Knowledge-Based Systems 326 (2025) 113990

Contents lists available at ScienceDirect

Knowledge-Based Systems

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

Semi-supervised equilibrium K-means for imbalanced data clustering

Yudong He

Department of Industry Engineering and Decision Analytics, The Hong Kong University of Science and Technology, Hong Kong

ARTICLE INFO

Keywords:
Fuzzy clustering
Equilibrium K-means
Semi-supervised learning
Imbalanced clustering

ABSTRACT

Equilibrium K-means (EKM) represents a novel advancement in fuzzy clustering methodologies, outperforming Bezdek's fuzzy C-means (FCM) algorithm when applied to datasets with imbalanced distributions. Nonetheless, EKM is inherently limited by its inability to integrate supervision information, rendering it less effective in scenarios wherein partial data labels are accessible. In this paper, we propose a semi-supervised variant of EKM (SSEKM) that can effectively leverage supervision knowledge. The effects of supervision knowledge on the model and convergence behavior are elucidated via theoretical analysis. Empirical evaluations conducted on four synthetic and 16 real-world datasets from medical, biological, and industrial sectors indicate that SSEKM exhibits competitive performance against semi-supervised FCM (SSFCM) and other state-of-the-art semi-supervised fuzzy clustering algorithms on balanced datasets and surpasses them on imbalanced datasets. Additionally, SSEKM maintains a computational complexity comparable to SSFCM and offers a higher convergence speed than most comparative algorithms.

1. Introduction

1.1. Semi-supervised Fuzzy clustering

The objective of fuzzy clustering is to allocate data objects to multiple clusters on the basis of their similarity, often quantified through metrics such as Euclidean distance. It assigns a membership value for each object, indicating the degree of association between the object and a cluster. This methodology is extensively applied in fields such as image segmentation, pattern recognition, and marketing. Nevertheless, traditional fuzzy clustering techniques are unsupervised, rendering them suboptimal when *partial supervision* is available, that is, when some data labels are known. In recent years, numerous semi-supervised fuzzy clustering (SSFC) methods have been devised to effectively integrate the information provided by partial supervision. Given that even a small degree of human intervention (e.g., 5%–10% labeled data) can substantially enhance clustering outcomes, SSFC has garnered significant interest within the machine-learning community. State-of-the-art (SOTA) SSFC algorithms have been comprehensively reviewed in the literature [1,2].

Numerous SSFC algorithms have been developed on the basis of Bezdek's fuzzy C-means (FCM) algorithm [3], with a focus on managing the influence of partial supervision on the outcomes of unsupervised models. Pedrycz and Waletzky [4] proposed a semi-supervised FCM (SSFCM) framework that additively combines unsupervised and supervised objectives. Subsequent research [5–8] has extended Pedrycz's SSFCM by refining its approach to utilizing partial supervision at different levels. Recently, Kmita et al. [9] proposed an innovative version of SSFCM that elucidates the impact of partial supervision. Additionally, alternative models to FCM have been adapted for semi-supervised frameworks [10–12], including possibilistic C-means (PCM) [13] and possibilistic FCM (PFCM) [14]. The primary distinction between FCM and PCM lies in the implementation and interpretation of the soft assignment of data points; specifically, membership in PCM is redefined as a measure of typicality. PFCM represents a hybridization of FCM and PCM.

1.2. Uniform effect: The drawback of FCM

Most fuzzy clustering algorithms inherently tend to form clusters of similar sizes [15]. This phenomenon is referred to as the uniform effect [16]. It substantially undermines the performance of these algorithms when applied to datasets characterized by class-size disparities, commonly known as imbalanced data. Such imbalanced datasets are prevalent in various domains, such as medical diagnosis, fraud detection, and rare event prediction. The uniform effect markedly hinders the effective application of fuzzy clustering algorithms in these sectors.

The uniform effect arises owing to several intrinsic factors. The fundamental principle of fuzzy clustering is to identify cluster prototypes, namely centroids, and to partition the data based on the distance between the data points and centroids. All aforementioned fuzzy

E-mail address: yhebh@connect.ust.hk.

https://doi.org/10.1016/j.knosys.2025.113990
Received 25 February 2025; Received in revised form 28 May 2025; Accepted 18 June 2025
Available online 5 July 2025

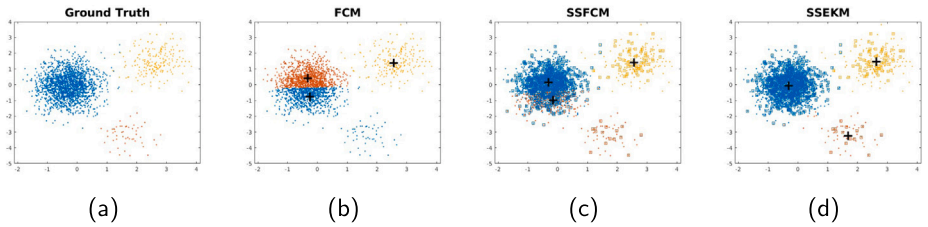

Fig. 1. Clustering results of a highly imbalanced dataset. (a) Ground truth. The colors represent the reference labels. (b) Clustering by Bezdek's FCM [3]. (c) Clustering by a SOTA SSFCM [9]. (d) Clustering by the proposed SSEKM. Supervision points are marked with square boxes.

clustering algorithms, unsupervised and semi-supervised, employ a similar formulation for centroid calculation: centroids are computed as weighted means of data feature vectors, with the weights corresponding to the value or exponent of the membership. This centroid computation method becomes problematic in scenarios involving highly imbalanced datasets. As membership values are non-negative and larger classes contain more data points, the cumulative weight from the larger classes dominates the centroid updates. This systematic bias draws all centroids toward larger classes, resulting in clusters of equal size regardless of the true data distribution. Fig. 1 illustrates the uniform effect. Fig. 1(b) and Fig. 1(c) depict the clustering results of Bezdek's FCM [3] and a SOTA version of SSFCM [9], respectively. In these illustrations, the square boxes denote the data objects with known class labels, referred to as *supervision points*. These points are obtained by randomly selecting 30% of data objects from each class and treating them as labeled data. As evident from the figures, the centroids of both FCM and SSFCM deviate from the smaller cluster and converge within the larger cluster.

1.3. Existing efforts to overcome uniform effect

Research addressing the mitigation of the uniform effect in fuzzy clustering is sparse, primarily because of the limited knowledge of the true distribution of data under unsupervised and semi-supervised learning. This subsection briefly reviews the existing efforts aimed at overcoming this challenge. Some researchers [17–19] have proposed calculating the centroids of supervised data, referred to as supervision centroids, and subsequently adjusting the final centroids to align more closely with the supervision centroids. However, a notable limitation of this approach concerns the necessity of carefully adjusting weights assigned to supervision points when computing supervision centroids. If not meticulously calibrated, even low weights assigned for supervision points distant from the class centroid can introduce errors, which would not occur otherwise, in the final partitions [17]. This sensitivity undermines the reliability of the method, particularly when the quality of partial supervision is uncertain.

Other researchers [20–22] have attempted to counter the uniform effect by weighting membership on the basis of the relative size of the cluster to which a data point belongs. However, this method lacks robustness and is susceptible to noise and outliers [22]. Alternatively, researchers [23,24] have proposed generating a greater number of clusters, termed sub-clusters, than initially required, and then merging these sub-clusters to form the final clusters. While effective for imbalanced data, this method is complex and necessitates multiple stages. More importantly, these algorithms [20–24] are purely unsupervised and do not incorporate the insights provided by partial supervision. Recently, a novel fuzzy clustering method known as equilibrium K-means (EKM) has been proposed [25]. EKM modifies the constraints in FCM and allows data objects to exert a repulsive force on centroids, thereby achieving excellent performance on imbalanced data. However, similar to other methods, EKM does not leverage partial supervision knowledge.

1.4. Research gap

Despite advancements in SSFC, the challenge of effectively clustering imbalanced data persists. The uniform effect necessitates the development of more robust solutions. Current methodologies for addressing imbalanced data clustering often do not fully exploit partial supervision, thereby missing opportunities to enhance clustering accuracy. By bridging this gap, the efficacy and applicability of SSFC techniques can be remarkably enhanced.

1.5. Contribution

This study aimed to address the aforementioned research gap by integrating supervision knowledge into EKM, thereby enhancing its clustering performance on imbalanced datasets when partial supervision is available. Specifically, the objective function of EKM is first modified to incorporate the influence of known data labels. Our analysis explicitly demonstrates how partial supervision affects centroid formation. Second, we derive a convergence condition for the proposed method, referred to as semi-supervised equilibrium K-means (SSEKM). This condition elucidates the relationship among the convergence speed, quantity of labeled data, and parameter controlling the impact of partial supervision. Generally, an increase in supervised data correlates with faster convergence, whereas the impact of the parameter is more complex. Finally, the performance of SSEKM is validated across 20 datasets, benchmarking it against numerous SOTA SSFC algorithms. The experimental results underscore the efficiency of the proposed SSEKM on imbalanced datasets. One example is presented in Fig. 1(d). The SSEKM framework is illustrated in Fig. 2. In contrast to FCM and SSFCM, SSEKM perfectly accomplishes an imbalanced data clustering. SSEKM outperforms EKM and is more robust to parameter variations. In terms of computational complexity, SSEKM is theoretically and practically comparable to SSFCM, and it is faster than many SOTA SSFC algorithms.

1.6. Organization

The rest of the paper is organized as follows. In Section 2, we introduce preliminaries about FCM and EKM, highlighting their similarity and difference. We formulate the proposed SSEKM in Section 3 along with theoretical analysis. Experimental results and the related discussion are presented in Section 4. Finally, we conclude in Section 5.

2. Preliminaries

2.1. Fuzzy C-means

The similarity between two points is defined using the Euclidean distance $d_{kn} = \|\mathbf{x}_n - \mathbf{c}_k\|_2$, where $\mathbf{x}_n$ is the nth data point and $\mathbf{c}_k$ is the

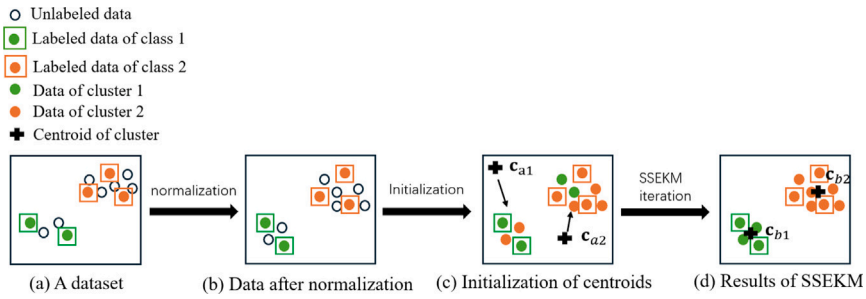

Fig. 2. An illustration of the SSEKM framework for data clustering. (a) A dataset with partially human-labeled data. (b) Data are normalized with zero mean and unit variance for each feature. (c) Cluster centroids ($\mathbf{c}_{a1}$ and $\mathbf{c}_{a2}$) are initially selected by K-means++, and each data point is allocated to the cluster where the nearest cluster centroid is located. (d) Cluster centroids are iteratively updated under the guidance of partial supervision knowledge, until the stopping criteria are met and the final centroids ($\mathbf{c}_{b1}$ and $\mathbf{c}_{b2}$) are obtained.

centroid of the kth cluster. FCM calculates the centroids by solving the following optimization problem

$$\begin{aligned} \min_{\mathbf{c}_1,\ldots,\mathbf{c}_K,[u_{kn}]} \quad & \sum_{n=1}^{N}\sum_{k=1}^{K}(u_{kn})^m d_{kn}^2 \\ \text{subject to} \quad & u_{kn} \in [0,\,1]\,\forall k,n, \\ & \sum_{k=1}^{K} u_{kn} = 1\,\forall n, \\ & 0 < \sum_{n=1}^{N} u_{kn} < N\,\forall k, \end{aligned} \tag{1}$$

where u_{kn} is the membership value of the nth data point belonging to the kth cluster, K is the number of clusters, N is the number of data points, and $m \in (1,+\infty)$ is a parameter controlling the fuzziness level. Applying the Lagrange multiplier method, an iterative algorithm can be derived from the necessary conditions for finding the optimal solution, as given below [26]:

1. Calculate the membership value of the nth data point belonging to the kth cluster:

$$u_{kn}^{(\tau)} = \frac{1}{\sum_{i=1}^{K}\left(\frac{d_{kn}^{(\tau)}}{d_{in}^{(\tau)}}\right)^{\frac{2}{m-1}}}. \tag{2}$$

2. Recalculate the weighted centroid of the kth cluster by:

$$\mathbf{c}_k^{(\tau+1)} = \frac{\sum_n (u_{kn}^{(\tau)})^m \mathbf{x}_n}{\sum_n (u_{kn}^{(\tau)})^m}, \tag{3}$$

where τ denotes the number of iterations and $d_{kn}^{(\tau)} = \|\mathbf{x}_n - \mathbf{c}_k^{(\tau)}\|_2$. The time complexity of one iteration is $O(NK^2)$. Linear convergence of FCM can be guaranteed with arbitrary initial centroids [27].

2.2. Equilibrium K-means

Alternatively, EKM finds centroids by solving the following optimization problem

$$\begin{aligned} \min_{\mathbf{c}_1,\ldots,\mathbf{c}_K} \quad & \sum_{n=1}^{N}\sum_{k=1}^{K} u_{kn} d_{kn}^2 \\ \text{subject to} \quad & u_{kn} = \frac{\exp(-\alpha d_{kn}^2)}{\sum_{k=1}^{K}\exp(-\alpha d_{kn}^2)}\,\forall k,n, \end{aligned} \tag{4}$$

where $\sum_{k=1}^{K} u_{kn} = 1$ and $\alpha \in (0,+\infty)$ is a suitable positive number. The optimization algorithm is based on a quasi-Newton method, given by

1. Calculate the individual contribution of the nth data point to the kth cluster:

$$w_{kn}^{(\tau)} = u_{kn}^{(\tau)}\left[1 - \alpha\left((d_{kn}^{(\tau)})^2 - \sum_{i=1}^{K} u_{in}(d_{in}^{(\tau)})^2\right)\right], \tag{5}$$

2. Recalculate the weighted centroid of the kth cluster by:

$$\mathbf{c}_k^{(\tau+1)} = \frac{\sum_{n=1}^{N} w_{kn}^{(\tau)} \mathbf{x}_n}{\sum_{n=1}^{N} w_{kn}^{(\tau)}}. \tag{6}$$

The time complexity of one iteration is the same as that of FCM, which is $O(NK^2)$. Linear convergence can be guaranteed with arbitrary initial centroids at a sufficiently small α (kindly refer to Section 3.4). Practically, α can be determined by statistical measures, such as variance [25].

2.3. Comparison between FCM and EKM

Both FCM (3) and EKM (6) calculate the centroid as the weighted mean of data points. The distinction lies in the individual contribution, which is defined as u_{kn}^m in FCM and w_{kn} in EKM. Given that $u_{kn}^m \geq 0, \forall k,n$, and $m > 1$, the centroids computed by FCM tend to gravitate towards larger clusters. Consequently, FCM exhibits a uniform effect when applied to imbalanced data, leading to suboptimal clustering outcomes. Most fuzzy clustering algorithms (including both unsupervised and semi-supervised approaches) employ a similar definition of individual contribution as in FCM, which limits their efficacy on imbalanced datasets. By contrast, w_{kn} can assume either negative or positive values. If the kth centroid is nearest to the nth data point, the centroid is attracted; if it is farthest, it is repelled. As a result, centroids computed by EKM are not concentrated in large clusters but are repelled by them instead. Ultimately, these centroids are accommodated by smaller clusters.

Nevertheless, one limitation of EKM is the necessity of precise tuning α to achieve optimal clustering performance. Partial supervision offers an opportunity to adjust individual contributions in EKM when the value of α is suboptimal.

3. Semi-supervised equilibrium K-means

3.1. Objective function

Typically, partial supervision knowledge is represented as a binary matrix $\mathbf{F} = [f_{kn}]$ (called the prior matrix), which has the same dimension as that of the membership matrix. If the nth data point is known to belong to the kth cluster, then $f_{kn} = 1$; otherwise, $f_{kn} = 0$. If a column in $\mathbf{F}$ contains all zero entries, it indicates that the label of the corresponding data point is unknown and is therefore classified as unsupervised. Conversely, a non-zero entry signifies that the data point is supervised. For convenience, we define an auxiliary variable b_n; $b_n = 1$ if the nth data point is supervised (implying $\sum_{k=1}^{K} f_{kn} = 1$) and $b_n = 0$ if unsupervised (implying $\sum_{k=1}^{K} f_{kn} = 0$). Notably, $\mathbf{F}$ need not be strictly binary. The choice of $\mathbf{F}$ hold nuanced implications, which will be discussed in Section 3.3.

The proposed objective function for SSEKM is formulated as follows:

$$J_{\text{SSEKM}} = \sum_{n=1}^{N}\sum_{k=1}^{K} u_{kn} d_{kn}^2 + \theta \sum_{n=1}^{N} b_n \sum_{k=1}^{K} (f_{kn} - u_{kn}) d_{kn}^2. \tag{7}$$

where the parameter $\theta \in [0, +\infty)$ controls the trade-off between unsupervised and supervised components. Larger θ values prioritize supervision, forcing centroids to align with the labeled data. By contrast, smaller θ values emphasize the intrinsic data structure. This objective function is an additive combination of two terms where the first term is purely unsupervised and the second term is supervised. One advantage of this objective function is that the effect of partial supervision on the model and convergence is explainable. Kindly refer to Section 3.3 and Section 3.4 for the explanation. The minimization problem for SSEKM is defined as:

$$\begin{aligned} &\min_{\mathbf{c}_1,\ldots,\mathbf{c}_K} \quad J_{\text{SSEKM}}(\mathbf{c}_1, \ldots, \mathbf{c}_K) \\ &\text{subject to} \quad u_{kn} = \frac{\exp(-\alpha d_{kn}^2)}{\sum_{k=1}^{K} \exp(-\alpha d_{kn}^2)}, \forall k, n. \end{aligned} \tag{8}$$

When $\theta = 0$ or $\forall n, b_n = 0$ (i.e., no supervision points), the optimization problem (8) reduces to (4), rendering the SSEKM equivalent to EKM.

3.2. Optimization

We solve the above optimization problem using a quasi-Newton method. The objective function J_{SSEKM} contains the first-order partial derivative of

$$\nabla_{\mathbf{c}_k} J_{\text{SSEKM}} = -2 \sum_{n=1}^{N} \tilde{w}_{kn} (\mathbf{x}_n - \mathbf{c}_k), \tag{9}$$

where

$$\tilde{w}_{kn} = w_{kn} + \theta b_n (f_{kn} - w_{kn}) \forall k, n, \tag{10}$$

and w_{kn} is the weight as defined in (5). The Hessian of J_{SSEKM} is given by

$$\frac{\partial^2 J_{\text{SSEKM}}}{\partial \mathbf{c}_k^2} = 2 \sum_{n=1}^{N} \left(\tilde{w}_{kn} \mathbf{I} - 2\alpha(1 - \theta b_n) u_{kn} \sum_{i \neq k} w_{in} \mathbf{D}_{kn} \right), \tag{11}$$

where $\mathbf{I}$ is the identity matrix and $\mathbf{D}_{kn} = (\mathbf{x}_n - \mathbf{c}_k)(\mathbf{x}_n - \mathbf{c}_k)^T$ is a rank-1 matrix. Notice that $u_{kn} \sum_{i \neq k} w_{in} \propto u_{kn} \sum_{i \neq k} u_{in} = u_{kn}(1 - u_{kn})$, which is smaller than u_{kn} especially when $u_{kn} \to 1$ (which means the data has well-separated structure). Hence, we can approximate the Hessian by the first term: $\frac{\partial^2 J_{\text{SSEKM}}}{\partial \mathbf{c}_k^2} \approx 2 \sum_{n=1}^{N} \tilde{w}_{kn} \mathbf{I}$. According to the Newton method, we have the following optimization process

$$\mathbf{c}_k^{(\tau+1)} = \mathbf{c}_k^{(\tau)} + \sum_{n=1}^{N} \frac{\tilde{w}_{kn}^{(\tau)}}{\sum_{n=1}^{N} \tilde{w}_{kn}^{(\tau)}} (\mathbf{x_n} - \mathbf{c_k^{(\tau)}}). \tag{12}$$

Or we can rewrite it into a more concise form of the mean of data points, which is

$$\mathbf{c}_k^{(\tau+1)} = \frac{\sum_{n=1}^{N} \tilde{w}_{kn}^{(\tau)} \mathbf{x}_n}{\sum_{n=1}^{N} \tilde{w}_{kn}^{(\tau)}}. \tag{13}$$

The convergence condition for the iterative process is given in Section 3.4. For clarity, the optimization procedure of SSEKM is summarized in Algorithm 1. In each iteration, updating weights and centroids requires $O(NK^2)$ and $O(NK)$ operations, respectively. Hence, the time complexity of SSEKM for a single iteration is $O(NK^2)$, the same as that of EKM. In practice, SSEKM often demonstrates a higher convergence rate than EKM, as it requires fewer iterations. This is attributed to the convergence condition of SSEKM, as elucidated in Section 3.4.

Algorithm 1: Semi-Supervised Equilibrium K-Means

Input: A dataset $\mathbf{X} = \{\mathbf{x}_n\}_{n=1}^{N}$, initial centroids $\{\mathbf{c}_k^{(0)}\}_{k=1}^{K}$, prior matrix $\mathbf{F} = [f_{kn}]$, parameter values of α and θ.
Output: Centroids $\{\mathbf{c}_1, \cdots, \mathbf{c}_K\}$ and membership matrix $\mathbf{U} = [u_{kn}]$

$\tau = 0$;
repeat
  Compute weight $\tilde{w}_{kn}^{(\tau)}$ by (5) and (10) for all k, n;
  Update centroid $\mathbf{c}_k^{(\tau+1)}$ by (13) for all k;
  $\tau = \tau + 1$;
until *convergence*;
Calculate membership matrix $\mathbf{U} = [u_{kn}]$ according to its constraint in (8);
return $\{\mathbf{c}_k^{(\tau)}\}_{k=1}^{K}$ *and* $\mathbf{U}$

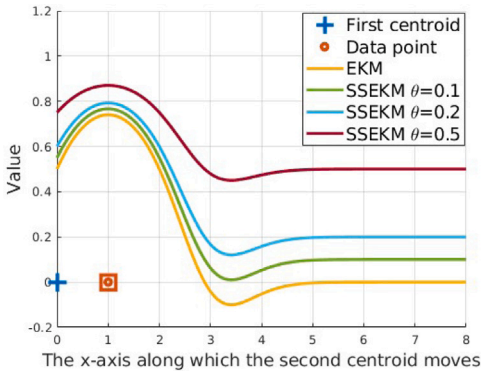

Fig. 3. Individual contribution of a data point to the second centroid, i.e., w_{2n} for EKM and $\tilde{w}_{2n}$ for SSEKM. The first centroid and the nth data point are fixed at $x = 0$ and $x = 1$, respectively. The curves are functions of the position of the second centroid. The data point is a supervision point known to belong to the second cluster, i.e., $f_{2n} = 1$.

3.3. Impact of partial supervision on model

The individual contribution $\tilde{w}_{kn}$ is expanded for two types of data points:

$$\tilde{w}_{kn} = \begin{cases} w_{kn}, & \text{if } b_n = 0, \\ w_{kn} + \theta(f_{kn} - w_{kn}), & \text{if } b_n = 1 \end{cases} \tag{14}$$

As evident from the equations, SSEKM processes unsupervised data similar to that of EKM. For supervised data (i.e., $b_n = 1$), the learning is adjusted through the term $\theta(f_{kn} - w_{kn})$. If $f_{kn} > w_{kn}$, then $\tilde{w}_{kn} > w_{kn}$ because $\theta > 0$. This implies that partial supervision enhances the alignment between the labeled data and centroids. If $f_{kn} < w_{kn}$, then $\tilde{w}_{kn} < w_{kn}$, mitigating the harmful influence of the mismatch between class centroids and cluster centroids obtained by the algorithm. This correction is optimal when the value of f_{kn} equals an ideal individual contribution that generates optimal partitions and centroids. Therefore, it is beneficial to investigate the construction of $\mathbf{F}$. However, this is out of the scope of this study. In this paper, $\mathbf{F}$ is set to be a binary matrix following the conventions in literature.

To illustrate the positive effect of the impact, consider a simple example on the x-axis. Fix the first centroid at $x = 0$ and the nth data point at $x = 1$. Let the nth data point be known to belong to the second cluster. Fig. 3 shows the plots of w_{2n} and $\tilde{w}_{2n}$ as functions of the position of the second-cluster centroid. As observed, w_{2n} is negative in the range $x \in [3, 5]$ and nearly zero for $x > 5$. This is undesired as the nth data point pushes the second-cluster centroid further. By contrast, $\tilde{w}_{2n}$ remains positive even for $x >= 3$, indicating that the nth data point draws the second-cluster centroid closer to it.

3.4. Convergence analysis

We derive a convergence condition of SSEKM based on the fixed-point theorem, which gives fruitful insights regarding the impact of parameters and supervision knowledge on the convergence of SSEKM.

Theorem 1 (*Convergence Condition*). *The centroid sequence obtained by (13) converges linearly to a stationary point* $\{\mathbf{c}_1^*, \dots, \mathbf{c}_K^*\}$ *of the objective function* J_{SSEKM} *(7) if a constant* $0 < \upsilon < 1$ *exists with*

$$M_k(\alpha; \{\mathbf{c}_1, \dots, \mathbf{c}_K\}) \le \upsilon, \quad \forall k, \{\mathbf{c}_1, \dots, \mathbf{c}_K\} \in \mathbb{S},$$

where

$$\mathbb{S} := \{\{\mathbf{c}_1, \dots, \mathbf{c}_K\} : \|\mathbf{c}_k - \mathbf{c}_k^*\|_2 \le \|\mathbf{c}_k^{(0)} - \mathbf{c}_k^*\|_2 \, \forall k\},$$

and

$$M_k := |\sum_{n=1}^{N} \tilde{w}_{kn}|^{-1} \sum_{n=1}^{N} |1 - \theta b_n| 2\alpha u_{kn} |1 - w_{kn}| d_{kn} \cdot \|\mathbf{x}_n - \frac{\sum_n \tilde{w}_{kn}\mathbf{x}_n}{\sum_n \tilde{w}_{kn}}\|_2.$$

Proof. Define $f(\mathbf{c}_k) = (\sum_{n=1}^{N} \tilde{w}_{kn}\mathbf{x}_n)/(\sum_{n=1}^{N} \tilde{w}_{kn})$. Observe the gradient of J_{SSEKM} given in (9), it is clear that any stationary point of J_{SSEKM} is a fixed point of f and vice versa. Hence, we have

$$\begin{aligned}\|\mathbf{c}_k^{(\tau)} - \mathbf{c}_k^*\|_2 &= \|f(\mathbf{c}_k^{(\tau-1)}) - \mathbf{c}_k^*\|_2 \\ &\le \|f'(\xi_k)\|_F \|\mathbf{c}_k^{(\tau-1)} - \mathbf{c}_k^*\|_2,\end{aligned}$$

where f' denotes the first derivative of f, ξ_k is a point on the line with $\mathbf{c}_k^{(\tau-1)}$ and $\mathbf{c}_k^*$ as endpoints, and $\|\cdot\|_F$ denotes the Frobenius norm. The inequality holds according to the median value theorem. The function f has the first derivative of

$$f'(\mathbf{c}_k) = (\sum_{n=1}^{N} \tilde{w}_{kn})^{-1} \sum_{n=1}^{N} (1 - \theta b_n) 2\alpha u_{kn} (1 - w_{kn}) \cdot (\mathbf{x}_n - \mathbf{c}_k)(\mathbf{x}_n^\top - \frac{\sum_{n=1}^{N} \tilde{w}_{kn}\mathbf{x}_n^\top}{\sum_{n=1}^{N} \tilde{w}_{kn}})$$

Note that $\|f'(\mathbf{c}_k)\|_F \le M_k \, \forall \mathbf{c}_k$ by triangular inequality and $\{\xi_1, \dots, \xi_K\} \in \mathbb{S}$, we have

$$\begin{aligned}\|\mathbf{c}_k^{(\tau)} - \mathbf{c}_k^*\|_2 &\le \upsilon \|\mathbf{c}_k^{(\tau-1)} - \mathbf{c}_k^*\|_2 \le \dots \\ &\le \upsilon^\tau \|\mathbf{c}_k^{(0)} - \mathbf{c}_k^*\|_2 \, \forall k = 1, \dots, K.\end{aligned}$$

Since $\upsilon < 1$,

$$\lim_{\tau \to \infty} \|\mathbf{c}_k^{(\tau)} - \mathbf{c}_k^*\|_2 \le \lim_{\tau \to \infty} \upsilon^\tau \|\mathbf{c}_k^{(0)} - \mathbf{c}_k^*\|_2 = 0 \, \forall k \quad \blacksquare$$

Here, we prove that when α is sufficiently small, the convergence condition given in Theorem 1 can be satisfied with arbitrary initial centroids. As $\alpha \to 0$, $u_{kn} \to 1/K$, $w_{kn} \to 1/K$, and $\tilde{w}_{kn} \to 1/K$ if $b_n = 0$ or $\tilde{w}_{kn} \to 1/K + \theta(f_{kn} - 1/K)$ if $b_n = 1$. Consequently, $M_k \to 0$, and the convergence is achieved in a single iteration. As M_k is a continuous function of α, the convergence condition of SSEKM can be satisfied when α is sufficiently small.

Next, we discuss the benefit of partial supervision for faster convergence, providing a theoretical justification for the results observed in the subsequent experiments. The term M_k is the sum of N non-negative terms, each multiplied by a factor of $|1 - \theta b_n|$. Therefore, when $0 < \theta < 2$, terms with $b_n = 1$ become smaller than their values in the unsupervised scenario, resulting in a reduction of M_k. Consequently, M_k decreases with increasing number of supervision points. As the value of M_k sets the lower bound of the distance traveled by centroids toward stationary points in a single iteration, a smaller M_k indicates a higher convergence rate. In other words, partial supervision accelerates the convergence, and the speed of convergence increases as the number of supervision points increases. When all observations are supervised and $\theta = 1$, SSEKM can converge in a single step because $M_k = 0$. Although M_k increases when $\theta > 2$, it does not necessarily imply a lower convergence speed for SSEKM. A tighter lower bound of convergence speed is given by $\|f'(\mathbf{c}_k)\|_2$, which may decrease (indicating faster convergence) when $\theta > 2$ because of the cancellation of positive and negative terms in the sum. Experimental evidences demonstrate that SSEKM converges faster than EKM when $\theta > 2$.

3.5. The choice of parameters

SSEKM contains three tunable parameters: α, θ, and $\{f_{kn}\}_{k,n}$. The parameter α is utilized to scale the distance, and it is recommended that it should be inversely proportional to the data variance [25]. Such a choice ensures that the convergence behavior and the trained model are more independent of similarity measures.

The parameter θ is used to balance the supervised and unsupervised components within the objective function. In [4], the authors suggest that its value be proportional to the ratio N/S, where N is the total number of data points and S is the number of supervision points. This ensures an equal weighting for the two components. We endorsed this choice and experimentally determined that setting $\theta = N/S$ yielded optimal clustering results in most scenarios.

The parameter set $\{f_{kn}\}_{k,n}$ is intended to represent the ideal value of the individual contribution for supervision points. Conventionally, $f_{kn} = 1$ if the nth data point is known to belong to the kth cluster, and $f_{kn} = 0$ otherwise. We follow this binary definition for a fair comparison in our experiment. Although this definition may not be optimal for SSEKM, it remains effective and the most practical choice.

4. Numerical experiments

4.1. Experimental setup

Numerical experiments were conducted to compare the performance of our proposed SSEKM algorithm with one hard and 11 fuzzy clustering algorithms, namely 1. HKM [28] (Lloyd, 1982) 2. FCM [3] (Bezdek et al. 1984) 3. MEFC [29] (Karayiannis, 1994) 4. PFCM [14] (Pal et al. 2005) 5. EKM [25] (He, 2024) 6. eSFCM [7] (Yasunori et al. 2009) 7. $\text{SSFCM}_{\text{p97}}$ [4] (Pedrycz and Waletzky, 1997) 8. $\text{SSFCM}_{\text{k24}}$ [9] (Kmita et al. 2024) 9. SFC-ER [8] (Salehi et al. 2021) 10. SSPCM [10] (Liu and Wu, 2013) 11. SPFCM [11] (Antoine et al. 2018) 12. FWSSPCM [19] (Yu et al. 2024). The first five comparative algorithms are unsupervised, and the other seven are semi-supervised. The algorithms were implemented in MATLAB R2022a and executed on an Ubuntu 18.04.1 LTS system equipped with an Intel Core i9-9900K CPU (3.60 GHz X 16 threads) and 62.7 GiB of memory.

The experimental datasets consisted of four synthetic datasets that we generated, named Data-A, Data-B, Data-C, and Data-D, in addition to 13 real-world datasets sourced from the UCI repository [30]. These UCI datasets were as follows: Image Segmentation (IS), Seeds, Wine, Rice, Wisconsin Diagnostic Breast Cancer (WDBC), Ecoli, Htru2, Zoo, Glass, Shill Bidding, Anuran Calls, Occupancy Detection, and Machine Failure. Furthermore, three datasets were incorporated from Kaggle: Heart Disease, Pulsar Cleaned, and Bert-Embedded Spam. In total, 20 datasets were used, and Table 1 lists their attributes, such as name, number of instances, number of features, number of reference classes, and coefficient of variation (CV). In previous literature [31], the CV was used to quantify the degree of imbalance in a data distribution, and it was calculated by dividing the standard deviation of class sizes by the mean. For class sizes $N_1, \dots, N_K$, the CV is expressed as follows

$$\text{CV} = s/\bar{N}, \tag{15}$$

where

$$\bar{N} = \frac{\sum_{k=1}^{K} N_k}{K}, \; s = \sqrt{\frac{\sum_{k=1}^{K} (N_k - \bar{N})^2}{K - 1}}.$$

Table 1
Detailed description of 20 Datasets.

ID	Name	Instances	Feature	Classes	CV
D1	Data-A	2250	2	3	1.4468
D2	Data-B	2250	2	3	1.4468
D3	Data-C	5200	2	2	1.3054
D4	Data-D	5400	2	9	2.7500
D5	IS	2310	19	7	0
D6	Seeds	210	7	3	0
D7	Heart Disease	1125	13	2	0.0373
D8	Wine	178	13	3	0.1939
D9	Rice	3810	7	2	0.2042
D10	WDBC	569	30	3	0.3604
D11	Zoo	101	16	7	0.8937
D12	Glass	214	9	6	1.0767
D13	Ecoli	336	7	8	1.1604
D14	Htru2	17 898	8	2	1.1552
D15	Shill Bidding	6321	9	2	1.1122
D16	Anuran Calls	7195	22	10	1.6016
D17	Occupancy Detection	20 560	5	2	0.7608
D18	Machine Failure	9815	7	2	1.3318
D19	Pulsar Cleaned	14 987	7	2	1.3561
D20	Bert-Embedded Spam	5572	768	2	1.0350

In [31], a CV greater than 1 signifies a significant imbalance in class sizes, whereas a CV below 0.3 suggests uniform class sizes. However, no universally accepted CV threshold exists for determining whether a dataset is balanced or imbalanced. For clarity, datasets with a CV under 0.4 were categorized as balanced, and those with a CV above 0.7 were considered imbalanced. Notably, no datasets have a CV between 0.4 and 0.7; hence, this range was undefined.

All datasets were normalized to ensure that each feature has a zero mean and unit variance, and all features were used for clustering. This standard normalization ensures baseline comparability. In practical application, domain-specific preprocessing techniques (such as hybrid feature selection for agricultural sensing [32] and noise reduction for global positioning systems in urban areas [33]) can be utilized to enhance clustering performance by suppressing irrelevant features or outliers in scenarios such as radiation oncology [34], steganography [35], and near-infrared breast imaging [36]. The number of clusters was set to match the reference class count. Convergence was considered achieved when the change in centroid positions between successive iterations became sufficiently small relative to their magnitude, i.e.,

$$\frac{\left(\sum_{k=1}^{K}\|\mathbf{c}_k^{(\tau)}-\mathbf{c}_k^{(\tau-1)}\|_2^2\right)^{1/2}}{\left(\sum_{k=1}^{K}\|\mathbf{c}_k^{(\tau)}\|_2^2\right)^{1/2}}\le 1\mathrm{e}{-3}. \tag{16}$$

To prevent indefinite iterations due to non-convergence, the maximum number of iterations was limited to 500. In our experiments, this limit suffices for the convergence of all algorithms. The centroids were initialized using the K-means++ algorithm [37]. Given the variability in K-means++ outputs and the non-convex nature of the clustering algorithms, which allow multiple local optima, convergence may occur at a local optimum. To approximate the global optimum, the algorithm was executed 100 times, and the solution with the lowest objective value was selected. This process was termed a trial. For each trial, we randomly selected 30% of all the instances from each class as labeled instances, in accordance with previous studies [8,12,38]. To mitigate the bias originating from the selection of labeled instances, 50 trials were conducted and the results were averaged to ensure experimental reliability. Conducting 50 trials with 100 repetitions per trial proved to be sufficient to produce statistically reliable results, as indicated by a low standard deviation. The normalized mutual information (NMI) [39], the adjusted rand index (ARI) [40], and the clustering accuracy index (ACC) [41] were employed as the evaluation metrics. Both NMI and ACC range from 0 to 1, with 0 indicating no mutual agreement and 1 representing the perfect agreement between two clustering results. ARI ranges from −1 to 1, in which −1 implies that the two data clustering results are completely dissimilar, 0 indicates that they are essentially random, and 1 implies that they are perfectly aligned. These evaluation metrics are widely used in existing semi-supervised clustering studies [12,19,38].

Unless otherwise specified, all comparative algorithms were implemented using the default parameters recommended in their respective literature. The scaling factor θ was set to be equal to N/S where N is the total number of instances and S is the number of labeled instances. When 30% of instances are selected as labeled instances, $\theta = 1/0.3 \approx 3.33$. The parameter θ was set such that the unsupervised and supervised terms in the objective function shared the same importance. The same specification was adopted for the comparative algorithms ($\mathrm{SSFCM_{p97}}$, $\mathrm{SSFCM_{k24}}$, SFC-ER, SSPCM, and SPFCM) with such a scaling parameter. This step was done to ensure a fair comparison. In addition, a θ value of $1/0.3$ was observed to provide the optimal results for all the above semi-supervised clustering algorithms.

FCM, PFCM, $\mathrm{SSFCM_{p97}}$, $\mathrm{SSFCM_{k24}}$, SSPCM, SPFCM, FWSSPCM employed a typical fuzzifier value of $m = 2$ [42]. For SSEKM, we set $\alpha = 1$ for the four artificial datasets. For the 16 real-world datasets with varying feature numbers, the parameter α was set proportional to the data variance, as follows

$$\alpha = 2/\bar{d}_{0n}, \tag{17}$$

where $\bar{d}_{0n} = \frac{1}{2}\sum_{n=1}^{N}\|\mathbf{x}_n\|_2^2/N$. The rationale for this selection is explained in Section 3.5. For the prior matrix $\mathbf{F} = [f_{kn}]$, if the nth instance is known to belong to kth cluster, $f_{kn} = 1$; otherwise $f_{kn} = 0$. All the semi-supervised clustering algorithms used for comparison have such prior matrices and the same configuration was adopted for a fair comparison.

4.2. Artificial datasets

Fig. 4 shows the scatter plots for the four artificial datasets alongside selected clustering results. The average values and standard deviations of the evaluation metrics across 50 trials are presented in Table 2, with the best outcomes highlighted in bold. The data points in Data-A followed a Gaussian distribution, whereas Data-B featured points that followed a uniform distribution. Data-C and Data-D comprised a mix of Gaussian and uniform distributions, with Data-C predominantly containing Gaussian-distributed points and Data-D primarily consisting of uniformly distributed points. Additionally, Data-D included a greater number of classes than Data-C. All four artificial datasets exhibited significant imbalance, as indicated by their CV values exceeding one.

As evident from Table 2, the proposed SSEKM demonstrated superior performance on these artificial datasets, achieving over 0.9 NMI, 0.95 ARI, and 99% ACC across all datasets. SSEKM significantly outperformed SSFCM and FWSSPCM, the latter being an SOTA algorithm specifically designed for imbalanced data. SSEKM surpassed EKM on Data-A, Data-B, and Data-D. Additionally, it exhibited a comparable performance with EKM on Data-C.

4.3. Real datasets with balanced data

To evaluate the effectiveness of SSEKM on balanced datasets, it was applied to six real-world datasets: IS, Seeds, Heart Disease, Wine, Rice, and WDBC. These datasets exhibited uniform class sizes, with CV values below 0.4. Therefore, they were classified as balanced according to our criteria. The clustering results are presented in Table 3. SSEKM outperformed all the comparative algorithms on the IS, Heart Disease, and WDBC datasets. For the remaining three datasets, SSEKM performs competitively with the best comparative algorithm, with a small ACC gap of below 1% for the Seeds and Wine datasets, and 2% for the Rice dataset. SSEKM enhanced the performance of EKM across all six

Table 2
Experimental results on four artificial datasets with $CV > 1$. The true labels of 30% of the data are given as supervised knowledge (the best performance is in bold).

Dataset	Measurement	HKM	FCM	MEFC	PFCM	EKM	eSFCM	$SSFCM_{P97}$	$SSFCM_{K24}$	SFC-ER	SSPCM	SPFCM	FWSSPCM	SSEKM
Data-A	NMI	0.5150 ±0.0001	0.4982 ±0.0001	0.2270 ±0.0002	0.4984 ±0.0001	0.9126 ±0.0000	0.5452 ±0.0043	0.5449 ±0.0041	0.5720 ±0.0046	0.5452 ±0.0043	0.6157 ±0.0127	0.5230 ±0.0061	0.8338 ±0.1063	**0.9372** ±0.0136
	ARI	0.2906 ±0.0002	0.2485 ±0.0002	0.0756 ±0.0003	0.2489 ±0.0003	0.9616 ±0.0000	0.3670 ±0.0041	0.3567 ±0.0064	0.4245 ±0.0083	0.3670 ±0.0042	0.6170 ±0.0235	0.3936 ±0.0075	0.9114 ±0.0885	**0.9717** ±0.0081
	ACC	0.6893 ±0.0003	0.6075 ±0.0006	0.4818 ±0.0003	0.6089 ±0.0009	0.9933 ±0.0000	0.7700 ±0.0034	0.7610 ±0.0056	0.8118 ±0.0053	0.7699 ±0.0034	0.9135 ±0.0078	0.8092 ±0.0052	0.9763 ±0.0216	**0.9950** ±0.0015
Data-B	NMI	0.5193 ±0.0000	0.5160 ±0.0000	0.2121 ±0.0001	0.5181 ±0.0001	0.9981 ±0.0057	0.5596 ±0.0016	0.5503 ±0.0020	0.5675 ±0.0027	0.5596 ±0.0016	0.5554 ±0.0060	0.5491 ±0.0056	0.7054 ±0.2248	**1.0000** ±0.0000
	ARI	0.2529 ±0.0000	0.2452 ±0.0000	0.0547 ±0.0000	0.2498 ±0.0003	0.9992 ±0.0023	0.3536 ±0.0041	0.3304 ±0.0050	0.3735 ±0.0067	0.3536 ±0.0040	0.4329 ±0.0128	0.4185 ±0.0113	0.7465 ±0.2008	**1.0000** ±0.0000
	ACC	0.6116 ±0.0000	0.5852 ±0.0002	0.4607 ±0.0016	0.6022 ±0.0009	0.9999 ±0.0004	0.7566 ±0.0037	0.7340 ±0.0053	0.7736 ±0.0055	0.7566 ±0.0037	0.8315 ±0.0075	0.8203 ±0.0069	0.9279 ±0.0562	**1.0000** ±0.0000
Data-C	NMI	0.0909 ±0.0000	0.0797 ±0.0000	0.0892 ±0.0000	0.0782 ±0.0001	**0.9514** ±0.0000	0.1804 ±0.0029	0.1380 ±0.0018	0.1833 ±0.0043	0.1804 ±0.0029	0.1125 ±0.0201	0.1644 ±0.0035	0.2260 ±0.0041	0.9152 ±0.0163
	ARI	0.0181 ±0.0000	0.0034 ±0.0001	0.0159 ±0.0001	0.0015 ±0.0001	**0.9807** ±0.0000	0.1470 ±0.0043	0.0842 ±0.0027	0.1512 ±0.0065	0.1469 ±0.0043	0.1094 ±0.0073	0.1231 ±0.0052	0.2155 ±0.0061	0.9617 ±0.0091
	ACC	0.5754 ±0.0000	0.5302 ±0.0002	0.5689 ±0.0002	0.5237 ±0.0002	**0.9987** ±0.0000	0.7835 ±0.0041	0.7094 ±0.0040	0.7894 ±0.0059	0.7835 ±0.0040	0.7598 ±0.0073	0.7591 ±0.0057	0.8365 ±0.0039	0.9973 ±0.0007
Data-D	NMI	0.4799 ±0.0316	0.3185 ±0.0010	0.3963 ±0.0670	0.2353 ±0.0554	0.9463 ±0.0104	0.3626 ±0.0036	0.3225 ±0.0364	0.4102 ±0.0234	0.3625 ±0.0027	0.2002 ±0.0549	0.2527 ±0.0591	0.7892 ±0.2065	**0.9915** ±0.0599
	ARI	0.1096 ±0.0195	0.0467 ±0.0007	0.1506 ±0.0360	0.0389 ±0.0183	0.9954 ±0.0049	0.0633 ±0.0023	0.0779 ±0.0140	0.1278 ±0.0160	0.0631 ±0.0025	0.0320 ±0.0161	0.0425 ±0.0330	0.6642 ±0.3222	**0.9845** ±0.1096
	ACC	0.3593 ±0.0555	0.2559 ±0.0080	0.4893 ±0.0133	0.3699 ±0.0380	0.9763 ±0.0045	0.3342 ±0.0124	0.4628 ±0.0248	0.5702 ±0.0277	0.3342 ±0.0124	0.4333 ±0.1254	0.3911 ±0.0684	0.8959 ±0.1237	**0.9927** ±0.0516

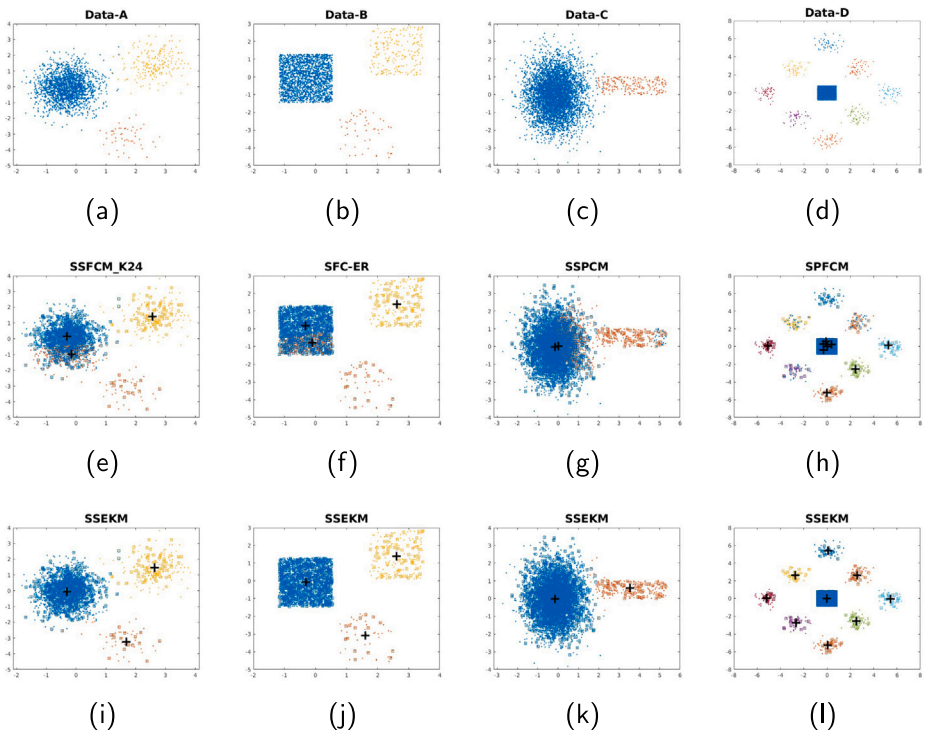

Fig. 4. Scatter diagrams of artificial datasets. (a)–(d) Primitive scatter diagrams with reference class labels (indicated by colors). (e)–(f) Clustering results of some SOTA SSFC algorithms. (i)–(l) Clustering results of the proposed SSEKM. The black crosses represent centroids obtained by each algorithm.

datasets, particularly on IS, where ACC increased by 30%, and on Heart Disease, where ACC improved by 10%. Notably, although SSEKM was specifically designed for imbalanced datasets, it also showed promising results on balanced datasets.

4.4. Real datasets with imbalanced data

The real datasets considered in this study comprised the following: Ecoli, Htru2, Zoo, Glass, Shill Bidding, Anuran Calls, Occupancy Detection, Machine Failure, Pulsar Cleaned, and Bert-Embedded Spam. Among these, Htru2 and Occupancy Detection contained large volumes of data. Glass and Ecoli comprised six and eight classes, respectively, and Zoo was noted for its high dimensionality. These datasets exhibited imbalanced class sizes, with CV values exceeding 0.7. Bert-Embedded Spam featured more than 700 attributes. Given that Euclidean distance is ineffective for clustering in high-dimensional spaces, we applied principal component analysis to the Bert-Embedded Spam dataset. The clustering was performed using the first five principal components, as additional components did not enhance and even degraded the performance. The results are presented in Table 4, with the best outcomes highlighted in bold.

As illustrated in Table 4, SSEKM outperformed all comparative algorithms on seven datasets by a substantial margin. For example, SSEKM achieved ARI scores of 0.7919, 0.9493, and 0.9879 on the Ecoli, Anuran Calls, and Machine Failure datasets, respectively, whereas the corresponding scores obtained by $SSFCM_{k24}$ were 0.6309, 0.6399, and 0.0613. The highest scores exhibited by the comparative algorithms on the above datasets were 0.6309, 0.6721, and 0.9482, respectively. SSEKM ranked second on the Zoo and Shill Bidding datasets, and fourth on the Occupancy Detection. $SSFCM_{k24}$ achieved the highest performance on Zoo, whereas FWSSPCM excelled on Shill Bidding and Occupancy Detection. SSEKM and $SSFCM_{k24}$ exhibited nearly identical NMI and ACC scores (less than their standard deviation) on Zoo, with SSEKM displaying a higher ARI score. FWSSPCM performed more effectively on the Shill Bidding dataset, which comprised nine features, as its feature weighting mechanism is specifically designed for high-dimensional data. For the Occupancy Detection dataset, data points were considerably decentralized in the feature space. Therefore, fuzzy clustering algorithms with a typicality- or possibilistic-based partition (such as SSPCM, SPFCM, and FWSSPCM) offer greater advantages than other algorithms. Overall, this experiment demonstrated the superiority of SSEKM over existing SOTA algorithms on imbalanced datasets. Moreover, SSEKM was one of the most computationally efficient algorithms tested, whereas FWSSPCM exhibited the highest computational cost (kindly refer to Section 4.7). As expected, SSEKM improved the performance of EKM on all the tested datasets.

4.5. Study of parameter impact

4.5.1. Impact of α

We investigated the influence of α on SSEKM performance. SSEKM was executed with α values of 0.1,0.2,0.5,0.8,1,2, 5, 8, and 10 on the four artificial datasets. Fig. 5 presents the corresponding NMI values for the above α values, with the NMI value of EKM serving as a reference. The results demonstrated that SSEKM was highly robust to variations in the value of α.

4.5.2. Impact of θ

The impact of the parameter θ on SSEKM performance was assessed. Recall that θ is used to balance the unsupervised and supervised components within the objective function. We set θ to be proportional to N/S, and in this experiment, $N/S = 1/0.3$. Fig. 6 shows the NMI score as a function of θ for four selected imbalanced datasets. The same trends were observed across the remaining datasets; therefore, the corresponding results were omitted for brevity. Excessively large or small values of θ evidently deteriorated the performance of SSEKM. The optimal value of θ was observed in the range where the unsupervised and supervised components displayed similar importance, which was approximately around $\theta = N/S$. This result highlights the significance of balancing the knowledge of data structure (i.e., unsupervised knowledge) and supervised knowledge.

Table 3
Experimental results on the selected six real-world datasets with $CV < 0.4$. The true labels of 30% of the data are given as supervised knowledge (The best performance is in bold).

Dataset	Measurement	HKM	FCM	MEFC	PFCM	EKM	eSFCM	$SSFCM_{P97}$	$SSFCM_{K24}$	SFC-ER	SSPCM	SPFCM	FWSSPCM	SSEKM
IS	NMI	0.5873 ±0.0008	0.5950 ±0.0057	0.6206 ±0.0009	0.4962 ±0.0241	0.6618 ±0.0138	0.7752 ±0.0079	0.7720 ±0.0091	0.7921 ±0.0100	0.7329 ±0.0158	0.6093 ±0.0409	0.6970 ±0.0147	0.3546 ±0.0319	**0.8027** ±0.0090
	ARI	0.4608 ±0.0005	0.4960 ±0.0059	0.4979 ±0.0005	0.3650 ±0.0149	0.4810 ±0.0089	0.7444 ±0.0129	0.7418 ±0.0111	0.7672 ±0.0160	0.6648 ±0.0181	0.4049 ±0.0629	0.5620 ±0.0180	0.0784 ±0.0652	**0.7825** ±0.0114
	ACC	0.5456 ±0.0003	0.6578 ±0.0146	0.5943 ±0.0002	0.5276 ±0.0139	0.5609 ±0.0118	0.8701 ±0.0097	0.8676 ±0.0068	0.8835 ±0.0099	0.8054 ±0.0111	0.6576 ±0.0373	0.7580 ±0.0207	0.4382 ±0.0547	**0.8930** ±0.0067
Seeds	NMI	0.7279 ±0.0000	0.7318 ±0.0056	0.7496 ±0.0016	0.7170 ±0.0000	0.7315 ±0.0000	0.8091 ±0.0220	0.7985 ±0.0230	0.7981 ±0.0214	**0.8093** ±0.0216	0.7839 ±0.0247	0.7905 ±0.0253	0.7343 ±0.0290	0.7948 ±0.0223
	ARI	0.7733 ±0.0000	0.7768 ±0.0058	0.7968 ±0.0017	0.7607 ±0.0000	0.7715 ±0.0000	0.8512 ±0.0218	0.8418 ±0.0225	0.8408 ±0.0222	**0.8514** ±0.0215	0.8260 ±0.0266	0.8324 ±0.0271	0.7690 ±0.0298	0.8363 ±0.0228
	ACC	0.9190 ±0.0000	0.9209 ±0.0023	0.9285 ±0.0007	0.9143 ±0.0000	0.9190 ±0.0000	0.9483 ±0.0079	0.9450 ±0.0083	0.9446 ±0.0081	**0.9484** ±0.0078	0.9389 ±0.0099	0.9415 ±0.0100	0.9162 ±0.0119	0.9427 ±0.0084
Heart Disease	NMI	0.3162 ±0.0000	0.0142 ±0.0132	0.2500 ±0.0000	0.2326 ±0.0975	0.2571 ±0.0000	0.4272 ±0.0199	0.3967 ±0.0166	0.4273 ±0.0224	0.4259 ±0.0198	0.4345 ±0.0206	0.4207 ±0.0243	0.3971 ±0.0187	**0.4562** ±0.0248
	ARI	0.3641 ±0.0000	0.0033 ±0.0042	0.3140 ±0.0000	0.2995 ±0.1230	0.3162 ±0.0000	0.5178 ±0.0208	0.4940 ±0.0187	0.5249 ±0.0238	0.5145 ±0.0207	0.5364 ±0.0220	0.5212 ±0.0267	0.4959 ±0.0211	**0.5541** ±0.0269
	ACC	0.8020 ±0.0000	0.5249 ±0.0202	0.7805 ±0.0000	0.7653 ±0.0689	0.7815 ±0.0000	0.8592 ±0.0073	0.8516 ±0.0067	0.8623 ±0.0082	0.8588 ±0.0072	0.8663 ±0.0075	0.8610 ±0.0092	0.8522 ±0.0075	**0.8722** ±0.0091
Wine	NMI	0.8759 ±0.0000	0.8759 ±0.0000	0.8759 ±0.0000	0.8097 ±0.0171	0.8920 ±0.0000	0.9137 ±0.0234	0.9090 ±0.0231	0.9063 ±0.0271	**0.9141** ±0.0232	0.8993 ±0.0300	0.9136 ±0.0309	0.8510 ±0.0247	0.9087 ±0.0293
	ARI	0.8975 ±0.0000	0.8975 ±0.0000	0.8975 ±0.0000	0.8249 ±0.0142	0.9134 ±0.0000	0.9329 ±0.0199	0.9289 ±0.0231	0.9259 ±0.0245	0.9332 ±0.0197	0.9270 ±0.0239	**0.9383** ±0.0231	0.8670 ±0.0292	0.9275 ±0.0271
	ACC	0.9663 ±0.0000	0.9663 ±0.0000	0.9663 ±0.0000	0.9396 ±0.0051	0.9719 ±0.0000	0.9781 ±0.0065	0.9767 ±0.0066	0.9757 ±0.0081	0.9782 ±0.0065	0.9763 ±0.0080	**0.9800** ±0.0076	0.9560 ±0.0102	0.9764 ±0.0089
Rice	NMI	0.5685 ±0.0000	0.5688 ±0.0000	0.5682 ±0.0004	0.5742 ±0.0008	0.5659 ±0.0005	0.6706 ±0.0075	0.6675 ±0.0075	0.6715 ±0.0077	0.6705 ±0.0075	0.6732 ±0.0082	0.6699 ±0.0076	**0.6764** ±0.0076	0.6735 ±0.0077
	ARI	0.6815 ±0.0000	0.6824 ±0.0000	0.6817 ±0.0004	0.6876 ±0.0008	0.6777 ±0.0004	0.7758 ±0.0066	0.7733 ±0.0066	0.7768 ±0.0067	0.7758 ±0.0066	0.7781 ±0.0072	0.7749 ±0.0067	**0.7810** ±0.0067	0.7785 ±0.0068
	ACC	0.9129 ±0.0000	0.9131 ±0.0000	0.9129 ±0.0001	0.9147 ±0.0002	0.9117 ±0.0001	0.9404 ±0.0019	0.9397 ±0.0019	0.9407 ±0.0019	0.9404 ±0.0019	0.9411 ±0.0021	0.9402 ±0.0019	**0.9419** ±0.0019	0.9412 ±0.0019
WBDC	NMI	0.5547 ±0.0000	0.5612 ±0.0000	0.5547 ±0.0000	0.5700 ±0.0000	0.5513 ±0.0025	0.7016 ±0.0263	0.6674 ±0.0232	0.6832 ±0.0262	0.6790 ±0.0275	0.6000 ±0.0288	0.6321 ±0.0232	**0.7301** ±0.0283	0.7122 ±0.0324
	ARI	0.6707 ±0.0000	0.6829 ±0.0000	0.6707 ±0.0000	0.6895 ±0.0000	0.6444 ±0.0028	0.8052 ±0.0221	0.7778 ±0.0195	0.7898 ±0.0222	0.7839 ±0.0231	0.6444 ±0.0325	0.6806 ±0.0232	0.8079 ±0.0282	**0.8188** ±0.0261
	ACC	0.9104 ±0.0000	0.9139 ±0.0000	0.9104 ±0.0000	0.9156 ±0.0000	0.9027 ±0.0009	0.9490 ±0.0061	0.9414 ±0.0055	0.9447 ±0.0062	0.9431 ±0.0065	0.9028 ±0.0099	0.9137 ±0.0082	0.9498 ±0.0077	**0.9508** ±0.0072

Table 4
Experimental results on the selected ten real-world datasets with $CV > 0.7$. The true labels of 30% of the data are given as supervised knowledge (the best performance in bold).

Dataset	Measurement	HKM	FCM	MEFC	PFCM	EKM	eSFCM	$SSFCM_{P97}$	$SSFCM_{K24}$	SFC-ER	SSPCM	SPFCM	FWSSPCM	SSEKM
Zoo	NMI	0.8381 ±0.0268	0.7764 ±0.0019	0.8179 ±0.0000	0.6611 ±0.0806	0.7912 ±0.0188	0.8981 ±0.0246	0.8308 ±0.0158	**0.9187** ±0.0257	0.8976 ±0.0247	0.7370 ±0.0529	0.8196 ±0.0634	0.7593 ±0.0875	0.9143 ±0.0343
	ARI	0.7546 ±0.0487	0.6235 ±0.0010	0.6444 ±0.0000	0.4868 ±0.0829	0.8181 ±0.0783	0.8972 ±0.0495	0.6976 ±0.0382	0.9305 ±0.0268	0.8956 ±0.0247	0.5800 ±0.1122	0.7410 ±0.1447	0.7227 ±0.1428	**0.9324** ±0.0434
	ACC	0.8248 ±0.0310	0.6832 ±0.0000	0.7228 ±0.0000	0.6267 ±0.0536	0.8507 ±0.0566	0.9232 ±0.0261	0.8085 ±0.0271	**0.9521** ±0.0174	0.9222 ±0.0258	0.7717 ±0.0793	0.8624 ±0.0868	0.8075 ±0.0734	0.9420 ±0.0251
Glass	NMI	0.3140 ±0.0046	0.3073 ±0.0003	0.3120 ±0.0012	0.0721 ±0.0069	0.3764 ±0.0370	0.3916 ±0.0326	0.3849 ±0.0249	0.4240 ±0.0357	0.4072 ±0.0221	0.3286 ±0.0594	0.3500 ±0.0270	0.3589 ±0.0493	**0.4510** ±0.0385
	ARI	0.1702 ±0.0026	0.1529 ±0.0005	0.1651 ±0.0009	0.0070 ±0.0059	0.1978 ±0.0193	0.2650 ±0.0417	0.2382 ±0.0235	0.2704 ±0.0335	0.2711 ±0.0246	0.1290 ±0.0494	0.1731 ±0.0286	0.1472 ±0.0512	**0.3275** ±0.0533
	ACC	0.4586 ±0.0076	0.4021 ±0.0011	0.4487 ±0.0007	0.3368 ±0.0054	0.4796 ±0.0086	0.5601 ±0.0355	0.5187 ±0.0319	0.5737 ±0.0384	0.5704 ±0.0258	0.4989 ±0.0338	0.5056 ±0.0390	0.4608 ±0.0598	**0.6316** ±0.0468
Ecoli	NMI	0.6379 ±0.0040	0.5695 ±0.0115	0.6096 ±0.0076	0.5012 ±0.0649	0.6426 ±0.0024	0.7063 ±0.0172	0.6846 ±0.0210	0.7119 ±0.0195	0.7072 ±0.0174	0.5987 ±0.0506	0.6749 ±0.0354	0.3852 ±0.0078	**0.7443** ±0.0212
	ARI	0.5032 ±0.0070	0.4142 ±0.0204	0.4815 ±0.0270	0.3357 ±0.0752	0.5157 ±0.0013	0.5995 ±0.0253	0.5588 ±0.0405	0.6309 ±0.0223	0.6022 ±0.0210	0.4661 ±0.0963	0.5944 ±0.1046	0.1897 ±0.0059	**0.7919** ±0.0334
	ACC	0.6461 ±0.0103	0.5769 ±0.0160	0.6238 ±0.0218	0.5302 ±0.0435	0.6482 ±0.0043	0.7506 ±0.0182	0.7255 ±0.0289	0.7756 ±0.0170	0.7533 ±0.0168	0.6595 ±0.0683	0.7408 ±0.0719	0.6095 ±0.0035	**0.8670** ±0.0188
Htru2	NMI	0.4068 ±0.0000	0.4100 ±0.0004	0.4075 ±0.0002	0.1289 ±0.0002	0.5872 ±0.0003	0.6627 ±0.0188	0.5458 ±0.0068	0.6022 ±0.0073	0.6782 ±0.0098	0.3563 ±0.0000	0.4955 ±0.0613	0.4738 ±0.0155	**0.6901** ±0.0091
	ARI	0.6071 ±0.0000	0.5796 ±0.0005	0.6075 ±0.0002	0.0462 ±0.0002	0.7333 ±0.0002	0.8204 ±0.0131	0.7204 ±0.0064	0.7760 ±0.0057	0.8288 ±0.0056	0.4063 ±0.0000	0.6044 ±0.0588	0.6305 ±0.0172	**0.8388** ±0.0057
	ACC	0.9366 ±0.0000	0.9252 ±0.0001	0.9366 ±0.0000	0.6091 ±0.0002	0.9661 ±0.0000	0.9750 ±0.0023	0.9557 ±0.0013	0.9667 ±0.0010	0.9766 ±0.0009	0.9359 ±0.0000	0.9518 ±0.0169	0.9351 ±0.0040	**0.9777** ±0.0008
Shill Bidding	NMI	0.0021 ±0.0001	0.0023 ±0.0000	0.0042 ±0.0002	0.0012 ±0.0000	0.6216 ±0.0000	0.4786 ±0.2058	0.1061 ±0.0077	0.3094 ±0.0098	0.7143 ±0.0110	0.1563 ±0.0145	0.2954 ±0.0040	**0.7935** ±0.0059	0.7588 ±0.0128
	ARI	−0.0006 ±0.0000	−0.0002 ±0.0000	−0.0002 ±0.0000	−0.0002 ±0.0000	0.7914 ±0.0000	0.5320 ±0.2625	0.1159 ±0.0058	0.2954 ±0.0159	0.8344 ±0.0088	0.1721 ±0.0103	0.2738 ±0.0066	**0.8845** ±0.0043	0.8733 ±0.0091
	ACC	0.5020 ±0.0002	0.5059 ±0.0000	0.5090 ±0.0003	0.5035 ±0.0002	0.9674 ±0.0000	0.8777 ±0.0814	0.6939 ±0.0041	0.8001 ±0.0081	0.9718 ±0.0017	0.7353 ±0.0061	0.7887 ±0.0037	**0.9808** ±0.0008	0.9791 ±0.0016
Anuran Calls	NMI	0.6775 ±0.0195	0.5699 ±0.0002	0.6155 ±0.0022	0.4270 ±0.0153	0.6229 ±0.0134	0.6743 ±0.0247	0.6727 ±0.0042	0.7471 ±0.0090	0.7261 ±0.0250	0.4506 ±0.0224	0.6168 ±0.0221	0.3462 ±0.0027	**0.8467** ±0.0079
	ARI	0.5826 ±0.0174	0.3414 ±0.0009	0.4017 ±0.0027	0.2254 ±0.0694	0.5145 ±0.0329	0.5340 ±0.0577	0.4693 ±0.0040	0.6399 ±0.0128	0.6721 ±0.0663	0.3632 ±0.0929	0.5308 ±0.0645	0.0690 ±0.0012	**0.9493** ±0.0033
	ACC	0.6441 ±0.0134	0.4363 ±0.0009	0.5103 ±0.0073	0.3705 ±0.0230	0.5821 ±0.0420	0.6574 ±0.0347	0.6219 ±0.0047	0.7445 ±0.0110	0.7484 ±0.0389	0.5404 ±0.0621	0.6579 ±0.0532	0.4796 ±0.0017	**0.9276** ±0.0070
Occupancy Detection	NMI	0.4890 ±0.0001	0.4720 ±0.0003	0.4710 ±0.0002	0.0539 ±0.0007	0.5389 ±0.0002	0.6297 ±0.0054	0.5886 ±0.0042	0.6250 ±0.0042	0.6305 ±0.0053	0.7340 ±0.0072	0.7087 ±0.0068	**0.7951** ±0.0048	0.6987 ±0.0065
	ARI	0.5959 ±0.0001	0.5699 ±0.0004	0.5670 ±0.0002	0.0209 ±0.0002	0.6750 ±0.0002	0.7479 ±0.0053	0.6968 ±0.0040	0.7365 ±0.0041	0.7487 ±0.0052	0.8548 ±0.0050	0.8369 ±0.0049	**0.8698** ±0.0042	0.8140 ±0.0055
	ACC	0.8912 ±0.0000	0.8825 ±0.0001	0.8815 ±0.0001	0.5772 ±0.0004	0.9163 ±0.0001	0.9366 ±0.0015	0.9218 ±0.0012	0.9332 ±0.0012	0.9368 ±0.0015	0.9656 ±0.0012	0.9612 ±0.0012	**0.9686** ±0.0011	0.9547 ±0.0015

(*continued on next page*)

Table 4 (*continued*).

Machine Failure	NMI	0.0248 ±0.0000	0.0281 ±0.0000	0.0310 ±0.0000	0.0284 ±0.0003	0.5825 ±0.1869	0.0785 ±0.0281	0.0697 ±0.0176	0.0992 ±0.0028	0.8975 ±0.0513	0.0172 ±0.0.0034	0.0005 ±0.0004	0.5079 ±0.0415	**0.9680** ±0.0055
	ARI	−0.0116 ±0.0000	−0.0083 ±0.0000	−0.0052 ±0.0000	0.0007 ±0.0001	0.6506 ±0.1987	0.0515 ±0.0128	0.0378 ±0.0077	0.0613 ±0.0018	0.9482 ±0.0290	0.0654 ±0.0100	0.0050 ±0.0021	0.5882 ±0.0497	**0.9879** ±0.0025
	ACC	0.5673 ±0.0000	0.5418 ±0.0002	0.5202 ±0.0001	0.5146 ±0.0007	0.9863 ±0.0068	0.7003 ±0.0165	0.6599 ±0.0102	0.7015 ±0.0032	0.9974 ±0.0015	0.8683 ±0.0113	0.7149 ±0.0117	0.9836 ±0.0015	**0.9994** ±0.0001
Pulsar Cleaned	NMI	0.0311 ±0.0002	0.0167 ±0.0000	0.0201 ±0.0000	0.0236 ±0.0011	0.0544 ±0.0001	0.0057 ±0.0007	0.0551 ±0.0032	0.0641 ±0.0031	0.0961 ±0.0040	0.0299 ±0.0014	0.0363 ±0.0156	0.0883 ±0.0054	**0.1720** ±0.0150
	ARI	0.0370 ±0.0003	0.0069 ±0.0000	0.0131 ±0.0001	−0.0011 ±0.0002	0.1082 ±0.0002	0.0155 ±0.0011	0.0468 ±0.0017	0.0588 ±0.0014	0.1297 ±0.0048	0.0911 ±0.0034	0.1039 ±0.0364	0.1087 ±0.0065	**0.2545** ±0.0289
	ACC	0.7329 ±0.0008	0.5714 ±0.0001	0.6171 ±0.0004	0.5019 ±0.0012	0.8834 ±0.0002	0.7291 ±0.0029	0.7285 ±0.0028	0.7559 ±0.0019	0.8641 ±0.0039	0.9140 ±0.0026	0.9143 ±0.0225	0.8397 ±0.0065	**0.9267** ±0.0109
Bert-Embedded Spam	NMI	0.1517 ±0.0001	0.1249 ±0.0000	0.0472 ±0.0374	0.1122 ±0.0023	0.6704 ±0.0000	0.6105 ±0.0099	0.3480 ±0.0084	0.4333 ±0.0074	0.6103 ±0.0101	0.2604 ±0.0129	0.4053 ±0.0063	0.3603 ±0.0091	**0.7057** ±0.0115
	ARI	0.0830 ±0.0003	0.0550 ±0.0000	0.0289 ±0.0265	−0.0026 ±0.0007	0.8216 ±0.0000	0.7263 ±0.0110	0.3599 ±0.0118	0.4935 ±0.0100	0.7262 ±0.0112	0.2887 ±0.0133	0.4516 ±0.0101	0.4217 ±0.0109	**0.8281** ±0.0103
	ACC	0.6453 ±0.0002	0.6174 ±0.0000	0.5837 ±0.0390	0.5482 ±0.0012	0.9664 ±0.0000	0.9426 ±0.0028	0.8195 ±0.0053	0.8731 ±0.0035	0.9426 ±0.0028	0.7888 ±0.0065	0.8576 ±0.0039	0.8477 ±0.0043	**0.9665** ±0.0022

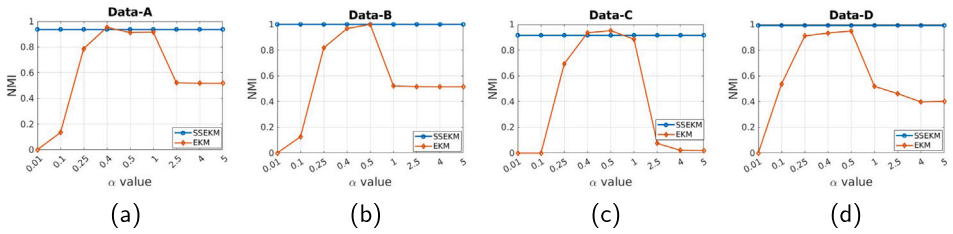

Fig. 5. NMI of SSEKM with different α values on the four artificial datasets. The blue curve is the NMI of SSEKM, and the red curve is the NMI of EKM as a reference.

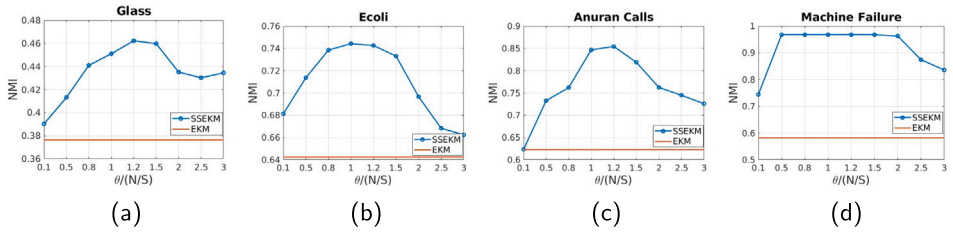

Fig. 6. NMI of SSEKM with different θ values on four real imbalanced datasets. The blue curve is the NMI of SSEKM, and the red curve is the NMI of EKM as a reference.

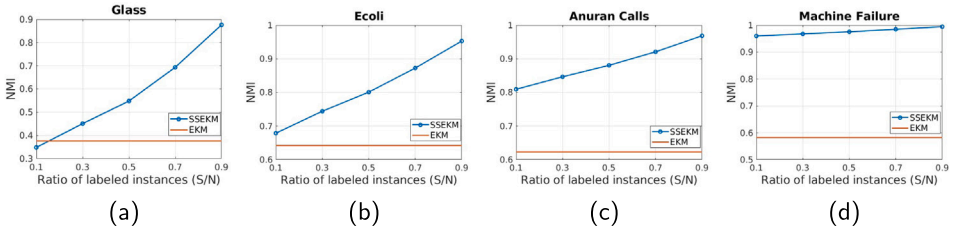

Fig. 7. NMI of SSEKM with different ratios of labeled instances on four real imbalanced datasets. The blue curve is the NMI of SSEKM, and the red curve is the NMI of EKM as a reference.

4.5.3. Impact of the ratio of labeled instances

The impact of the ratio of labeled instances on the performance of SSEKM was examined. Fig. 7 displays the NMI score as a function of this ratio. For some datasets, such as Anuran Calls and Machine Failure, even a small amount of labeled data significantly enhanced the performance of SSEKM. For other datasets, such as Glass, a small amount of labeled data yielded only limited improvement. Nonetheless, a consistent trend was observed: the performance of SSEKM improved as the number of labeled instances increased, as anticipated.

4.6. Robustness against noise

We evaluated the robustness of SSEKM against the presence of noise. We assumed that noise data were uniformly distributed in the feature space, with their boundaries defined by the farthest clean data points. We defined the noise ratio as the ratio of the number of noise samples to the clean data. Fig. 8 shows the NMI score of SSEKM as a function of the noise ratio. As observed, the proposed SSEKM is highly robust against uniformly distributed noise data, as the NMI score is maintained even when the noise ratio increases up to 50%. This is primarily attributed to the equilibrium mechanism inherited from EKM. The final clusters are formed in the equilibrium state, where the attractive and repulsive forces are offset. The equilibrium state is highly stationary, making SSEKM less sensitive to noise.

4.7. Empirical convergence and algorithm efficiency

Table 5 presents the average number of iterations and the average time per run required for the algorithms tested in this study, with each algorithm executed 5000 times. The numbers in brackets indicate the ranking of the corresponding algorithms. AVK and MDK denote the average and median number of iterations and computational time across all test datasets, respectively.

As evident from the table, SSEKM ranks among the fastest algorithms. In terms of AVK, the four fastest algorithms are HKM, SFC-ER, $SSFCM_{k24}$, and SSEKM. As expected, HKM is the fastest, as it is a hard clustering algorithm that does not involve membership calculations. The speeds of the other three fuzzy clustering algorithms are comparable. SSEKM is slower than SFC-ER and SSFCM by 11% and 2%, respectively; however, it is 760% faster than FWSSPCM. Therefore, SSEKM proves to be a highly efficient algorithm. Additionally, for 17 out of 20 datasets, SSEKM requires fewer iterations than EKM to converge, aligning with our theoretical analysis (refer to our convergence analysis, Section 3.4). Consequently, SSEKM is faster than EKM.

5. Conclusion

This study investigated the clustering of imbalanced data by integrating supervision knowledge into an EKM, an unsupervised fuzzy clustering technique. The proposed method, termed SSEKM, was shown to exhibit enhanced clustering efficacy, higher robustness to parameter variations, and lower computational costs. Theoretical analyses elucidated the impact of supervision knowledge on model performance and confirmed the convergence of SSEKM. Comprehensive experiments on both synthetic and real-world datasets revealed that SSEKM surpassed SOTA methods, particularly on imbalanced datasets. SSEKM is also notable for its simplicity and high efficiency, achieving convergence more rapidly than many leading SSFC algorithms. SSEKM introduces a novel clustering strategy applicable in fields such as medical diagnosis and fraud detection, where imbalanced data frequently occurs. The limitation of this study is that it relies on the assumption of a binary prior matrix, which is not optimal for SSEKM. Future research may

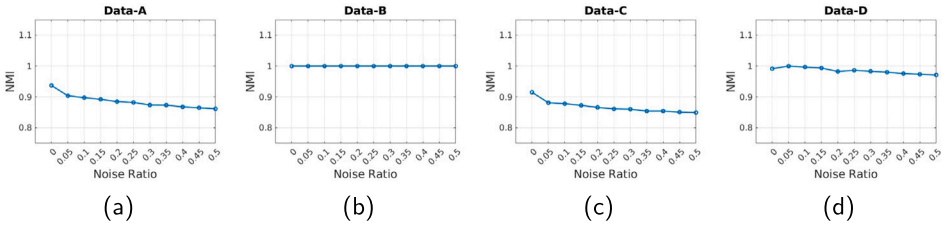

Fig. 8. NMI of SSEKM with different noise ratios on the four artificial imbalanced datasets.

Table 5
Average number of iterations and average calculation time of each algorithm.

Dataset	Measurement	HKM	FCM	MEFC	PFCM	EKM	eSFCM	$SSFCM_{P97}$	$SSFCM_{K24}$	SFC-ER	SSPCM	SPFCM	FWSSPCM	SSEKM
Data-A	Iter	17.3	13.7	29.0	33.5	12.1	14.9	13.5	20.1	15.2	27.7	24.9	8.1	5.8
	Time	0.005(1)	0.008(6)	0.013(9)	0.033(12)	0.007(2)	0.011(8)	0.008(5)	0.011(7)	0.008(4)	0.020(10)	0.032(11)	0.036(13)	0.008(3)
Data-B	Iter	9.2	13.2	30.1	28.8	17.5	16.0	14.1	13.7	15.7	40.4	24.3	4.5	5.4
	Time	0.003(1)	0.008(3)	0.013(9)	0.031(12)	0.009(7)	0.012(8)	0.008(5)	0.008(6)	0.009(4)	0.022(10)	0.031(13)	0.024(11)	0.007(2)
Data-C	Iter	16.3	17.6	36.0	23.9	23.5	26.6	18.3	40.1	26.3	34.4	47.2	9.2	4.6
	Time	0.008(1)	0.014(4)	0.023(8)	0.048(11)	0.017(6)	0.021(7)	0.013(3)	0.025(9)	0.017(5)	0.034(10)	0.076(13)	0.060(12)	0.008(2)
Data-D	Iter	14.8	23.6	25.4	39.8	15.4	24.8	56.4	18.3	25.2	35.9	32.2	4.0	4.7
	Time	0.033(1)	0.091(6)	0.082(5)	0.364(12)	0.107(7)	0.180(9)	0.246(11)	0.080(4)	0.068(3)	0.237(10)	0.367(13)	0.143(8)	0.047(2)
IS	Iter	15.9	40.1	20.9	53.0	24.1	20.5	23.2	8.8	25.7	22.0	16.6	9.5	8.1
	Time	0.021(2)	0.110(9)	0.058(7)	0.218(12)	0.074(8)	0.057(6)	0.045(5)	0.021(1)	0.036(4)	0.158(11)	0.146(10)	0.792(13)	0.030(3)
Seeds	Iter	8.5	10.4	11.1	13.3	12.0	12.6	13.5	8.5	13.2	12.1	8.5	5.2	5.3
	Time	0.001(1)	0.001(5)	0.001(2)	0.003(7)	0.002(6)	0.004(8)	0.004(10)	0.001(4)	0.004(11)	0.004(9)	0.004(2)	0.011(13)	0.001(3)
Heart Disease	Iter	9.9	269.3	12.5	22.7	18.7	11.9	16.3	8.7	12.4	9.4	8.2	7.9	4.9
	Time	0.002(1)	0.074(10)	0.003(2)	0.081(13)	0.005(5)	0.005(6)	0.006(8)	0.005(4)	0.006(7)	0.074(11)	0.075(12)	0.066(9)	0.004(3)
Wine	Iter	7.0	13.5	10.6	25.3	11.7	9.9	12.7	7.5	10.3	11.3	7.9	4.9	7.3
	Time	0.000(1)	0.002(6)	0.001(2)	0.004(10)	0.001(4)	0.002(7)	0.003(8)	0.001(3)	0.003(9)	0.005(12)	0.005(11)	0.015(13)	0.001(5)
Rice	Iter	8.4	8.7	8.9	12.7	9.1	7.6	8.9	7.9	7.5	13.2	8.6	5.2	3.5
	Time	0.004(1)	0.009(9)	0.006(2)	0.021(12)	0.009(8)	0.008(7)	0.008(6)	0.007(5)	0.007(4)	0.017(10)	0.020(11)	0.091(13)	0.007(3)
WDBC	Iter	7.4	10.8	10.6	32.4	9.9	7.8	13.1	7.9	8.0	14.1	10.7	5.8	4.6
	Time	0.001(1)	0.003(4)	0.002(2)	0.009(12)	0.002(3)	0.005(9)	0.005(5)	0.005(8)	0.005(7)	0.008(11)	0.007(10)	0.067(13)	0.005(6)
Zoo	Iter	4.4	31.0	23.8	35.0	498.9	19.3	40.2	10.2	19.1	23.6	25.4	3.0	499.7
	Time	0.001(1)	0.005(4)	0.003(2)	0.009(8)	0.069(12)	0.006(6)	0.008(7)	0.005(3)	0.006(5)	0.011(10)	0.011(9)	0.016(11)	0.071(13)
Glass	Iter	8.5	27.2	29.4	18.8	25.3	22.1	45.8	16.3	22.5	13.5	16.1	5.1	499.8
	Time	0.001(1)	0.006(6)	0.006(3)	0.009(9)	0.006(5)	0.008(7)	0.009(8)	0.005(2)	0.006(4)	0.010(10)	0.011(11)	0.023(12)	0.100(13)
Ecoli	Iter	14.1	42.9	23.6	32.2	22.9	25.6	38.3	18.5	24.2	32.3	23.2	3.3	500
	Time	0.002(1)	0.019(8)	0.010(4)	0.029(12)	0.011(5)	0.014(7)	0.013(6)	0.008(3)	0.007(2)	0.027(10)	0.028(11)	0.026(9)	0.223(13)
Htru2	Iter	12.7	33.8	15.2	21.5	10.8	38.4	28.2	15.2	24.1	11.7	14.3	10.2	3.7
	Time	0.046(2)	0.175(9)	0.067(4)	0.357(12)	0.059(3)	0.163(7)	0.163(8)	0.093(5)	0.103(6)	0.319(11)	0.293(10)	0.849(13)	0.031(1)
Shill Bidding	Iter	12.0	15.0	24.3	22.7	22.3	14.0	22.2	22.9	15.0	24.2	21.1	8.0	4.3
	Time	0.011(1)	0.026(5)	0.029(7)	0.064(11)	0.033(9)	0.019(4)	0.028(6)	0.029(8)	0.018(3)	0.062(10)	0.067(12)	0.273(13)	0.012(2)
Anuran Calls	Iter	22.5	35.7	24.1	39.3	44.3	25.0	45.0	23.3	24.4	28.9	29.4	8.1	14.4
	Time	0.339(2)	0.527(7)	0.377(5)	1.502(11)	0.719(9)	0.431(6)	0.669(8)	0.366(3)	0.369(4)	2.967(12)	1.275(10)	3.913(13)	0.270(1)
Occupancy Detection	Iter	11.5	25.6	28.6	24.6	14.8	10.8	23.8	10.6	10.8	29.5	12.5	5.7	3.2
	Time	0.032(2)	0.132(9)	0.097(7)	0.291(11)	0.067(6)	0.048(4)	0.097(8)	0.049(5)	0.041(3)	0.331(12)	0.219(10)	0.386(13)	0.025(1)
Machine Failure	Iter	13.4	22.0	11.9	127.4	22.5	10.2	23.1	25.5	10.5	22.7	31.2	4.6	3.5
	Time	0.015(2)	0.055(9)	0.022(5)	0.442(13)	0.049(8)	0.022(4)	0.040(6)	0.044(7)	0.019(3)	0.100(10)	0.151(11)	0.211(12)	0.014(1)
Pulsar Cleaned	Iter	25.3	17.1	18.4	60.7	21.4	18.5	17.2	30.6	17.0	15.0	25.1	9.1	8.2
	Time	0.045(2)	0.068(7)	0.052(4)	0.435(12)	0.075(8)	0.056(5)	0.064(6)	0.107(9)	0.047(3)	0.170(10)	0.223(11)	0.564(13)	0.038(1)
Bert-Embedded Spam	Iter	31.6	500.0	498.7	21.2	41.6	20.9	26.2	12.6	21.0	11.8	14.6	13.7	4.4
	Time	0.015(3)	0.618(10)	0.388(9)	0.642(13)	0.045(7)	0.020(5)	0.021(6)	0.012(2)	0.017(4)	0.619(11)	0.633(12)	0.215(8)	0.009(1)
AVK	Iter	13.5	58.5	44.7	34.4	44.0	17.9	25.0	16.4	17.4	21.7	20.1	6.8	79.8
	Time	0.029(1)	0.098(9)	0.063(6)	0.230(11)	0.066(7)	0.055(5)	0.073(8)	0.044(3)	0.040(2)	0.260(12)	0.184(10)	0.389(13)	0.045(4)
MDK	Iter	12.4	22.8	23.7	27.1	20.0	17.3	22.7	14.5	16.4	22.4	18.9	5.8	5.1
	Time	0.007(1)	0.023(8)	0.018(7)	0.056(11)	0.025(9)	0.016(6)	0.013(5)	0.012(2)	0.012(3)	0.0482(10)	0.071(12)	0.079(13)	0.013(4)

explore the construction of more appropriate prior matrices to leverage partial supervision more efficiently.

CRediT authorship contribution statement

Yudong He: Writing – review & editing, Writing – original draft, Validation, Methodology, Investigation.

Declaration of competing interest

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

Data availability

Data will be made available on request.

References

[1] J. Cai, J. Hao, H. Yang, X. Zhao, Y. Yang, A review on semi-supervised clustering, Inform. Sci. 632 (2023) 164–200.
[2] K. Taha, Semi-supervised and un-supervised clustering: A review and experimental evaluation, Inf. Syst. 114 (2023) 102178.
[3] J.C. Bezdek, R. Ehrlich, W. Full, FCM: The fuzzy c-means clustering algorithm, Comput. Geosci. 10 (2–3) (1984) 191–203.
[4] W. Pedrycz, J. Waletzky, Fuzzy clustering with partial supervision, IEEE Trans. Syst. Man Cybern. B 27 (5) (1997) 787–795.
[5] C. Stutz, T.A. Runkler, Classification and prediction of road traffic using application-specific fuzzy clustering, IEEE Trans. Fuzzy Syst. 10 (3) (2002) 297–308.
[6] C. Li, L. Liu, W. Jiang, Objective function of semi-supervised fuzzy c-means clustering algorithm, in: 2008 6th IEEE International Conference on Industrial Informatics, IEEE, 2008, pp. 737–742.
[7] E. Yasunori, H. Yukihiro, Y. Makito, M. Sadaaki, On semi-supervised fuzzy c-means clustering, in: 2009 IEEE International Conference on Fuzzy Systems, IEEE, 2009, pp. 1119–1124.
[8] F. Salehi, M.R. Keyvanpour, A. Sharifi, SMKFC-ER: Semi-supervised multiple kernel fuzzy clustering based on entropy and relative entropy, Inform. Sci. 547 (2021) 667–688.
[9] K. Kmita, K. Kaczmarek-Majer, O. Hryniewicz, Explainable impact of partial supervision in semi-supervised fuzzy clustering, IEEE Trans. Fuzzy Syst. (2024).
[10] L. Liu, X.-J. Wu, Semi-supervised possibilistic fuzzy c-means clustering algorithm on maximized central distance, in: Conference of the 2nd International Conference on Computer Science and Electronics Engineering, ICCSEE 2013, Atlantis Press, 2013, pp. 1366–1370.
[11] V. Antoine, J.A. Guerrero, T. Boone, G. Romero, Possibilistic clustering with seeds, in: 2018 IEEE International Conference on Fuzzy Systems, FUZZ-IEEE, IEEE, 2018, pp. 1–7.
[12] V. Antoine, J.A. Guerrero, G. Romero, Possibilistic fuzzy c-means with partial supervision, Fuzzy Sets and Systems 449 (2022) 162–186.
[13] R. Krishnapuram, J.M. Keller, A possibilistic approach to clustering, IEEE Trans. Fuzzy Syst. 1 (2) (1993) 98–110.
[14] N.R. Pal, K. Pal, J.M. Keller, J.C. Bezdek, A possibilistic fuzzy c-means clustering algorithm, IEEE Trans. Fuzzy Syst. 13 (4) (2005) 517–530.
[15] K. Zhou, S. Yang, Effect of cluster size distribution on clustering: a comparative study of k-means and fuzzy c-means clustering, Pattern Anal. Appl. 23 (2020) 455–466.
[16] H. Xiong, J. Wu, J. Chen, K-means clustering versus validation measures: a data distribution perspective, in: Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2006, pp. 779–784.
[17] A.M. Bensaid, L.O. Hall, J.C. Bezdek, L.P. Clarke, Partially supervised clustering for image segmentation, Pattern Recognit. 29 (5) (1996) 859–871.
[18] S.D. Mai, L.T. Ngo, Multiple kernel approach to semi-supervised fuzzy clustering algorithm for land-cover classification, Eng. Appl. Artif. Intell. 68 (2018) 205–213.
[19] H. Yu, X. Xu, H. Li, Y. Wu, B. Lei, Semi-supervised possibilistic c-means clustering algorithm based on feature weights for imbalanced data, Knowl.-Based Syst. 286 (2024) 111388.
[20] J. Noordam, W. Van Den Broek, L. Buydens, Multivariate image segmentation with cluster size insensitive fuzzy C-means, Chemometr. Intell. Lab. Syst. 64 (1) (2002) 65–78.
[21] P.-L. Lin, P.-W. Huang, C.-H. Kuo, Y. Lai, A size-insensitive integrity-based fuzzy c-means method for data clustering, Pattern Recognit. 47 (5) (2014) 2042–2056.
[22] S. Askari, Fuzzy C-means clustering algorithm for data with unequal cluster sizes and contaminated with noise and outliers: Review and development, Expert Syst. Appl. 165 (2021) 113856.
[23] M. Liu, X. Jiang, A.C. Kot, A multi-prototype clustering algorithm, Pattern Recognit. 42 (5) (2009) 689–698.
[24] S. Zeng, X. Duan, J. Bai, W. Tao, K. Hu, Y. Tang, Soft multi-prototype clustering algorithm via two-layer semi-NMF, IEEE Trans. Fuzzy Syst. (2023).
[25] Y. He, Imbalanced data clustering using equilibrium K-means, 2024, arXiv preprint arXiv:2402.14490.
[26] J.C. Bezdek, Pattern Recognition with Fuzzy Objective Function Algorithms, Springer Science & Business Media, 2013.
[27] L. Groll, J. Jakel, A new convergence proof of fuzzy c-means, IEEE Trans. Fuzzy Syst. 13 (5) (2005) 717–720.
[28] S. Lloyd, Least squares quantization in PCM, IEEE Trans. Inform. Theory 28 (2) (1982) 129–137.
[29] N.B. Karayiannis, MECA: Maximum entropy clustering algorithm, in: Proceedings of 1994 IEEE 3rd International Fuzzy Systems Conference, IEEE, 1994, pp. 630–635.
[30] A. Asuncion, D. Newman, UCI machine learning repository, 2007.
[31] J. Wu, Advances in K-Means Clustering: a Data Mining Thinking, Springer Science & Business Media, 2012.
[32] S. Kaur, M.K. Sachan, A.K. Aggarwal, Classification of royal delicious apples using hybrid feature selection and feature weighting method based on SVM classifier, Scalable Comput.: Pr. Exp. 26 (2) (2025) 940–949.
[33] A. Kumar, Geo-tagged 3d geometric modeling of urban structures by mitigating reflected gps signals using a laser range sensor, in: Science and Information Conference, Springer, 2024, pp. 396–414.
[34] E. Huynh, A. Hosny, C. Guthier, D.S. Bitterman, S.F. Petit, D.A. Haas-Kogan, B. Kann, H.J. Aerts, R.H. Mak, Artificial intelligence in radiation oncology, Nat. Rev. Clin. Oncol. 17 (12) (2020) 771–781.
[35] M. Garg, J.S. Ubhi, A.K. Aggarwal, Current advancements in steganography: A review, Model. Optim. Signals using Mach. Learn. Tech. (2024) 327–348.
[36] A. Kumar, Near-infrared optical imaging of the breast, Int. J. Comput. Assist. Radiol. Surg. 1 (2006) 14–16.
[37] D. Arthur, S. Vassilvitskii, K-means++ the advantages of careful seeding, in: Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms, 2007, pp. 1027–1035.
[38] X. Yin, T. Shu, Q. Huang, Semi-supervised fuzzy clustering with metric learning and entropy regularization, Knowl.-Based Syst. 35 (2012) 304–311.
[39] A. Strehl, J. Ghosh, Cluster ensembles—a knowledge reuse framework for combining multiple partitions, J. Mach. Learn. Res. 3 (Dec) (2002) 583–617.
[40] K.Y. Yeung, W.L. Ruzzo, Details of the adjusted rand index and clustering algorithms, supplement to the paper an empirical study on principal component analysis for clustering gene expression data, Bioinformatics 17 (9) (2001) 763–774.
[41] D. Graves, W. Pedrycz, Kernel-based fuzzy clustering and fuzzy clustering: A comparative experimental study, Fuzzy Sets and Systems 161 (4) (2010) 522–543.
[42] M. Huang, Z. Xia, H. Wang, Q. Zeng, Q. Wang, The range of the value for the fuzzifier of the fuzzy c-means algorithm, Pattern Recognit. Lett. 33 (16) (2012) 2280–2284.