵ . Therefore, to provide better robustness, we use capped ℓ_1 -norm and introduce a new objective function based on capped ℓ_1 norm for n -Pythagorean setting.

$$(3.2) \quad \min_{\mathbf{U} \mathbf{1} = \mathbf{1}, \mathbf{U} \geq 0, \mathbf{C}} \sum_{i=1}^N \sum_{j=1}^K u_{ij} \min(\|\omega_i - \mathbf{c}_j\|_2, \epsilon) + \gamma \|\mathbf{U}\|_F^2.$$

To promote sparsity, we include a regularization term. Note that if a data point is fully assigned to one cluster, $\sum_{j=1}^K u_{ij}^2 = 1$, while distributing membership across multiple clusters reduces this sum. Thus, minimizing $\|\mathbf{U}\|_F^2$ encourages more distributed assignments. The parameter γ controls this behavior: a larger γ results in less sparse (more distributed) clustering.

Equation (3.1) can be optimized using the iterative re-weighted method, where the ℓ_1 -norm is approximated by a weighted ℓ_2 -norm with the help of an auxiliary variable. Using this approach, the objective function (2) can be reformulated as:

$$(3.3) \quad \min_{\mathbf{U} \mathbf{1} = \mathbf{1}, \mathbf{U} \geq \mathbf{0}, \mathbf{C}, \mathbf{S}} \sum_{i=1}^N \sum_{j=1}^K s_{ij} u_{ij} \|\boldsymbol{\omega}_i - \mathbf{c}_j\|_2^2 + \gamma \|\mathbf{U}\|_F^2$$

We solve Equation (3.2) by alternately updating the variables $\mathbf{S}$, $\mathbf{U}$, and $\mathbf{C}$.

3.1 Update of $\mathbf{S}$ with Fix $\mathbf{C}$ and $\mathbf{U}$

The auxiliary variable s_{ij} is updated using the following rule:

$$(3.4) \quad s_{ij} = \begin{cases} \frac{1}{2\|\boldsymbol{\omega}_i - \mathbf{c}_j\|_2}, & \text{if } \|\boldsymbol{\omega}_i - \mathbf{c}_j\|_2 \leq \epsilon \\ 0, & \text{otherwise} \end{cases}$$

When a data point $\boldsymbol{\omega}_i$ lies far from the cluster center $\mathbf{c}_j$, i.e., $\|\boldsymbol{\omega}_i - \mathbf{c}_j\|_2 \geq \epsilon$, the corresponding weight s_{ij} is set to zero. This thresholding mechanism enhances the robustness of the model by effectively down-weighting outliers during the clustering process.

3.2 Update of $\mathbf{C}$ with Fix $\mathbf{S}$ and $\mathbf{U}$

With $\mathbf{U}$ and $\mathbf{S}$ fixed, the objective function in Equation (3.3) reduces to for differentiation:

$$(3.5) \quad \min_{\mathbf{C}} \sum_{i=1}^N \sum_{j=1}^K s_{ij} u_{ij} \|\boldsymbol{\omega}_i - \mathbf{c}_j\|_2^2.$$

This problem decomposes into K independent subproblems of the form:

$$(3.6) \quad \min_{\mathbf{c}_j} \sum_{i=1}^N s_{ij} u_{ij} \|\boldsymbol{\omega}_i - \mathbf{c}_j\|_2^2.$$

Each subproblem in Equation (7) involves minimizing a quadratic function. The update rule for the cluster center $\mathbf{c}_j$ is derived by differentiating the objective with respect to $\mathbf{c}_j$ and setting the derivative to zero. This yields the following closed-form solution:

$$(3.7) \quad \mathbf{c}_j = \frac{\sum_{i=1}^N s_{ij} u_{ij} \boldsymbol{\omega}_i}{\sum_{i=1}^N s_{ij} u_{ij}}.$$

This update ensures that each cluster center is a weighted average of the data points, where the weights incorporate both membership values and the robustness-inducing auxiliary variables.

3.3 Update of $\mathbf{U}$ with Fix $\mathbf{C}$ and $\mathbf{S}$

For fixed $\mathbf{C}$ and $\mathbf{S}$, we can rewrite the objective function as follows:

$$(3.8) \quad \min_{\mathbf{U} \mathbf{1} = \mathbf{1}, \mathbf{U} \geq \mathbf{0}, \mathbf{C}} \sum_{i=1}^N \sum_{j=1}^K \left(s_{ij} u_{ij} \|\boldsymbol{\omega}_i - \mathbf{c}_j\|_2^2 + \gamma u_{ij}^2 \right).$$

Since the constraints on $\mathbf{U}$ are imposed row-wise and element-wise for each data point $\boldsymbol{\omega}_i$, Equation (3.8) can be reformulated as N independent subproblems.

$$(3.9) \quad \min_{\mathbf{u}_i^T \mathbf{1}=1, \mathbf{u}_i \geq 0} \sum_{j=1}^K (h_{ij} u_{ij} + \gamma u_{ij}^2).$$

Here, $\mathbf{u}_i$ represents the i -th row of the matrix $\mathbf{U}$, and $h_{ij} = s_{ij} \|\boldsymbol{\omega}_i - \mathbf{c}_j\|_2^2$ denotes the (i, j) -th entry of the matrix $\mathbf{H}$.

3.4 Simplified Objective Function

Upon simplification, Equation (3.9) can be expressed in the following form:

$$(3.10) \quad \min_{\mathbf{u}_i^T \mathbf{1}=1, \mathbf{u}_i \geq 0} \|\mathbf{u}_i - \mathbf{g}_i\|_2^2.$$

Here $\mathbf{g}_i = -\frac{\mathbf{h}_i}{2\gamma}$ is a constant at this stage.

3.5 Lagrangian and KKT Conditions

The Lagrangian function of problem Equation (11) is

$$(3.11) \quad \mathcal{L}(\mathbf{u}_i, \lambda, \boldsymbol{\delta}) = \|\mathbf{u}_i - \mathbf{g}_i\|_2^2 - \lambda(\mathbf{u}_i^T \mathbf{1} - 1) - \boldsymbol{\delta}^T \mathbf{u}_i$$

where λ is a scalar and $\boldsymbol{\delta}$ is a Lagrangian coefficient vector to be determined. Assume that $\mathbf{u}^*$ is the optimal solution and λ^* and $\boldsymbol{\delta}^*$ are the corresponding Lagrangian multipliers. According to the KKT conditions, we have the following equations:

$$(3.12) \quad \begin{cases} \forall t \in [1, K], & u_t^* - g_t - \lambda^* - \delta_t^* = 0 \\ \forall t \in [1, K], & u_t^* \geq 0 \\ \forall t \in [1, K], & \delta_t^* \geq 0 \\ \forall t \in [1, K], & u_t^* \delta_t^* = 0 \end{cases}$$

Here u_t^* is the t^{th} scalar element of a vector $\mathbf{u}^*$. The first Equation from (3.12) can be rewritten in vector form as:

$$(3.13) \quad \mathbf{u}^* - \mathbf{g} - \lambda^* \mathbf{1} - \boldsymbol{\delta}^* = 0.$$

3.6 Derivation of λ^*

Considering the constraint $\mathbf{u}^T \mathbf{1} = 1$, Equation (3.13) can be reformulated as

$$(3.14) \quad \lambda^* = \frac{\mathbf{1} - \mathbf{1}^T \mathbf{g} - \mathbf{1}^T \boldsymbol{\delta}^*}{K}.$$

Substituting (3.14) into (3.13), we have:

$$(3.15) \quad \mathbf{u}^* = \mathbf{g} - \frac{\mathbf{1} \mathbf{1}^T}{K} \mathbf{g} + \frac{\mathbf{1}}{K} - \frac{\mathbf{1}^T \boldsymbol{\delta}^*}{K} \mathbf{1} + \boldsymbol{\delta}^*.$$

3.7 Simplified Expression for $\mathbf{u}^*$

Denote $\bar{\boldsymbol{\delta}}^* = \frac{\mathbf{1}^T \boldsymbol{\delta}^*}{K}$ and $\mathbf{m} = \mathbf{g} - \frac{\mathbf{1} \mathbf{1}^T}{K} \mathbf{g} + \frac{\mathbf{1}}{K}$, the above equation can be rewritten as:

$$(3.16) \quad \mathbf{u}^* = \mathbf{m} + \boldsymbol{\delta}^* - \bar{\boldsymbol{\delta}}^* \mathbf{1}.$$

Therefore, for each $t \in [1, K]$, we have:

$$(3.17) \quad u_t^* = m_t + \delta_t^* - \bar{\delta}^*.$$

3.8 Final Update Rule for $\mathbf{u}_t^*$

According to Equations (3.12) and (3.17), we have:

$$(3.18) \quad u_t^* = (m_t - \bar{\delta}^*)_+$$

Here $(m_t - \bar{\delta}^*)_+ = \max(m_t - \bar{\delta}^*, 0)$.

3.8 Solving for $\bar{\delta}^*$

We can rewrite Equation (18) as $\delta_t^* = u_t^* + \bar{\delta}^* - m_t$ such that $\delta_t^* = (\bar{\delta}^* - m_t)_+$. Therefore, using the constraint $\mathbf{u}^T \mathbf{1} = 1$ we have:

$$(3.19) \quad f(\bar{\delta}) = \frac{1}{K} \sum_{t=1}^K (m_t - \bar{\delta})_+ - 1,$$

and optimal $\bar{\delta}^*$ should satisfy $f(\bar{\delta}^*) = 0$. When Equation (3.19) equals zero, the optimal $\bar{\delta}^*$ can be obtained by Newton's method, namely:

$$\bar{\delta}_{(k+1)} = \bar{\delta}_k - \frac{f(\bar{\delta})}{f'(\bar{\delta})}.$$

Once the optimal value of $\bar{\delta}$ is obtained, we proceed to update the membership vector u_t for the t -th data point using the closed-form expression provided in Equation (3.18). This update ensures that the constraints on the membership values such as non-negativity and summing to one, are satisfied. The complete iterative process for updating the entire membership matrix $\mathbf{U}$ across all data points is outlined in Algorithm 1.

Algorithm 1: Algorithm for Updating the rows of $\mathbf{U}$

Input: $\mathbf{g}_i$;

Output: $\mathbf{u}_i$;

1. Compute $\mathbf{m} = \mathbf{g} - \frac{\mathbf{1}\mathbf{1}^T \mathbf{g}}{K} + \frac{\mathbf{1}}{K}$;

2. Initialize $\bar{\delta} = 0$;

3. while the stopping criterion is not met do;

Update $\bar{\delta}$ using Newton's method;

end

4. for each $t \in [1, K]$ do

Update $u_t^* = (m_t - \bar{\delta}^*)_+$

end

To incorporate n -Pythagorean fuzzy logic into the clustering process, generate non-membership using n -. This hesitancy degree plays a crucial role by influencing the upper bound of the membership degree in each iteration. Consequently, the updated membership value is calculated by considering both the original membership u_{ij} and its corresponding hesitancy π_{ij} .

The n -Pythagorean fuzzy hesitancy degree π_{ij} for Yager class is

$$(3.20) \quad \pi_{ij}^n = 1 - u_{ij}^n - \left(1 - (u_{ij})^\alpha\right)^{\frac{n}{\alpha}}, \quad \text{where } 0 < \alpha \leq n.$$

Since we want to use the upper bound of membership value, we update u_{ij}

$$(3.21) \quad u_{ij}^* = (u_{ij}^n + \pi_{ij}^n)^{\frac{1}{n}}$$

Substituting the definition of π_{ij}^n from Equation (3.20) into Equation (3.21), the modified membership degree can be expressed as:

$$(3.22) \quad u_{ij}^* = \left(1 - \left(1 - (u_{ij})^\alpha\right)^{\frac{n}{\alpha}}\right)^{\frac{1}{n}}.$$

A dataset Ω containing N data points; number of clusters K ; stopping criterion $\eta > 0$; threshold value ϵ sparsity factor γ ; parameter n for n -Pythagorean fuzzy sets; hesitation control parameter α ; and the maximum number of iterations max-iter to prevent indefinite execution in case convergence is not achieved. The algorithm returns the membership matrix $\mathbf{U} \in \mathbb{R}^{N \times K}$ and the cluster centers $\mathbf{C}$. Implimentation of this algorithm is given in Algorithm 2.

Algorithm 2: RS-nPyFKC Algorithm

Input: $\Omega, K, \eta, \epsilon, \gamma, n, \alpha, \text{max-iter}$

Output: $\mathbf{U}, \mathbf{C}$

Initialize the membership matrix randomly $\mathbf{U}^0$ and which satisfies

$$\sum_{j=1}^c u_{ij} = 1 \quad \forall 1 \leq i \leq N;$$

Initialize the centers $\mathbf{C}$ randomly from Ω

iter = 0;

while $|u_{ij}^{\text{new}} - u_{ij}^{\text{pre}}| > \eta$ and iter $\leq$ max-iter do

Store $u_{ij}^{\text{pre}} = u_{ij}$;

Update $\mathbf{S}$ using Equation (3.4);

Update $\mathbf{C}$ using Equation (3.8);

Update $\mathbf{U}$ using Algorithm 1;

Update $\mathbf{U}$ to $\mathbf{U}^{\text{new}}$ using Equation (3.22);

iter = iter+1

end

4. Auto-Tuning Of Parameters

To ensure the optimal performance of the RS-nPyFKM algorithm, a comprehensive parameter tuning procedure was implemented. Given the algorithm's dependence on four specific parameters, Bayesian optimization was utilized to efficiently explore the hyperparameter space. The objective function for this optimization was defined by the four clustering evaluation metrics: Silhouette, Davies-Bouldin, Calinski-Harabasz, and Adjusted Rand Index. For each metric, values of $\eta = 10^{-8}$ and $m = 2$ was fixed. The optimization ran for 50 trials. The predefined ranges for parameter exploration were: $n \in 1,2,3,4,5$, $\gamma \in [0.1,20]$, $\epsilon \in [1,5]$ and $\alpha \in [0.3,n]$. The resulting optimal parameter settings that yielded the best performance for each clustering metric on the synthetic, Iris, and Wine datasets are in Table 1, Table 2 and Table 3, respectively.

Table 1: Parameter tuning for the best value of different matrices for the Synthetic dataset

Metric	n	α	γ	ϵ
Silhouette	2	0.7561	3.4271	2.0988
Davies Bouldin	3	2.3043	4.2183	2.1407
Calinski Harabasz	5	3.1573	17.2213	2.0458
Adjusted Rand Index	4	2.9727	7.3352	2.0714

Table 2: Parameter tuning for the best value of different matrices for the Iris dataset

Metric	n	α	γ	ϵ
Silhouette	2	0.3033	1.0667	4.4272
Davies Bouldin	3	1.264	3.7299	4.696
Calinski Harabasz	4	2.8096	0.8996	4.9872
Adjusted Rand Index	4	2.7475	4.037	4.4462

Table 3: Parameter tuning for the best value of different matrices for the Wine dataset

Metric	n	α	γ	ϵ
Silhouette	3	1.3776	6.0257	4.3855
Davies Bouldin	2	1.1731	12.5275	1.4016
Calinski Harabasz	4	3.1766	17.4428	4.3979
Adjusted Rand Index	2	0.8256	13.7041	3.8161

5. Comparative Analysis

To assess the competitive performance of the RS-nPyFKM algorithm, it was systematically compared against three prominent clustering algorithms: K-Means, Fuzzy C-Means (FCM), and Gaussian Mixture Models (GMM). The RS-nPyFKM algorithm was instantiated with its optimally tuned parameters obtained from the procedure outlined in the previous section. Performance evaluation was uniformly applied across all algorithms using the four previously specified clustering metrics (Silhouette, Davies-Bouldin, Calinski-Harabasz, and Adjusted Rand Index). To mitigate the effects of random initialization and dataset sampling, a bootstrapping strategy involving 100 resamples was employed for each experimental run. The comprehensive comparative results for the synthetic, Iris, and Wine datasets are presented in Table 4, Table 5 and Table 6, respectively.

Table 4: Clustering performances on Synthetic Dataset (mean±std)

Metric	KMeans	FCM	GMM	RSnPyFKM
Silhouette	0.8191±0.0066	0.8191±0.0066	0.8183 ±0.0074	0.8417 ±0.1460
Davies Bouldin	0.2510 ±0.0106	0.2510±0.0106	0.2525 ±0.0121	0.2304 ±0.0175
Calinski Harabasz	4756.2664 ±505.7099	4756.2664 ±505.7099	4703.1724 ±546.9138	4671.5076±406.9558
Adjusted Rand Index	0.9414±0.0118	0.9414±0.0118	0.9415±0.0117	0.9572 ±0.0224

Table 5: Clustering performances on Iris Dataset (mean±std)

Metric	KMeans	FCM	GMM	RSnPyFKM
Silhouette	0.4821 ±0.0334	0.4710 ±0.0255	0.4297 ±0.0554	0.5201 ±0.0345
Davies Bouldin	0.7880 ±0.0784	0.8227 ±0.0394	0.9318 ±0.1571	0.8053 ±0.5738
Calinski Harabasz	242.8881 ±39.0586	252.6790 ±29.8910	207.8654 ±39.0771	264.2255 ±48.0562
Adjusted Rand Index	0.6071 ±0.0859	0.6451 ±0.0628	0.7063 ±0.1730	0.6869 ±0.1485

Table 6: Clustering performances on Wine Dataset (mean±std)

Metric	KMeans	FCM	GMM	RSnPyFKM
Silhouette	0.2969 ±	0.2994 ±	0.2954 ±	0.5190 ±

	0.0204	0.0163	0.0210	0.0324
Davies Bouldin	1.3594 $\pm$ 0.0628	1.3662 $\pm$ 0.0621	1.3665 $\pm$ 0.0650	0.7453 $\pm$ 0.0838
Calinski Harabasz	73.8849 $\pm$ 8.8380	75.1860 $\pm$ 6.3377	73.4840 $\pm$ 8.8439	123.7503 $\pm$ 8.1235
Adjusted Rand Index	0.8236 $\pm$ 0.1023	0.8575 $\pm$ 0.0574	0.8164 $\pm$ 0.1098	0.9126 $\pm$ 0.1199

6. Conclusion

This study concludes with the introduction of the Robust Sparse n-Pythagorean Fuzzy K-Means (RS-nPyFKM) algorithm, a novel approach designed to advance fuzzy clustering capabilities. Our findings indicate that RS-nPyFKM produces more accurate results compared to classical clustering algorithms and demonstrates significant resilience against outlier data points. Despite these encouraging results, the current auto-tuning procedure for RS-nPyFKM is computationally demanding, primarily due to the number of inherent parameters. Consequently, future research will prioritize exploring strategies to enhance the algorithm's convergence efficiency and expand its evaluation to include a greater diversity of datasets.

References

1. J. C. Dunn; A Fuzzy Relative of the ISODATA Process and Its Use in Detecting Compact, Well-Separated Clusters, *Journal of Cybernetics*, 3 (1973), 32-57.
2. J. C. Bezdek; *Pattern Recognition with Fuzzy Objective Function Algorithms*, Springer US, 1981.
3. C. Qiu, D. Zhang, Z. Zhao; Sparse Fuzzy C-Means Clustering, *IEEE Transactions on Fuzzy Systems*, 20 (2012), 1109-1121.
4. D. M. Witten, R. Tibshirani; A Framework for Feature Selection in Clustering, *Journal of the American Statistical Association*, 105 (2010), 713-726.
5. J. Xu, J. Han, K. Xiong, F. Nie; Robust and Sparse Fuzzy K-Means Clustering, In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, (2016), 2224-2230.
6. N. Billor, S. Chatterjee, A.S. Hadi; A Re-Weighted Least Squares Method for Robust Regression Estimation, *American Journal of Mathematical and Management Sciences*, 26 (2006), 229-252.
7. D. Q. Zhang, S. C. Chen; Fuzzy Clustering Using Kernel Methods, *Soft Computing*, 7 (2003), 222-229.

8. Z. Ji, J. Xu, R. Li; Fuzzy K-Means Clustering with Local Spatial Weights for Image Segmentation, *Pattern Recognition Letters*, 33 (2012), 1546-1552.
9. F. Zhao, C. Li, X. Ma, H. Shi; Fuzzy K-Means Clustering with Non-Local Spatial Constraints for Image Segmentation, *IEEE Transactions on Fuzzy Systems*, 21 (2013), 1058-1070.
10. Z. Ji, Z. Deng, J. Xu; Fuzzy K-Means Clustering with Embedded Spatial Information for Image Segmentation, *Pattern Recognition Letters*, 45 (2014), 1-7.
11. S. R. Kannan, R. Devi, S. Ramathilagam, T. P. Hong; Effective Fuzzy C-Means Clustering with Normed Kernel Function and Center Initialization, *International Journal of Innovative Computing, Information and Control*, 8 (2012), 4873-4888.
12. X. Wang, X. Song, Y. Zhang; Adaptive Spatial Information-Theoretic Fuzzy Clustering for Image Segmentation, *IEEE Transactions on Fuzzy Systems*, 21 (2013), 1046-1057.
13. T. Li, Y. Chen, Y. Zhang; Regularized Fuzzy C-Means Clustering and Its Application to Gene Expression Data, *Bioinformatics*, 24 (2008), 1445-1451.
14. H. Namkoong, F. Bach; Regularization-Based Fuzzy Clustering, *Journal of Machine Learning Research*, 11 (2010), 2235-2267.
15. R. P. Li, M. Mukaidono; A Maximum Entropy Approach to Fuzzy Clustering, In *Proceedings of the Fourth IEEE International Conference on Fuzzy Systems*, 4 (1995), 2227-2232.
16. S. Miyamoto, K. Umayahara; Fuzzy Clustering by Quadratic Regularization, In *Proceedings of the 1998 IEEE International Conference on Fuzzy Systems*, 2 (1998), 1394-1399.
17. D. Ozdemir, L. Akarun; Fuzzy Algorithms for Combined Quantization and Dithering, *IEEE Transactions on Image Processing*, 11 (2002), 260-270.
18. J. Yu, T. Yang; Regularized Fuzzy C-Means Clustering with Spatial Information for Image Segmentation, *Pattern Recognition Letters*, 28 (2007), 1707-1714.
19. K. T. Atanassov; *Intuitionistic Fuzzy Sets: Theory and Applications*, Physica-Verlag HD, Heidelberg, 1999, 1-137.
20. Z. Xu; Intuitionistic Fuzzy C-Means Algorithm, *Journal of Computer Applications*, 29 (2009), 3465-3474.
21. A. Bryniarska; n-Pythagorean Fuzzy Sets. *Fuzzy Sets and Systems*, 335 (2018), 1-13.
22. Xian, Sidong, Yue Cheng; Pythagorean fuzzy time series model based on Pythagorean fuzzy c-means and improved Markov weighted in the prediction of the new COVID-19 cases, *Soft Computing* 25.22 (2021): 13881-13896.