Distance Metric Learning for Soft Subspace Clustering in Composite Kernel Space

Jun Wang[1], Zhaohong Deng[1], Kup-Sze Choi[2], Yizhang Jiang[1], Shitong Wang[1]

[1]School of Digital Media, Jiangnan University, China

[2]Centre for Smart Health, School of Nursing, Hong Kong Polytechnic University, Hong Kong, China

Corresponding author: Jun Wang, Tel: +8613771123570, email: wangjun_sytu@hotmail.com

***Abstract*—Soft subspace clustering algorithms have been successfully used for high dimensional data in recent years. However, the existing algorithms often utilize only one distance function to evaluate the distance between data items on each feature, which cannot deal with datasets with complex inner structures. In this paper, a composite kernel space (CKS) is constructed based on a set of basis kernels and a novel framework of soft subspace clustering is proposed by integrating distance metric learning in the CKS. Two soft subspace clustering algorithms, i.e., entropy weighting fuzzy clustering in CKS for kernel space (CKS-EWFC-K) and feature space (CKS-EWFC-F) are thus developed. In both algorithms, the prototype in the feature space is mapped into the CKS by multiple simultaneous mappings, one mapping for each cluster, which is distinct from existing kernel-based clustering algorithms. By evaluating the distance on each feature in the CKS, both CKS-EWFC-K and CKS-EWFC-F learn the distance function adaptively during the clustering process. Experimental results have demonstrated that the proposed algorithms in general outperform classical clustering algorithms and are immune to ineffective kernels and irrelevant features in soft subspace.**

Keywords: fuzzy clustering, soft subspace clustering, composite kernel space, distance metric learning

1 INTRODUCTION

Clustering has a wide range of applications, including statistics, data mining, and database. It has been extensively studied and many algorithms have been developed [1-7]. Among the studies, soft subspace clustering has emerged as a hot research topic in the fields of data mining in recent years [8-17, 39]. Under the classical framework of *k*-means or fuzzy c-means clustering algorithms, data objects in the entire data space are grouped but assigned with different weights for different dimensions of the clusters. The assignment is based on the importance of the features in identifying the corresponding clusters. For datasets with different clusters correlating to different subsets of features, soft subspace clustering is a more suitable approach since different vectors of feature weights are assigned to each cluster.

According to the ways of dataset partitioning, soft subspace clustering algorithms [8-20] can be divided into two categories, namely, *soft subspace hard clustering* and *soft subspace fuzzy clustering*. For the former, each data object belongs to only one cluster [8, 11-13], while for the latter, each data object belongs to every cluster to a certain degree [10, 17]. Besides, soft subspace fuzzy clustering can deal with overlapping cluster boundaries. On the other hand, according to the way of soft subspace weighting, soft subspace clustering can also be classified into f*uzzy weighting subspace clustering* and *entropy weighting subspace clustering* [10]. Typical fuzzy weighting subspace clustering algorithms include attributes-weighting algorithm (AWA) [8], fuzzy weighting *k*-means (FWKM) [12], fuzzy subspace clustering (FSC) [11] and partition-indexed soft subspace clustering (PI-SSC) [17]. The algorithms assign a fuzzy weight w_{jh}^{α} to the *h*th feature of the *j*th cluster and adjust the feature weights for each cluster automatically during the clustering process. Entropy weighting subspace clustering algorithms include entropy weighting *k*-means (EWKM) [13], clustering objects on subsets of attributes (COSA) [20] and enhanced soft subspace clustering (ESSC) [10]. The algorithms utilize entropy to control the feature weights in each cluster.

Although many soft subspace clustering algorithms have been developed for different application areas, there are still rooms to further improve the performance. A major weakness of soft subspace clustering is the lack of algorithms that are universal for various real world applications. In other words, given a particular soft subspace clustering algorithm, the clustering results can be satisfactory for some datasets while inferior for others. This is because existing soft subspace clustering algorithms utilize only one fixed distance function to evaluate the relationships between data items in two patterns during the clustering process. However, data items in two patterns of different datasets could exhibit different and complex relationships which cannot be described simply by a distance function. Moreover, as the clustering process proceeds, the relationships between data items may change from time to time while the existing soft subspace clustering techniques cannot adapt to the change by updating the distance computation, thereby leading to performance degradation.

To improve the performance of soft subspace clustering, it is necessary to evaluate the relationship between data items

adaptively and a distance metric learning strategy is thus in demand. Recent studies have shown that learning the distance function from the data can improve the performance effectively. Depending on the availability of the training data, algorithms for distance metric learning can be divided into *supervised* and *unsupervised* approaches. In supervised distance metric learning algorithms, labeled data or side information are utilized to learn the distance function such that data points from the same class are put closely together whereas those from different classes to moved far apart. Representative approaches include convex optimization approach [21], information-theoretic approach [22], smooth optimization approach [23] and alternating optimization approach [40]. On the other hand, unsupervised distance metric learning is a more challenging approach due to the lack of any prior knowledge. In the absence of constraint or class label information, most unsupervised distance metric learning algorithms are in general developed to exploit the underlying manifold structure of the data. Typical unsupervised approaches include adaptive metric learning algorithm (AML) [29], nonlinear adaptive distance metric learning algorithm (NAML) [25], adaptive metric learning for self-organizing incremental neural network (SOINN-AML) [27], locally linear metric adaptation (LLMA) [24]. However, all the above clustering algorithms are developed based on distance computation in full space, which is different from the situation in soft subspace clustering algorithms where distance computation is performed based on data items along with each feature. Thus, it is necessary to develop distance metric learning approach so that the most suitable relationship between data items along with each feature can be learned in an unsupervised way.

In this paper, a distance metric learning mechanism for soft subspace clustering is investigated. First, a composite kernel space (CKS) is constructed by linear combination of a set of basis kernel mappings. With the mechanism of distance metric learning, the distance between data items on each feature can be learned adaptively in this CKS. Accordingly, a novel framework of soft subspace clustering is proposed by integrating distance metric learning in the CKS. Especially, two novel soft subspace clustering algorithms, i.e., entropy weighting fuzzy clustering in CKS for kernel space (CKS-EWFC-K) and feature space (CKS-EWFC-F) are proposed, with suffixes K and F in the abbreviations standing for the kernel space and feature space respectively. In both algorithms, the prototype in the feature space is mapped into the CKS by a class of mappings simultaneously, one mapping for each cluster. The mechanism is different from existing kernel-based clustering algorithms. Based on fuzzy partition of the datasets, the proposed algorithms simultaneously locate clusters in CKS and identify the optimal kernel weights for a combination of kernel sets. The incorporation of soft subspace and the automatic adjustment of kernel weights in CKS enable adaptive computation of the distance between data items. Hence, the clustering quality of CKS-EWFC-K and CKS-EWFC-F can be improved for various applications. For easy reference and to enhance the readability of the paper, the major notations used in this paper are summarized in Table 1.

Table 1 Notations used in this paper

Notations	Descriptions
c	Cluster number
m	Fuzziness of membership
n	Size of dataset
s	Number of features
p	Number of mappings or kernels
u_{ik}	Fuzzy memberships
w_{ih}	Feature weight
$\mathbf{Z}$	Cluster center matrix
$\mathbf{W}$	Fuzzy weighting matrix
$\mathbf{U}$	Fuzzy partition matrix in fuzzy clustering algorithms, or hard partition matrix in hard clustering algorithms
$\mathbf{V}$	Kernel weights matrix
α	fuzziness of $\mathbf{W}$
η, γ, ε, ε_u, ε_w	Coefficients for penalty terms
x_{ih}	The hth feature of data point $\mathbf{x}_i$
z_{jh}	The hth feature of cluster center $\mathbf{v}_j$

The rest of the paper is organized as follows. In Section 2, related work on soft subspace clustering is reviewed. In Section 3, the composite kernel space is presented, followed by the discussion of the CKS-EWFC-K and CKS-EWFC-F algorithms and their properties. The experiment results are reported and analyzed in Section 4. Conclusions are given in Section 5.

2 RELATED WORK

Soft subspace clustering has been a hot research topic in recent years [8-20]. Many algorithms have been developed and the ultimate goal, generally speaking, is to find the local minimum of the objective function J below

$$J(\mathbf{U},\mathbf{W},\mathbf{Z})=\sum_{j=1}^{c}\sum_{i=1}^{n}u_{ji}^{m}\sum_{h=1}^{s}w_{jh}^{\alpha}d^{2}(x_{ih},z_{jh})+\mathrm{H}(\mathbf{U},\mathbf{W}), \quad (1)$$

under the constraints $\sum_{j=1}^{c}u_{ji}=1$ and $\sum_{h=1}^{s}w_{jh}=1$. In the equation, the first term $\sum_{j=1}^{c}\sum_{i=1}^{n}u_{ji}^{m}\sum_{h=1}^{s}w_{jh}^{\alpha}d^{2}(x_{ih},z_{jh})$ is interpreted as the total weighted distance between each data object $\mathbf{x}_i$, i=1, 2, …, n, and the cluster centers $\mathbf{z}_j$, j=1, 2, …, c; and the second term $\mathbf{H}(\mathbf{U},\mathbf{W})$ is a penalty term which is often used to optimize the performance of the algorithm. The term $d(x_{ih}, z_{jh})$ in Eq.(1) is a dissimilarity measure between x_{ih} and z_{jh}, which is often taken as the Euclidean distance, i.e. $d(x_{ih},z_{jh}) = ||x_{ih}-z_{jh}||$, in the original feature space. Other distance functions have also been used in some recent studies, e.g. Minkowski distance function [30], alternative distance function [15], ε-insensitive distance [10] and the Euclidean distance function in kernel space [16]. In this paper, we present a new taxonomy for soft subspace clustering based on the distance function adopted.

2.1 Euclidean distance

The attribute weighting algorithm proposed by Chan et al. is one of the earliest soft subspace clustering algorithms. It adopts the Euclidean distance function [8] and the fuzzy weighting strategy is incorporated into the learning criterion. The objective function of AWA J_{AWA} is formulated as follows:

$$J_{AWA}(\mathbf{U},\mathbf{W},\mathbf{Z})=\sum_{j=1}^{c}\sum_{i=1}^{n}u_{ji}\sum_{h=1}^{s}w_{jh}^{\alpha}\left(x_{ih}-z_{jh}\right)^2 \tag{2a}$$

$$s.t.\ \ u_{ji}\in\{0,1\},\ \sum_{j=1}^{c}u_{ji}=1,\ i=1,2,...,n$$
$$w_{jh}\in[0,1],\ \sum_{h=1}^{s}w_{jh}=1,\ j=1,2,...,c \tag{2b}$$

In order to avoid the problem of zero dispersion of a dimension in a cluster, Eq.(2a) is modified to Eq.(3a) by appending a penalty term, i.e.,

$$J_{FWKM}(\mathbf{U},\mathbf{W},\mathbf{Z})=\sum_{j=1}^{c}\sum_{i=1}^{n}u_{ji}\sum_{h=1}^{s}w_{jh}^{\alpha}\left(x_{ih}-z_{jh}\right)^2+\varepsilon\sum_{j=1}^{c}\sum_{i=1}^{n}u_{ji}\sum_{h=1}^{s}w_{jh}^{\alpha} \tag{3a}$$

(should it be J_{AWA} (U,W,Z) in this step)

$$s.t.\ \ u_{ji}\in\{0,1\},\ \sum_{j=1}^{c}u_{ji}=1,\ i=1,2,...,n$$
$$w_{jh}\in[0,1],\ \sum_{h=1}^{s}w_{jh}=1,\ j=1,2,...,c \tag{3b}$$

By using Eq.(3), a soft subspace clustering algorithm called FWKM is developed [12]. From Eq.(3a), the objective function J_{FWKM} is given by

$$J_{FWKM}(\mathbf{U},\mathbf{W},\mathbf{Z})=\sum_{j=1}^{c}\sum_{i=1}^{n}u_{ji}\sum_{h=1}^{s}w_{jh}^{\alpha}\left(\left(x_{ih}-z_{jh}\right)^2+\varepsilon\right), \tag{4}$$

where $d^2(x_{ih},z_{jh})=\left(x_{ih}-z_{jh}\right)^2+\varepsilon$, with ε>0, is the distance function. Hence, AWA can be regarded as a special case of FWKM when ε=0.

Using a similar learning criterion, FSC is also developed [11] and the objective function J_{FSC} can be formulated as follows:

$$J_{FSC}(\mathbf{U},\mathbf{W},\mathbf{Z})=\sum_{j=1}^{c}\sum_{i=1}^{n}u_{ji}\sum_{h=1}^{s}w_{jh}^{\alpha}\left(x_{ih}-z_{jh}\right)^2+\varepsilon\sum_{j=1}^{c}\sum_{h=1}^{s}w_{jh}^{\alpha}, \tag{5a}$$

$$s.t.\ \ u_{ji}\in\{0,1\},\ \sum_{j=1}^{c}u_{ji}=1,\ i=1,2,...,n$$
$$w_{jh}\in[0,1],\ \sum_{h=1}^{s}w_{jh}=1,\ j=1,2,...,c \tag{5b}$$

where $d^2(x_{ih},z_{jh})=\left(x_{ih}-z_{jh}\right)^2$ and $\mathrm{H}(\mathbf{U},\mathbf{W})=\varepsilon\sum_{j=1}^{c}\sum_{h=1}^{s}w_{jh}^{\alpha}$.

It is clear from Eq.(2) to Eq.(5) that a fuzzy weight $w_{jh}{}^{\alpha}$ is assigned to the features of different clusters with a fuzzy index α, which is a common characteristic of the above algorithms. Therefore, algorithms of this kind are called fuzzy weighting subspace clustering algorithms. In order to ensure the convergence of the algorithms, the fuzzy index α should be greater than 1.

Similarly, the concept of entropy has been introduced into the clustering process to develop weighting subspace clustering algorithms. For maximum entropy clustering algorithms, the fuzzy memberships are controlled by maximizing the entropy. EWKM [13] is a representative algorithm of this kind and the objective function J_{EWKM} is given by

$$J_{EWKM}(\mathbf{U},\mathbf{W},\mathbf{Z}) = \sum_{j=1}^{c}\sum_{i=1}^{n} u_{ji} \sum_{h=1}^{s} w_{jh}\left(x_{ih} - z_{jh}\right)^2 + \eta \sum_{j=1}^{c}\sum_{h=1}^{s} w_{jh} \ln w_{jh} \tag{6a}$$

$$s.t.\ \ u_{ji} \in \{0, 1\},\ \sum_{j=1}^{c} u_{ji} = 1,\ i = 1, 2, ..., n$$
$$w_{jh} \in [0, 1],\ \sum_{h=1}^{s} w_{jh} = 1,\ j = 1, 2, ..., c \tag{6b}$$

From Eq.(6a), it is clearly that EWKM simultaneously minimize the within-cluster dispersion and maximize the negative weight entropy in the clustering process, so as to improve the clustering quality of the algorithm.

All the algorithms mentioned above are based on hard partition of the datasets. The PI-SSC [17] proposed by Wang et al. is a typical soft subspace clustering algorithm based on fuzzy partition of the datasets. Similar to the aforementioned soft subspace clustering algorithms, PI-SSC adopts the Euclidean distance function to compute the dissimilarity between data items. In PI-SSC, partition index and fuzzy clustering are integrated into the framework of fuzzy weighting subspace clustering. The integration of the partition index as a piece of additional information can improve the clustering quality of PI-SSC. The objective function $J_{PI\text{-}SSC}$ is given by

$$J_{PI-SSC}(\mathbf{U},\mathbf{W},\mathbf{Z}) = \sum_{j=1}^{c}\sum_{i=1}^{n} u_{ji}^{m} \sum_{h=1}^{s} w_{jh}^{\alpha}\left(x_{ih} - z_{jh}\right)^2 + \varepsilon_w \sum_{j=1}^{c}\sum_{h=1}^{s} w_{jh}^{\alpha} + \varepsilon_u \sum_{i=1}^{n}\left(\sum_{j=1}^{c} u_{ji}^{m}\right). \tag{7a}$$

$$s.t.\ \ u_{ji} \in [0, 1],\ \sum_{j=1}^{c} u_{ji} = 1,\ i = 1, 2, ..., n$$
$$w_{jh} \in [0, 1],\ \sum_{h=1}^{s} w_{jh} = 1,\ j = 1, 2, ..., c \tag{7b}$$

2.2 Minkowski distance

While the Euclidean distance function is commonly used in most soft subspace clustering algorithms, other distance functions can also be considered. Amorim et al. attempted to improve fuzzy weighting soft subspace clustering algorithms by employing the Minkowski distance function to develop the Minkowski metric Weighted K-Means (MWK-Means) algorithm [14]. The objective function $J_{MWK\text{-}Means}$ of the algorithm is defined as follows:

$$J_{MWK-Means}(\mathbf{U},\mathbf{W},\mathbf{Z}) = \sum_{j=1}^{c}\sum_{i=1}^{n} u_{ji} \sum_{h=1}^{s} w_{jh}^{\alpha}\left|x_{ih} - z_{jh}\right|^{p}, \tag{8}$$

where the conventional Euclidean distance is replaced by the Minkowski metric, with p as the parameter. We can easily see that when p=2, AWA [8] is indeed a special case of MWK-Means.

2.3 Alternative distance

In order to improve the robustness of soft subspace fuzzy clustering, Pan et al. proposed a robust soft subspace clustering algorithm called Alternative Soft Subspace Clustering (ASSC) by incorporating an alternative distance metric into the framework of entropy weighting subspace clustering [15]. The learning criterion J_{ASSC} is formulated as follows:

$$J_{ASSC}(\mathbf{U},\mathbf{W},\mathbf{Z})=\sum_{j=1}^{c}\sum_{i=1}^{n}u_{ji}^{m}\sum_{h=1}^{s}w_{jh}\left(1-\exp\left(-\beta_h\left(x_{ih}-z_{jh}\right)^2\right)\right)+\eta\sum_{j=1}^{c}\sum_{h=1}^{s}w_{jh}\ln w_{jh}, \tag{9a}$$

under the following constraints

$$\begin{aligned} &s.t.\ \ u_{ji}\in[0,\,1],\ \textstyle\sum_{j=1}^{c}u_{ji}=1,\ i=1,\,2,\,...,\,n \\ &w_{jh}\in[0,\,1],\ \textstyle\sum_{h=1}^{s}w_{jh}=1,\ j=1,\,2,\,...,\,c \end{aligned} \tag{9b}$$

Theoretical analysis of the robustness of ASSC has been made based on M-estimator and the convergence of the algorithm is guaranteed [15].

2.4 Robust ε-insensitive distance

Based on the ε-insensitive distance function, the novel soft subspace clustering algorithm ESSC is proposed by Deng et al. to enhance the robustness of clustering. In ESSC, the idea of fuzzy clustering and the between-cluster information are integrated into the learning criterion J_{ESSC} of the entropy weighting subspace clustering. J_{ESSC} is given by [10]

$$J_{ESSC}(\mathbf{U},\mathbf{W},\mathbf{Z})=\sum_{j=1}^{c}\sum_{i=1}^{n}u_{ji}\sum_{h=1}^{s}w_{jh}\left(x_{ih}-z_{jh}\right)^2-\gamma\sum_{i=1}^{c}\sum_{k=1}^{n}u_{ji}\sum_{h=1}^{s}w_{ih}\left(z_{ih}-z_{0h}\right)^2+\eta\sum_{j=1}^{c}\sum_{h=1}^{s}w_{jh}\ln w_{jh} \tag{10a}$$

$$\begin{aligned} &s.t.\ \ u_{ji}\in[0,\,1],\ \textstyle\sum_{j=1}^{c}u_{ji}=1,\ i=1,\,2,\,...,\,n \\ &w_{jh}\in[0,\,1],\ \textstyle\sum_{h=1}^{s}w_{jh}=1,\ j=1,\,2,\,...,\,c \end{aligned} \tag{10b}$$

The objective function of ESSC can be further expressed as

$$J_{ESSC}(\mathbf{U},\mathbf{W},\mathbf{Z})=\sum_{j=1}^{c}\sum_{i=1}^{n}u_{ji}\sum_{h=1}^{s}w_{jh}\left(\left(x_{ih}-z_{jh}\right)^2-\gamma\left(z_{ih}-z_{0h}\right)^2\right)+\eta\sum_{j=1}^{c}\sum_{h=1}^{s}w_{jh}\ln w_{jh}, \tag{11}$$

where $d^2(x_{ih},z_{jh})=\left(x_{ih}-z_{jh}\right)^2-\gamma\left(z_{ih}-z_{0h}\right)^2$ is indeed a robust ε-insensitive distance function. Attributed to the introduction of the distance function into soft subspace fuzzy clustering, the ESSC outperforms some soft subspace clustering algorithms that adopt the Euclidean distance function.

2.5 Distance evaluation in kernel space

For better modeling and discovery of the nonlinear relationships underlying the data, a nonlinear mapping φ is used in kernel methods to map the input data from the original feature space to a new space of higher dimensionality, i.e. kernel space. For example, Shen et al. proposed the weighted fuzzy kernel clustering algorithm (WFKCA) [16] based on distance computation of

the data items in the kernel space and fuzzy partition of the datasets. It can be regarded as an extension of AWA [8]. The learning criterion J_{WFKCA} of WFKCA is formulated as follows:

$$J_{WFKCA}(\mathbf{U},\mathbf{W},\mathbf{Z})=\sum_{j=1}^{c}\sum_{i=1}^{n}u_{ji}^{m}\sum_{h=1}^{s}w_{jh}^{\alpha}\left(\varphi\left(x_{ih}\right)-\varphi\left(z_{jh}\right)\right)^{2} \tag{12a}$$

$$s.t.\ \ u_{ji}\in[0,\,1],\ \sum_{j=1}^{c}u_{ji}=1,\ i=1,\,2,\,...,\,n$$
$$w_{jh}\in[0,\,1],\ \sum_{h=1}^{s}w_{jh}=1,\ j=1,\,2,\,...,\,c \tag{12b}$$

WFKCA has demonstrated better performance than classical clustering algorithms on some popular datasets [16], but given a specific learning task, it is difficult to select suitable kernel types and inappropriate parameters for the kernels. Incorrect choices can seriously affect the performance of WFKCA.

2.6 The research problem

The soft subspace clustering algorithms discussed above are summarized in Table 2. It is clear that all these algorithms only utilize one single and fixed distance function. They are thus limited for the clustering of a few datasets, not universally applicable to datasets of different inner structures in different applications. This remains a challenging problem for soft subspace clustering applications. Distance metric learning appears to be a promising approach to solve this problem. The research has resulted in some effective solutions. In this paper, novel distance metric learning models for soft subspace clustering will be developed in the CKS which is constructed based on a set of basis kernels. Compared with existing distance metric learning approaches, the uniqueness of our work is that the distance function used for distance computation along with each feature is obtained by unsupervised learning during the soft subspace clustering process. Besides, the prototype in the feature space is mapped into the CKS through multiple simultaneous mappings, one mapping for each cluster. Two novel soft subspace clustering algorithms, i.e. CKS-EWFC-K and CKS-EWFC-F, are developed and their properties are investigated in detail. The effectiveness of the methods is proven with comprehensive experiments.

Table 2 Characteristics of soft subspace clustering algorithms with different distance functions

Clustering approach	*Algorithm*	$d^2(x_{ih},z_{jh})$	*partitioning method*	*subspace weighting method*	Penalty H(**U**,**W**)
Euclidean distance based soft subspace clustering	AWA[8]	$\left(x_{ih}-z_{jh}\right)^2$	*hard*	*fuzzy*	/
	FSC [11]		*hard*	*fuzzy*	$\varepsilon\sum_{j=1}^{c}\sum_{h=1}^{s}w_{jh}^{\alpha}$
	EWKM [13]		*hard*	*entropy*	$\eta\sum_{j=1}^{c}\sum_{h=1}^{s}w_{jh}\ln w_{jh}$
	PI-SSC[17]		*fuzzy*	*fuzzy*	$\varepsilon_w\sum_{j=1}^{c}\sum_{h=1}^{s}w_{jh}^{\alpha}+\varepsilon_u\sum_{i=1}^{n}\sum_{j=1}^{c}u_{ji}^{m}$
	FWKM [12]	$\left(x_{ih}-z_{jh}\right)^2+\varepsilon$ (ε>0)	*hard*	*fuzzy*	/

ε-insensitive distance based soft subspace clustering	ESSC [10]	$\left(x_{ih}-z_{jh}\right)^2-\gamma\left(z_{ih}-z_{0h}\right)^2$ (γ>0)	*fuzzy*	*entropy*	$\eta\sum_{j=1}^{c}\sum_{h=1}^{s}w_{jh}\ln w_{jh}$
Minkowski distance based soft subspace clustering	MWK-Means [14]	$\left\|x_{ih}-z_{jh}\right\|^p$	*hard*	*fuzzy*	/
Alternative distance based soft subspace clustering	ASSC [15]	$1-\exp\left(-\beta_h\left(x_{ih}-z_{jh}\right)^2\right)$	*fuzzy*	*entropy*	$\eta\sum_{j=1}^{c}\sum_{h=1}^{s}w_{jh}\ln w_{jh}$
Kernelized distance based soft subspace clustering	WFKCA [16]	$\left(\varphi\left(x_{ih}\right)-\varphi\left(z_{jh}\right)\right)^2$	*fuzzy*	*fuzzy*	/

3. DISTANCE METRIC LEARNING

3.1 Unsupervised distance metric learning in composite kernel space

To enable universal soft subspace clustering for various real world applications, the key is to adopt an approach that can learn a suitable distance function in an unsupervised manner during the clustering process. However, most current studies on unsupervised distance metric learning only consider it as a dimensional reduction problem [28], which cannot be directly applied to soft subspace clustering because the algorithms could only evaluate distance between data items on one single feature.

In kernel methods, data in the original feature space are mapped into an unknown high dimensional reproducible Hilbert space, i.e. kernel space, so that the actual distance function between the data items can be fully determined to improve the learning capability of the linear machines [31]. Consider a set of unknown mappings $\{\phi_t\}$, t=1, 2…, p. For each mapping, the input data $\mathbf{x}$ are mapped as a L_t-dimensional vector $\phi_t(\mathbf{x})$ in kernel space. Let $\{K_1, K_2, \ldots, K_p\}$ be the Mercer kernels corresponding to the implicit mappings, we have

$$K_t(\mathbf{x}_i,\mathbf{x}_j) = \phi_t(\mathbf{x}_i)^T \phi_t(\mathbf{x}_j),\ t=1, 2\ldots, p, \tag{13}$$

where $\mathbf{x}_i$ and $\mathbf{x}_j$ are two data points in the feature space. In Eq.(13), $K_t(\mathbf{x}_i, \mathbf{x}_j)$ can be regarded as a similarity measure between $\phi_t(\mathbf{x}_i)$ and $\phi_t(\mathbf{x}_j)$ in the kernel space generated by the tth mapping. To combine these kernels, a new set of independent mappings, $\Phi = \{\Phi_1, \Phi_2, \ldots, \Phi_p\}$, can be constructed from the original mappings as follows.

$$\Phi_1(\mathbf{x})=\begin{bmatrix}\phi_1(\mathbf{x})\\ \mathbf{0}\\ \ldots\\ \mathbf{0}\end{bmatrix},\ \Phi_1(\mathbf{x})=\begin{bmatrix}\mathbf{0}\\ \phi_2(\mathbf{x})\\ \ldots\\ \mathbf{0}\end{bmatrix},\ \ldots,\ \Phi_p(\mathbf{x})=\begin{bmatrix}\mathbf{0}\\ \mathbf{0}\\ \ldots\\ \phi_p(\mathbf{x})\end{bmatrix} \tag{14}$$

Each mapping $\Phi_t(\mathbf{x})$, t=1, 2…, p, converts $\mathbf{x}$ into an L-dimensional vector, where L_t is the dimensionality of ϕ_t and $L = \sum_{t=1}^{p} L_t$. While the implicit mappings ϕ_t do not have the same dimensionality, the kernel spaces spanned by Φ_t have the same

dimensionality. In this way, a well-defined linear combination can be achieved. Moreover, the following equation can also be obtained,

$$\Phi_{t_1}(\mathbf{x})^T \cdot \Phi_{t_2}(\mathbf{x}) = \begin{cases} \phi_t(\mathbf{x})^T \phi_t(\mathbf{x}) = K_t(\mathbf{x},\mathbf{x}) & t_1 = t_2 \\ 0 & t_1 \neq t_2 \end{cases},$$

where $\Phi_t(\mathbf{x})$ forms a set of orthogonal bases in the CKS. Thus, any *L*-dimensional vector in the CKS can be expressed as a linear combination of the orthogonal bases.

Suppose **x** is the prototype in feature space. Let

$$\Psi(\mathbf{x}) = \begin{bmatrix} \phi_1(\mathbf{x}) & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \phi_2(\mathbf{x}) & & \mathbf{0} \\ \vdots & \vdots & & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \phi_p(\mathbf{x}) \end{bmatrix} \begin{bmatrix} v_{11} \cdots v_{1p} \\ v_{21} \quad v_{2p} \\ \vdots \quad \vdots \\ v_{c1} \cdots v_{cp} \end{bmatrix}^T = \begin{bmatrix} \sum_{t=1}^{p} v_{1t}\Phi_t(\mathbf{x}) & \sum_{t=1}^{p} v_{2t}\Phi_t(\mathbf{x}) & \cdots & \sum_{t=1}^{p} v_{ct}\Phi_t(\mathbf{x}) \end{bmatrix} = \begin{bmatrix} \Psi_1(\mathbf{x}) & \cdots & \Psi_c(\mathbf{x}) \end{bmatrix},$$

where

$$\Psi_j(\mathbf{x}) = \sum_{t=1}^{p} v_{jt}\Phi_t(\mathbf{x}),\ j = 1,2,\ldots c \text{ and } v_{jt} \in R \tag{15}$$

denotes the *j*th vector in the CKS and the coordinates are $[v_{j1}\ldots\ v_{jp}]^T$. The equation can be interpreted with Fig. 1, where the prototype **x** in the feature space can be mapped into the CKS by a class of mappings simultaneously.

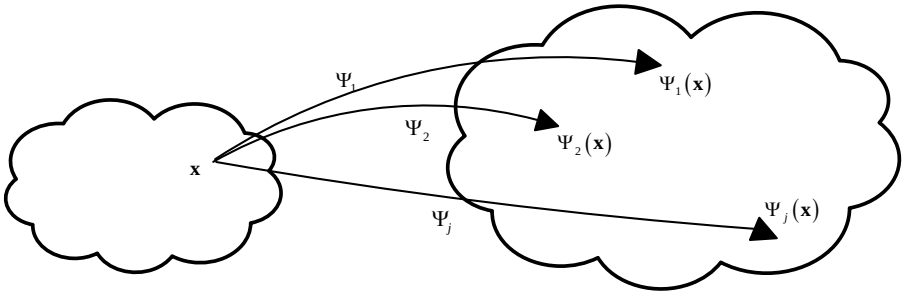


Fig. 1 Prototype in the feature space and its mappings in the CKS (**x** denotes the prototype in original feature space, and $\Psi_j(\mathbf{x})$ denotes the *j*th mapping of **x** from original feature space to CKS)

The general form of the objective function $J_{CKS\text{-}SSC}$ of soft subspace clustering in CKS (CKS-SSC) is given by

$$J_{CKS-SSC}(\mathbf{U},\mathbf{W},\mathbf{V}) = \sum_{j=1}^{c}\sum_{i=1}^{n} u_{ji}^{m} \sum_{h=1}^{s} w_{jh}^{\alpha} d^2(x_{ih}, z_{jh}) + \mathrm{H}(\mathbf{U},\mathbf{W},\mathbf{V}), \tag{16a}$$

where

$$\begin{gathered} \sum_{j=1}^{c} u_{ji} = 1,\ i=1, 2\ldots, n \\ \sum_{h=1}^{s} w_{jh} = 1,\ w_{jh} \in [0,1],\ j = 1, 2, \ldots, c \\ \sum_{t=1}^{p} v_{jt}^2 = 1,\ v_{jt} \in [-1,1],\ j = 1, 2, \ldots, c \end{gathered} \tag{16b}$$

The first term in $J_{CKS\text{-}SSC}$ is the sum of within-cluster dispersions and the second term is the penalty term which helps to improve the quality of clustering results. For hard clustering, m=1 and $u_{ji}\in\{0,1\}$; for fuzzy clustering, m>1 and $u_{ji}\in[0,1]$. c is the number of clusters and $\alpha\geq 1$. With different H(**U**,**W**,**V**) and α value, different variants of soft subspace fuzzy clustering in CKS can be developed.

Notice that in *CKS-SSC*, the prototype x_{ih} in the feature space is mapped into the CKS by c mappings simultaneously, one mapping for each cluster. This is distinct from existing kernel-based clustering algorithms in which all the prototypes are mapped to the kernel space via one single mapping only. By introducing the distance metric learning mechanism into *CKS-SSC*, the most suitable mapping can be learned for each cluster during the clustering process, which is helpful to improve the clustering performance on the datasets with clusters of different inner structures.

Similar to existing kernelized clustering algorithms [32], there are two general variants of *CKS-SSC*: (i) methods that implicitly leave the prototypes in the kernel space during the clustering process, and (ii) methods that perform an inverse mapping to obtain the prototypes in the original feature space. In this paper, methods of this first kind are referred to as *CKS-SSC-K*, and the distance function $d(x_{ih},z_{jh})$ is given by

$$d(x_{ih}, z_{jh}) = \left\| \Psi_j(x_{ih}) - z_{jh} \right\|. \tag{17}$$

By taking the derivative of Eq.(16) with respect to z_{jh} and setting it to zero, we have

$$\frac{\partial J_{CKS-SSC-K}(\mathbf{U},\mathbf{W},\mathbf{V})}{\partial z_{jh}} = -2w_{jh}^{\alpha}\sum_{t=1}^{n} u_{ji}^{m}\Psi_j(x_{ih}) + 2w_{jh}^{\alpha} z_{jh}\sum_{i=1}^{n} u_{ji}^{m} = 0\,.$$

Given a fixed fuzzy partition matrix **U**, the cluster center in CKS z_{jh} can be obtained using the following closed-form solution

$$z_{jh} = \frac{\sum_{i=1}^{n} u_{ji}^{m}\Psi_j(x_{ih})}{\sum_{i=1}^{n} u_{ji}^{m}} = \frac{\sum_{i=1}^{n} u_{ji}^{m}\left(\sum_{t=1}^{p} v_{jt}\Phi_t(x_{ih})\right)}{\sum_{i=1}^{n} u_{ji}^{m}} = \sum_{t=1}^{p} v_{jt}\left(\frac{\sum_{i=1}^{n} u_{ji}^{m}\Phi_t(x_{ih})}{\sum_{i=1}^{n} u_{ji}^{m}}\right). \tag{18}$$

Substituting Eq.(18) into Eq.(17), we have:

$$d(x_{ih}, z_{jh}) = \left\| \Psi_j(x_{ih}) - z_{jh} \right\| = \left\| \sum_{t=1}^{p} v_{jt}\Phi_t(x_{ih}) - \sum_{t=1}^{p} v_{jt}\left(\frac{\sum_{i=1}^{n} u_{ji}^{m}\Phi_t(x_{ih})}{\sum_{i=1}^{n} u_{ji}^{m}}\right)\right\| = \left\| \sum_{t=1}^{p} v_{jt}\left(\Phi_t(x_{ih}) - \frac{\sum_{i=1}^{n} u_{ji}^{m}\Phi_t(x_{ih})}{\sum_{i=1}^{n} u_{ji}^{m}}\right)\right\|.$$

After further derivation, $d^2(x_{ih}, z_{jh})$, the squared distance between x_{ih} and z_{jh} in the CKS, is given by

$$d^2(x_{ih}, z_{jh}) = \left\| \Psi_j(x_{ih}) - z_{jh} \right\|^2 = \left(\Psi_j(x_{ih}) - z_{jh}\right)^T\left(\Psi_j(x_{ih}) - z_{jh}\right) = \sum_{t=1}^{p} v_{jt}^2 e_{jih}^{(t)}\,, \tag{19}$$

where

$$e_{jih}^{(t)} = \left\| \Phi_t(x_{ih}) - \frac{\sum_{i=1}^{n} u_{ji}^m \Phi_t(x_{ih})}{\sum_{i=1}^{n} u_{ji}^m} \right\|^2 = \left(\Phi_t(x_{ih}) - \frac{\sum_{i=1}^{n} u_{ji}^m \Phi_t(x_{ih})}{\sum_{i=1}^{n} u_{ji}^m} \right)^T \left(\Phi_t(x_{ih}) - \frac{\sum_{i=1}^{n} u_{ji}^m \Phi_t(x_{ih})}{\sum_{i=1}^{n} u_{ji}^m} \right)$$

$$= K_t(x_{ih}, x_{ih}) - 2\frac{\sum_{k=1}^{n} u_{jk}^m K_t(x_{kh}, x_{ih})}{\sum_{i=1}^{n} u_{ik}^m} + \frac{\sum_{i1=1}^{n}\sum_{i2=1}^{n} u_{j,i1}^m u_{j,i2}^m K_t(x_{i1,h}, x_{i2,h})}{\left(\sum_{i=1}^{n} u_{ji}^m\right)^2} \tag{20}$$

denotes the squared distance between x_{ih} and z_{jh} in the kernel space induced by the implicit mapping ϕ_t.

On the other hand, the second kind of methods retains the prototypes in the feature space during clustering process. The methods are referred to as *CKS-SSC-F*. In the algorithms, $d(x_{ih}, z_{jh})$ can be evaluated by

$$d(x_{ih}, z_{jh}) = \left\| \Psi_j(x_{ih}) - \Psi_j(z_{jh}) \right\|. \tag{21}$$

Accordingly, the squared distance in kernel space is computed by

$$d^2(x_{ih}, z_{jh}) = \left\| \Psi_j(x_{ih}) - \Psi_j(z_{jh}) \right\|^2 = \left(\Psi_j(x_{ih}) - \Psi_j(z_{jh}) \right)^T \left(\Psi_j(x_{ih}) - \Psi_j(z_{jh}) \right) = \sum_{t=1}^{p} v_{jt}{}^2 e_{jih}^{(t)},$$

where

$$e_{jih}^{(t)} = K_t(x_{ih}, x_{ih}) + K_t(x_{ih}, z_{jh}) - 2K_t(x_{ih}, z_{jh}). \tag{22}$$

Here, consider the Gaussian kernel which is commonly used in the literatures, i.e.

$$K_t(x_{ih}, x_{jh}) = \exp\left(-\frac{\left(x_{ih} - x_{jh}\right)^2}{2\sigma_t^2} \right), t = 1, 2, ..., p,$$

then

$$K_t(x_{ih}, x_{ih}) = K_t(z_{jh}, z_{jh}) = 1$$

and

$$e_{jih}^{(t)} = 2\left(1 - K_t(x_{ih}, z_{jh})\right).$$

By taking the derivative of Eq.(16) with respect to z_{jh} and setting it to zero, we have

$$\frac{\partial J_{CKS-SSC-F}(\mathbf{U}, \mathbf{W}, \mathbf{V})}{\partial z_{jh}} = 2\sum_{i=1}^{n} u_{ji}{}^m w_{jh} \sum_{t=1}^{p} v_{jt}{}^2 \frac{\partial\left(1 - K_t(x_{ih}, z_{jh})\right)}{\partial z_{jh}} = 0$$

which leads to the expression

$$z_{jh} = \sum_{i=1}^{n} u_{ji}{}^m x_{ih} \sum_{t=1}^{p} v_{jt}{}^2 K_t(x_{ih}, z_{jh}) \Big/ \sum_{i=1}^{n} u_{ji}{}^m \sum_{t=1}^{p} v_{jt}{}^2 K_t(x_{ih}, z_{jh}).$$

It can be seen from both Eq.(19) and Eq.(22) that $d^2(x_{ih},z_{jh})$ is determined by the sum of $e_{jih}^{(t)}$ weighted by v_{jt}^2. This provides the facility for developing soft subspace clustering algorithms in the CKS. Also, unsupervised distance metric learning in CKS is achieved since $d(x_{ih},z_{jh})$ can be determined adaptively if v_{jt}^2 are updated automatically during the clustering process.

3.2 The CKS-EWFC-K algorithm

By incorporating the mechanism of distance metric learning into the framework of entropy weighting subspace clustering, CKS-EWFC-K is proposed based on the framework of *CKS-SSC-K*. The objective function $J_{CKS\text{-}EWFC\text{-}K}$ is given by

$$J_{CKS-EWFC-K}(\mathbf{U},\mathbf{W},\mathbf{V}) = \sum_{j=1}^{c}\sum_{i=1}^{n} u_{ji}^{m}\sum_{h=1}^{s} w_{jh} d^2(x_{ih},z_{jh}) + \gamma\sum_{j=1}^{c}\sum_{t=1}^{p} v_{jt}^2 \log v_{jt}^2 + \eta\sum_{j=1}^{c}\sum_{h=1}^{s} w_{jh}\log w_{jh}\ , \tag{23a}$$

subject to

$$\begin{gathered}\textstyle\sum_{j=1}^{c} u_{ji} = 1,\ u_{ji}\in[0,1],\ \ m>1\\ \textstyle\sum_{h=1}^{s} w_{jh} = 1, w_{jh}\in[0,1],\ j = 1, 2, \ldots, c\\ \textstyle\sum_{t=1}^{p} v_{jt}^2 = 1, j = 1, 2, \ldots, c\end{gathered} \tag{23b}$$

where $d^2(x_{ih},z_{jh})$ is evaluated with Eq.(19), $\mathbf{W} = [w_{jh}]$ is a $c\times s$ feature weight matrix, $\mathbf{V}=[v_{jt}]$ is a $c\times p$ kernel weights matrix and $\mathbf{U}=[u_{ji}]$ is the fuzzy partition matrix. Note that v_{jt} can be interpreted as the weight of the tth orthogonal base $\Phi_t(\cdot)$ in the jth mapping $\Psi_j(\mathbf{x})$. It can take both positive and negative values. Similar with the role of $\sum_{j=1}^{c}\sum_{h=1}^{s} w_{jh}\log w_{jh}$ in entropy weighting subspace clustering algorithms like EWKM [13] and ESSC [10], the penalty term $\sum_{j=1}^{c}\sum_{t=1}^{p} v_{jt}^2\log v_{jt}^2$ in Eq.(23a) is introduced to control the kernel weights in each cluster so that they can be optimized during the clustering process. Comparing Eq.(23) with Eq.(16), it can be seen that CKS-EWFC-K is a special case of *CKS-SSC* when H(**U**,**W**,**V**) takes the form of $\gamma\sum_{j=1}^{c}\sum_{t=1}^{p} v_{jt}^2\log v_{jt}^2 + \eta\sum_{j=1}^{c}\sum_{h=1}^{s} w_{jh}\log w_{jh}$ and α=1.

The main idea of CKS-EWFC-K is to minimize the sum of the within-cluster dispersions and the negative weight entropy in Eq.(23). It contains three terms, the within-cluster compactness in the kernel space, the negative entropy of both feature weights and the kernel weights. The positive parameters γ and η are used to control the influences of the entropy of both w_{jh} and v_{jt}.

The minimization of the objective function in Eq.(23) with the constraints is essentially a class of constrained nonlinear optimization problems. The usual strategy to optimize $J_{CKS\text{-}EWFC\text{-}K}$ is to achieve partial optimization for **U**, **W** and **V**, which can be achieved iteratively by solving the three minimization problems below:

1. Problem P1: Fix **U** and **W**, solve the reduced problem $J_{CKS-EWFC-K}(\tilde{\mathbf{U}},\tilde{\mathbf{W}},\mathbf{V})$;

2. Problem P2: Fix **U** and **V**, solve the reduced problem $J_{CKS-EWFC-K}(\tilde{\mathbf{U}},\mathbf{W},\tilde{\mathbf{V}})$;

3. Problem P3: Fix **W** and **V**, solve the reduced problem $J_{CKS-EWFC-K}\left(\mathbf{U},\tilde{\mathbf{W}},\tilde{\mathbf{V}}\right)$.

Problem P1 is solved by

$$v_{jt}^2 = \exp\left(\frac{-\beta_{jt}}{\gamma}\right)\Big/\sum_{t=1}^{p}\left(\exp\left(\frac{-\beta_{jt}}{\gamma}\right)\right) \tag{24}$$

where $\beta_{jt} = \sum_{i=1}^{n} u_{ji}^m \delta_{jit}$, $\delta_{jit} = \sum_{h=1}^{s} w_{jh} e_{jih}^{(t)}$.

Theorem 1. Given the fixed matrices **U** and **W**, $J_{CKS\text{-}EWFC\text{-}K}$ is minimized with **V** computed using Eq.(24).

Proof: Substituting Eq.(19) into Eq.(23), the objective function of CKS-EWFC-K can be rearranged as

$$J_{CKS-EWFC-K}\left(\mathbf{U},\mathbf{W},\mathbf{V}\right) = \sum_{i=1}^{n}\sum_{j=1}^{c}\sum_{t=1}^{p} v_{jt}^2 u_{ji}^m \delta_{jit} + \gamma\sum_{j=1}^{c}\sum_{t=1}^{p} v_{jt}{}^2 \log v_{jt}{}^2 + \eta\sum_{j=1}^{c}\sum_{h=1}^{s} w_{jh}\log w_{jh} ,$$

where $\delta_{jit} = \sum_{h=1}^{s} w_{jh} e_{jih}^{(t)}$.

Using the Lagrange multipliers for the constraint $\sum_{t=1}^{p} v_{jt}^2 = 1$, j=1,2,…,c, the Lagrange function L_λ is given by

$$L_\lambda\left(\mathbf{V},\lambda\right) = \sum_{i=1}^{n}\sum_{j=1}^{c}\sum_{t=1}^{p} v_{jt}^2 u_{ji}^m \delta_{jit} + \gamma\sum_{j=1}^{c}\sum_{t=1}^{p} v_{jt}^2 \log v_{jt}^2 + \eta\sum_{j=1}^{c}\sum_{h=1}^{s} w_{jh}\log w_{jh} - \sum_{j=1}^{c}\lambda_j\left(\sum_{t=1}^{p} v_{jt}^2 - 1\right). \tag{25}$$

For ease of description, we denote $J_i = \sum_{j=1}^{c}\sum_{t=1}^{p} v_{jt}^2 u_{ji}^m \delta_{jit}$. By setting the gradient of Eq.(25) with respect to $v_{jt}{}^2$ and λ_j to zero, we obtain

$$\frac{\partial L_\lambda}{\partial v_{jt}^2} = \sum_{i=1}^{n}\frac{\partial J_i}{\partial v_{jt}^2} + \gamma\frac{\partial\left(\sum_{j=1}^{c}\sum_{t=1}^{p} v_{jt}^2 \log v_{jt}^2\right)}{\partial v_{jt}^2} - \lambda_j = 0 \tag{26}$$

and

$$\frac{\partial L_\lambda}{\partial \lambda_j} = \sum_{t=1}^{p} v_{jt}^2 - 1 = 0 . \tag{27}$$

From Eq.(26), we have

$$\beta_{jt} + \gamma\left(1 + \log v_{jt}^2\right) - \lambda_j = 0 ,$$

where $\beta_{jt} = \sum_{i=1}^{n} u_{ji}{}^m \delta_{jit}$, $\delta_{jit} = \sum_{h=1}^{s} w_{jh} e_{jih}^{(t)}$. By substituting it into Eq.(27), λ_j is eliminated and the closed-form solution for $v_{jt}{}^2$ is obtained, i.e. $v_{jt}^2 = \exp\left(-\beta_{jt}/\gamma\right)\Big/\sum_{t=1}^{p}\exp\left(-\beta_{jt}/\gamma\right)$. Theorem 1 is thus proved. □

Problem P2 is solved by

$$w_{jh} = \exp\left(-\alpha_{jh}/\eta\right)\Big/\sum_{h=1}^{s}\left(\exp\left(-\alpha_{jh}/\eta\right)\right) \tag{28}$$

where $\alpha_{jh} = \sum_{i=1}^{n} u_{ji}^{m} \left(x_{ih} - z_{jh} \right)^2$.

Theorem 2. Given fixed matrices **U** and **V**, $J_{CKS\text{-}EWFC\text{-}K}$ is minimized with **W** computed using Eq.(28).

Proof: The objective function defined in Eq.(23) can be rearranged as

$$J_{CKS-EWFC-K}\left(\mathbf{U},\mathbf{W},\mathbf{V}\right) = \sum_{j=1}^{c}\sum_{h=1}^{s} w_{jh} \sum_{i=1}^{n} u_{ji}^{m} d^2(x_{ih}, z_{jh}) + \gamma \sum_{j=1}^{c}\sum_{t=1}^{p} v_{jt}^2 \log v_{jt}^2 + \eta \sum_{j=1}^{c}\sum_{h=1}^{s} w_{jh} \log w_{jh}$$

Using the Lagrange multipliers for the constraint $\sum_{h=1}^{s} w_{jh} = 1$, j=1,2,…,c, the Lagrange function L_δ is given by

$$L_\delta\left(\mathbf{W},\delta\right) = J_{CKS-EWFC-K}\left(\mathbf{U},\mathbf{W},\mathbf{V}\right) - \sum_{j=1}^{c} \delta_j \left(\sum_{h=1}^{s} w_{jh} - 1 \right). \tag{29}$$

By setting the gradient of Eq.(29) with respect to w_{jh} and δ_j to zero, we have

$$\frac{\partial L_\delta}{\partial w_{jh}} = \frac{\partial J}{\partial w_{jh}} - \delta_j = \sum_{i=1}^{n} u_{ji}^{m} d^2(x_{ih}, z_{jh}) + \eta\left(1 + \log w_{jh}\right) - \delta_j = 0 \tag{30}$$

and

$$\frac{\partial L_\delta}{\partial \delta_j} = \sum_{h=1}^{s} w_{jh} - 1 = 0 . \tag{31}$$

From Eq.(30), we have

$$w_{jh} = \exp\left(\delta_j / \eta\right) \exp\left(-\left(\sum_{i=1}^{n} u_{ji}{}^{m} d^2(x_{ih}, z_{jh}) + \eta \right) \Big/ \eta \right)$$

By substituting it into Eq.(31), δ_j is eliminated and the closed-form solution for optimal w_{jh} is obtained, i.e. $w_{jh} = \exp\left(-\alpha_{jh}/\eta\right) \Big/ \sum_{h=1}^{s} \left(\exp\left(-\alpha_{jh}/\eta\right) \right)$ with $\alpha_{jh} = \sum_{i=1}^{n} u_{ji}^{m} d^2(x_{ih}, z_{jh})$. Theorem 2 is thus proved. □

Problem P3 is solved by

$$u_{ji} = d_{ji}^{-\frac{2}{m-1}} \Big/ \sum_{r=1}^{c} d_{ri}^{-\frac{2}{m-1}} \tag{32}$$

in which d_{ji} is computed with $d_{ji}^2 = \sum_{h=1}^{s} w_{jh} d^2(x_{ih}, z_{jh})$.

Theorem 3. Given fixed matrices **W** and **V**, $J_{CKS\text{-}EWFC\text{-}K}$ is minimized with **U** computed using Eq.(32).

Proof: Denoting $d_{ji}^2 = \sum_{h=1}^{s} w_{jh} d^2(x_{ih}, z_{jh})$, the objective function defined in Eq.(23) can be rearranged as

$$J_{CKS-EWFC-K}\left(\mathbf{U},\mathbf{W},\mathbf{V}\right) = \sum_{j=1}^{c}\sum_{i=1}^{n} u_{ji}^{m} d_{ji}^2 + \gamma \sum_{j=1}^{c}\sum_{t=1}^{p} v_{jt}^2 \log v_{jt}^2 + \eta \sum_{j=1}^{c}\sum_{h=1}^{s} w_{jh} \log w_{jh}$$

Using the Lagrange multipliers for the constraint $\sum_{j=1}^{c} u_{ji} = 1$, i=1,2,…,n, the Lagrange function L_ς is given by

$$L_{\zeta}\left(\mathbf{U},\zeta\right)=\sum_{j=1}^{c}\sum_{i=1}^{n}u_{ji}^{m}d_{ji}^{2}+\gamma\sum_{j=1}^{c}\sum_{t=1}^{p}v_{jt}^{2}\log v_{jt}^{2}+\eta\sum_{j=1}^{c}\sum_{h=1}^{s}w_{jh}\log w_{jh}-\sum_{i=1}^{n}\zeta_{i}\left(\sum_{j=1}^{c}u_{ji}-1\right). \tag{33}$$

By setting the gradient of Eq.(33) with respect to u_{ji} and ζ_i to zero, we have

$$\frac{\partial L_{\zeta}}{\partial u_{ji}}=mu_{ji}^{m-1}d_{ji}^{2}-\zeta_{i}=0 \tag{34}$$

and

$$\frac{\partial L_{\zeta}}{\partial \zeta_{i}}=\sum\nolimits_{j=1}^{c}u_{ji}-1=0\ . \tag{35}$$

By substituting Eq.(34) into Eq.(35), ζ_i is eliminated and the closed-form solution for u_{ji} is obtained, i.e. $u_{ji}=d_{ji}^{-\frac{2}{m-1}}\Big/\sum_{r=1}^{c}d_{ri}^{-\frac{2}{m-1}}$

. Theorem 3 is thus proved. □

As the cluster centers are in the CKS and it is unlikely to evaluation of the centers directly, the theorems discussed above enable the elimination of the cluster centers from the objective function so that the objective function of CKS-EWFC-K defined in Eq.(23) can be readily optimized. Furthermore, the proposed CKS-EWFC-K algorithm can also be utilized for applications involving relational data.

The CKS-EWFC-K algorithm is summarized in Table 3. It starts by initializing the memberships and feature weights, followed by repeated updating of the kernel weights by fixing the memberships and feature members, until the number of changes per iteration in the membership matrix falls below a given threshold. That is, the objective function Eq.(23) is minimized interactively according to Theorems 1 to 3. Suppose in the qth iteration where partial minimization is achieved, the following relationship holds,

$$J(\mathbf{U}^{(q+1)},\mathbf{W}^{(q+1)},\mathbf{V}^{(q+1)})\le J(\mathbf{U}^{(q)},\mathbf{W}^{(q+1)},\mathbf{V}^{(q+1)})\le J(\mathbf{U}^{(q)},\mathbf{W}^{(q)},\mathbf{V}^{(q+1)})\le J(\mathbf{U}^{(q)},\mathbf{W}^{(q)},\mathbf{V}^{(q)}).$$

It implies that $J(\mathbf{U},\mathbf{W},\mathbf{V})$ is a decreasing function with respect to the iteration number q. Therefore, the proposed algorithm CKS-EWFC-K can subsequently converges to either a local optimal solution or a saddle point of the objective function.

Table 3 The pseudo-code of the CKS-EWFC-K algorithm

Algorithm 1. CKS-EWFC-K
Input: D—the dataset, c—the number of clusters, q—the iteration number, s—the number of features, ε—threshold for determination
1: Randomly initialize membership matrix and initialize $\mathbf{W}$ with $w_{jh}=1/s$
2: for $q=1$: *maxIter*
3: Update $\mathbf{V}^{(q+1)}$ according to Eq.(24) with $\mathbf{W}^{(q)}$ and $\mathbf{U}^{(q)}$;
4: Update $\mathbf{W}^{(q+1)}$ according to Eq.(28) with $\mathbf{V}^{(q+1)}$ and $\mathbf{U}^{(q)}$;
5: Update $\mathbf{U}^{(q+1)}$ according to Eq.(32) with $\mathbf{W}^{(q)}$ and $\mathbf{V}^{(q)}$;
6: Calculate the objective function $J(\mathbf{U}^{(q+1)},\mathbf{W}^{(q+1)},\mathbf{V}^{(q+1)})$ with Eq.(23);
7 if $\lvert J(\mathbf{U}^{(q+1)},\mathbf{W}^{(q+1)},\mathbf{V}^{(q+1)})- J(\mathbf{U}^{(q)},\mathbf{W}^{(q)},\mathbf{V}^{(q)})\rvert<\varepsilon$, break;
8: end for
9: Output $\mathbf{W}^{(q+1)}$, $\mathbf{V}^{(q+1)}$ and $\mathbf{U}^{(q+1)}$.

The computational complexity of CKS-EWFC-K per iteration is O(sn^2cp). The space required by the algorithm to store the kernel matrices, cluster centers **V**, feature weight matrix **W**, kernel weight matrix **V** and the partition matrix **U** is O(n^2p), O(cs), O(cs), O(cp) and O(cn) respectively.

3.3 The CKS-EWFC-F algorithm

The CKS-EWFC-F algorithm is developed based on the framework of *CKS-SSC-F* and using the distance function in Eq.(21). The objective function $J_{CKS\text{-}EWFC\text{-}F}$ is given by

$$J_{CKS-EWFC-F}\left(\mathbf{U},\mathbf{W},\mathbf{V},\mathbf{Z}\right)=\sum_{j=1}^{c}\sum_{i=1}^{n}u_{ji}^{m}\sum_{h=1}^{s}w_{jh}d^2(x_{ih},z_{jh})+\gamma\sum_{j=1}^{c}\sum_{t=1}^{p}v_{jt}^2\log v_{jt}^2+\eta\sum_{j=1}^{c}\sum_{h=1}^{s}w_{jh}\log w_{jh} \tag{36a}$$

subject to

$$\begin{gathered}\sum_{j=1}^{c}u_{ji}=1,\ u_{ji}\in[0,1],\ \ m>1\\ \sum_{h=1}^{s}w_{jh}=1,\ w_{jh}\in[0,1],\ j=1,2,\ldots,c\\ \sum_{t=1}^{p}v_{jt}^2=1,\ j=1,2,\ldots,c\end{gathered} \tag{36b}$$

where $d^2(x_{ih},z_{jh})$ is evaluated with Eq.(21). Similar to CKS-EWFC-K, to minimize the objective function $J_{CKS\text{-}EWFC\text{-}F}$, it is necessary to satisfy the three conditions below:

$$v_{jt}^{\ 2}=\exp\left(\frac{-\beta_{jt}}{\gamma}\right)\Big/\sum_{t=1}^{p}\left(\exp\left(\frac{-\beta_{jt}}{\gamma}\right)\right) \tag{37}$$

where $\beta_{jt}=\sum_{i=1}^{n}u_{ji}^{\ m}\delta_{jit}$, $\delta_{jit}=2\sum_{h=1}^{s}w_{jh}\left(1-K_t(x_{ih},z_{jh})\right)$,

$$w_{jh}=\exp\left(\frac{-\alpha_{jh}}{\eta}\right)\Big/\sum_{h=1}^{s}\left(\exp\left(\frac{-\alpha_{jh}}{\eta}\right)\right) \tag{38}$$

where $\alpha_{jh}=\sum_{i=1}^{n}u_{ji}^{\ m}d^2(x_{ih},z_{jh})$, and

$$u_{ji}=d_{ji}^{-\frac{2}{m-1}}\Big/\sum_{r=1}^{c}d_{ri}^{-\frac{2}{m-1}} \tag{39}$$

where d_{ji} is computed with $d_{ji}^2=\sum_{h=1}^{s}w_{jh}d^2(x_{ih},z_{jh})$,

$$z_{jh}=\sum_{i=1}^{n}u_{ji}^{\ m}x_{ih}\sum_{t=1}^{p}v_{jt}^{\ 2}K_t(x_{ih},z_{jh})\Big/\sum_{i=1}^{n}u_{ji}^{\ m}\sum_{t=1}^{p}v_{jt}^{\ 2}K_t(x_{ih},z_{jh}) \tag{40}$$

Note that z_{jh} in Eq.(40) cannot be solved directly and the following strategy is adopted. Let the right-hand side of Eq.(40) be $f(z_{jh})$. The first step here is to specify the initial value $z_{jh}^{(0)}$ and then compute $f(z_{jh}^{(0)})$ and set it to $z_{jh}^{(1)}$. The step is repeated until the (q+1)th solution $z_{jh}^{(q+1)}$ is very close to the qth solution. Thus, if Eq.(37), Eq.(38) and Eq.(39) are used to solve for v_{jt}^2, w_{jh} and u_{ji} respectively, an iterative method is also needed to solve z_{jh} before advancing to next iteration step of the CKS-EWFC-F

algorithm. In practice, however, it is found to be sufficient to use one step to approximate $z_{jh}^{(l)}$ in each iteration step of the CKS-EWFC-F algorithm. Table 4 summarizes the CKS-EWFC-F algorithm.

Table 4 The pseudo-code of the CKS-EWFC-F algorithm

Algorithm 2. CKS-EWFC-F
Input: D—the dataset, *c*—the number of clusters, *q*—the iteration number, *s*—the number of features, ε—threshold for determination 1: Randomly initialize membership matrix and initialize $\mathbf{W}$ with $w_{jh} = 1/s$ 2: for $q = 1$: *maxIter* 3: Update $\mathbf{V}^{(q+1)}$ according to Eq.(37) with $\mathbf{Z}^{(q)}$, $\mathbf{W}^{(q)}$ and $\mathbf{U}^{(q)}$; 4: Update $\mathbf{Z}^{(q+1)}$ according to Eq.(40) with $\mathbf{U}^{(q)}$ and $\mathbf{V}^{(q+1)}$; 5: Update $\mathbf{W}^{(q+1)}$ according to Eq.(38) with $\mathbf{U}^{(q)}$, $\mathbf{V}^{(q+1)}$ and $\mathbf{Z}^{(q+1)}$; 6: Update $\mathbf{U}^{(q+1)}$ according to Eq.(39) with $\mathbf{V}^{(q+1)}$, $\mathbf{W}^{(q+1)}$ and $\mathbf{Z}^{(q+1)}$; 7: Calculate the objective function $J(\mathbf{U}^{(q+1)}, \mathbf{W}^{(q+1)}, \mathbf{V}^{(q+1)}, \mathbf{Z}^{(q+1)})$ with Eq.(36); 8 if $\lvert J(\mathbf{U}^{(q+1)}, \mathbf{W}^{(q+1)}, \mathbf{V}^{(q+1)}, \mathbf{Z}^{(q+1)}) - J(\mathbf{U}^{(q)}, \mathbf{W}^{(q)}, \mathbf{V}^{(q)}, \mathbf{Z}^{(q)})\rvert < \varepsilon$, break; 9: end for 10: Output $\mathbf{W}^{(q+1)}$, $\mathbf{V}^{(q+1)}$ and $\mathbf{U}^{(q+1)}$.

Using a strategy similar to that in Section 3.2, we can prove that CKS-EWFC-F is also convergent and the proof is not shown here for the sake of space. Instead, we will investigate the robustness of CKS-EWFC-F to noise or outliers which often exist in real-world applications. The utilization of the distance function in Eq.(21) can improve the robustness since noise or outliers influence the centers of clusters. We will show that according to influence function analysis, the estimator resulting from Eq. (36) is an M-estimator respect to $\mathbf{z}$ which is robust to noise and outliers.

Let $\{x_1, \ldots, x_n\}$ be a dataset of interest and θ be an unknown parameter to be estimated. The M-estimator uses a suitable symmetric positive-definite function called the *robust-loss function* and the objective function is constructed by summing the loss over all data points. Thus, in the M-estimator approach, the objective function $J(\theta)$ can be written as

$$J(\theta) = \sum_{j=1}^{n} \rho(x_j; \theta), \tag{41}$$

where ρ is an arbitrary function that can measure the loss of x_j and θ. Then, the necessary condition for minimizing Eq.(41) is obtained by setting the derivative of Eq.(41) to zero, i.e.

$$\sum_{j=1}^{n} \varphi(x_j - \theta) = 0, \tag{42}$$

where $\varphi(x_j - \theta) = \partial\rho(x_j - \theta)/\partial\theta$. Let $w(x_j\text{-}\theta)$ be the weight function defined by

$$\varphi(x_j\text{-}\theta)=(x_j\text{-}\theta)w(x_j\text{-}\theta).$$

By substituting it into Eq.(42), the M-estimator can be expressed as the weighted mean

$$\theta = \sum_{j=1}^{n} \frac{w_j(x_j - \theta)}{\sum_{i=1}^{n} w_i(x_j - \theta)} x_j . \tag{43}$$

Given a starting value for θ, the fixed-point iteration or Newton's method can be applied to obtain a solution to Eq.(43) iteratively.

Usually, the influence function or influence curve (IC) can be used to assess the relative influence of individual observations toward the value of an estimate. The influence function of the M-estimator is proportional to the function φ and can be calculated as

$$IC\left(x;F,\theta\right)=\frac{\varphi(x-\theta)}{\int\varphi'(x-\theta)dF_X(x)}, \tag{44}$$

where $F_X(\bullet)$ denotes the distribution function of X. If the influence function of the estimator is unbounded, an outlier might cause problems.

In order to analyze the robustness of CKS-EWFC-F, the first term in Eq.(36) can be rewritten as

$$J_1(\mathbf{U},\mathbf{W},\mathbf{V}) = \sum_{h=1}^{c}\sum_{h=1}^{s}\sum_{i=1}^{n}u_{ji}{}^{m}w_{jh}\sum_{t=1}^{p}v_{jt}{}^{2}\left(1-K_t(x_{ih},z_{jh})\right). \tag{45}$$

Accordingly, the estimate of $\mathbf{z}$ resulting from Eq.(45) is an M-estimator with

$$\rho(x\text{-}z) = \sum_{j=1}^{c}\sum_{h=1}^{s}u_{ji}^{m}w_{jh}\sum_{t=1}^{p}v_{jt}^{2}\left(1-K_t(x_{ih},z_{jh})\right) \tag{46}$$

and

$$\varphi\left(x-z\right)= u_{ji}^{m}w_{jh}\left(x_{ih}-z_{jh}\right)\sum_{t=1}^{p}v_{jt}^{2}\frac{K_t(x_{ih},z_{jh})}{\sigma_t{}^{2}} = \sum_{t=1}^{p}\frac{v_{jt}^{2}u_{ji}{}^{m}w_{jh}\left(x_{ih}-z_{jh}\right)}{\sigma_t{}^{2}\exp\left(\left(x_{ih}-z_{jh}\right)^{2}/2\sigma_t{}^{2}\right)}. \tag{47}$$

By applying the L'Hospital's rule, we have

$$\lim_{x\to\infty}\varphi\left(x-z\right)=-\lim_{x\to\infty}\left(\sum_{t=1}^{p}v_{jt}^{2}u_{ji}^{m}w_{jh}\left(x_{ih}-z_{jh}\right)\Big/\sigma_t{}^{2}\exp\left(\frac{\left(x_{ih}-z_{jh}\right)^{2}}{2\sigma_t{}^{2}}\right)\right)=0 \tag{48}$$

On the other hand, we can easily obtain the maximum and minimum values of $\varphi(x\text{-}z)$ by solving $\partial\varphi\left(x-\theta\right)/\partial x=0$. Thus, the function $\varphi(x\text{-}z)$ computed using Eq.(47) is bounded and continuous, which implies that our new estimator has a bounded and continuous influence function, with finite gross error sensitivity. Hence, CKS-EWFC-F is robust to noise or outliers.

3.4 Connection with other clustering algorithms

In the proposed methods, distance metric learning is achieved with all the data compared in the CKS. The CKS is constructed by combining several kernel spaces and the knowledge from different feature spaces is integrated so that the clustering algorithm can learn the most suitable combination of feature spaces through an iterative process.

From another perspective, both CKS-EWFC-K and CKS-EWFC-F can be regarded as multiple-kernel extensions of EWKM [13], which is one of the most popular entropy weighting soft subspace clustering algorithms. Besides, CKS-EWFC-K can also be regarded as a soft subspace extension of multiple kernel fuzzy clustering (MKFC) [33].

Both CKS-EWFC-K and CKS-EWFC-F can be readily extended to handle multi-view data. In multi-view clustering, a dataset is partitioned into groups by considering multiple views simultaneously during the clustering process. In order to extend the two algorithms to handle multi-view data, each feature in the algorithms can be extended to a feature group which is interpreted as features in different views. In each view, evaluation of the similarities between the data points within each view can be performed in the CKS. The features in the same group share an identical weight. The development of multiple-view clustering algorithms in CKS is an important future work of our research.

Similar to the argument that *k*-means is a special case of FCM, both CKS-EWFC-K and CKS-EWFC-F algorithms are reducible to special cases depending on the value of u_{ji} as follows:

$$u_{ji} = \begin{cases} 1 & \text{if } d_{ji} \le d_{ri},\ r=1,2,...,c \\ 0 & \text{otherwise} \end{cases}$$

Hence, both algorithms can be reduced to a soft subspace clustering algorithm based on hard partition. To the best of our knowledge, there is no previous study attempting to extend EWKM to the CKS version.

3.5 Parameter setting

Like most FCM-based clustering algorithms, it is necessary to set the fuzzy index *m* appropriately in both CKS-EWFC-K and CKS-EWFC-F. The results from a considerable number of experiments have shown empirically that the appropriate range is $m \in [1.05, 1.2]$. It is also necessary to set γ and η appropriately. In our experiments, satisfactory results can be obtained for both algorithms with γ between 1 and 1000, and η between 1 and 10000. However, for a specific application, the tuning and choice of the values for γ and η is dependent on the domain knowledge which is always unavailable.

4. EXPERIMENTS

The proposed CKS-EWFC-K and CKS-EWFC-F algorithms were evaluated with a large number of experiments on real datasets of different complexities. The clustering results were compared with those obtained using several classical soft subspace clustering algorithms. All the experiments were implemented on a computer with an Intel i5-3230M CPU and 4GM RAM.

4.1 Performance metrics and settings

To evaluate the quality of clustering results, two measures, i.e. the rand index (RI) [34] and the normalized mutual information (NMI) [20], were used for evaluating the quality of clustering results.

Let R be a *reference partition* containing *m* classes, and Q be the *hard partition* containing K clusters given by a clustering algorithm. Furthermore, suppose (i) the number of pairs of data objects belonging to a same class in R and to a same cluster in

Q is *a*; (ii) the number of pairs of data objects belonging to a same class in R but to a different cluster in Q is *b*; (iii) the number of pairs of data objects belonging to a different class in R but to a same cluster in Q is *c*; and (iv) the number of pairs of data objects belonging to different classes in R and to a different cluster in Q is *d*, RI is computed as

$$RI = \frac{a+d}{a+b+c+d}. \tag{49}$$

Let n be the total number of data samples, n_i be the number of data samples in class i, n_j be the number of samples in cluster j, and $n_{i,j}$ be the number of samples in class i and cluster j. NMI is then defined as

$$NMI = \sum_{i,j} n_{i,j} \log\left(\frac{n \cdot n_{i,j}}{n_i \cdot n_j}\right) \Bigg/ \sqrt{\left(\sum_i n_i \log\frac{n_i}{n}\right)\left(\sum_j n_j \log\frac{n_j}{n}\right)} \tag{50}$$

Obviously, NMI is equal to 1 when the clustering results perfectly match the external category labels, and close to 0 for random partitioning.

Furthermore, paired t-test was used to check whether the average difference in the performance, in terms of RI and NMI, between CKS-EWFC-K and its rivals is statistically significant. The smaller the p-value, the more significant the difference. A p-value of 0.05 or less is usually considered statistically significant [35].

As discussed in Section 3.1, the CKS is constructed using multiple kernel functions (essentially a similarity measure for pairs of data) and can be used in many different ways. Similarly, different sets of kernel mappings can also be constructed. There are two common approaches to construct the kernel functions. First, given a set of representative vectors for data items, one can employ a number of reproducible kernel functions in the Hilbert space for the construction of multiple kernels. Second, given a set of raw data, different types of feature vectors can be extracted. These feature vectors often correspond to different cues and the similarities can be measured in different feature spaces. The first approach was adopted in this paper to construct the kernel functions for the experiments using the UCI dataset.

Basis kernel mappings with different kernel functions were constructed to evaluate the distance between the data items on each feature in the CKS. To determine the optimal distance for a particular feature, a set of reasonable kernels frequently used by kernel methods were selected for our experiments, including a polynomial kernel with θ=1 and p=2, a linear kernel, and seven Gaussian kernels whose bandwidths are log(0.1), log(0.05), log(0.01), log(0.005), log(0.001), log(0.0005) and log(0.0001) respectively. Table 5 and Table 6 give the details of the basis kernels used for CKS-EWFC-K and CKS-EWFC-F respectively. After generating the kernel matrices for the whole dataset, the values of the elements were normalized to the range of [0, 1].

Table 5 Basis kernel for CKS-EWFC-K

id	Kernel	Parameters settings
K1	Polynomial kernel	d=1, p=2

	$k(\mathbf{x}_1, \mathbf{x}_2) = \left(\mathbf{x}_1^T \mathbf{x}_2 + d\right)^p$	
K2	Gaussian kernel $k(\mathbf{x}_1, \mathbf{x}_2) = \exp\left(-\frac{\|\mathbf{x}_1 - \mathbf{x}_2\|^2}{2\sigma^2}\right)$	σ=log(0.1)
K3		σ=log(0.05)
K4		σ=log(0.01)
K5		σ=log(0.005)
K6		σ=log(0.001)
K7		σ=log(0.0005)
K8		σ=log(0.0001)
K9	Linear kernel $k(\mathbf{x}_1, \mathbf{x}_2) = \mathbf{x}_1^T \mathbf{x}_2$	/

Table 6 Basis kernel for CKS-EWFC-F

id	kernel type	Parameters settings
K2	Gaussian kernel $k(\mathbf{x}_1, \mathbf{x}_2) = \exp\left(-\frac{\|\mathbf{x}_1 - \mathbf{x}_2\|^2}{2\sigma^2}\right)$	σ=log(0.1)
K3		σ=log(0.05)
K4		σ=log(0.01)
K5		σ=log(0.005)
K6		σ=log(0.001)
K7		σ=log(0.0005)
K8		σ=log(0.0001)

4.2 Experiments on UCI datasets

The proposed methods were evaluated with experiments conducted using datasets obtained from the UCI repository. Only the extracted feature vectors are available from the datasets, the raw data are not provided. The datasets are described with a data matrix of "objects × features". The details are shown in Table 7.

Table 7 Details of the UCI datasets

Dataset	Number of instances	Number of features	Number of clusters
Australian	690	15	2
Breast Tissue	106	9	6
Bupa	345	6	2
Heart	270	13	2
Iris	150	4	3
Parkinsons	195	23	2
Pima Indians Diabetes	768	8	2
Vehicle	846	18	4
Wdbc	569	30	2
Wine	178	13	3

In the experiments, the clustering performance of both CKS-EWFC-K and CKS-EWFC-F was compared with that of six classical soft subspace clustering algorithms. Note that the algorithms as KEWFC-K and KEWFC-F in the table refer to the implementation of CKS-EWFC-K and CKS-EWFC-F with one basis kernel only. This will be further discussed in Section 4.3. The parameters of these 10 algorithms and their settings are tabulated in Table 8.

Table 8 Algorithms and the setting of the parameters in the experiments

Algorithms	Parameter setting
CKS-EWFC-K	m=1.05; 1.2 γ = 1;5;10;50;100;1000 η = 1;5;10;50;100;1000;10000
CKS-EWFC-F	m=1.05; 1.2 γ = 1;5;10;50;100;1000 η = 1;5;10;50;100;1000;10000
KEWFC-K	The parameters m, γ and η take values with which CKS-EWFC-K obtained the best clustering result

KEWFC-F	The parameters m, γ and η take values with which CKS-EWFC-F obtained the best clustering result
EWKM [13]	γ = 1e-3; 1e-2; 1e-1; 1e0; 1e1; 1e2; 1e3; 1e4; 1e5; 1e6
FWKM [12]	β = 1.25; 1.5; 1.75; 2.0; 2.25; 2.5; 2.75; 3.0; 3.25; 3.5; 3.75; 4.0; 4.25; 4.5; 4.75; 5.0; 5.25; 5.5; 5.75; 6.0
AWA [8]	α= 1.25; 1.5; 1.75; 2.0; 2.25; 2.5; 2.75; 3.0; 3.25; 3.5; 3.75; 4.0; 4.25; 4.5; 4.75; 5.0; 5.25; 5.5; 5.75; 6.0
LAC [38]	h = 1;1/2; 1/5; 1/10; 1/50; 1/100; 1/1000
FSC[11]	α= 1.1;1.2;1.3;1.4;1.5;1.75;2;2.5;3;3.5;4 ε= 0.0001;0.001;0.01;0.1
MPC [9]	/

To evaluate the clustering algorithms with RI and NMI, the degrees of fuzzy membership were converted to hard assignments by assigning each data to the cluster with the highest degree of membership. In the experiments, under a fixed parameter setting, each algorithm was executed 10 times with different initial partitions. The best results achieved by the 10 algorithms, expressed in terms of RI and NMI, are tabulated in Table 9 and Table 10 respectively. It can be seen from the tables show that no algorithm could give the best result for all the datasets. The overall clustering quality of CKS-EWFC-K was the best although classical soft subspace clustering algorithms outperformed in some cases. Another observation from the experiments is that the clustering performance was unstable for some soft subspace clustering algorithms with a fixed distance function when the algorithm was not properly initialized. For example, when MPC was performed on the datasets *Breast Tissue*, *Parkinsons* and *Pima*, the clusters always merged together if MPC was not initialized properly, which degraded the clustering performance considerably. This was not an issue for CKS-EWFC-K which always gave satisfactory clustering results even if the initialization was suboptimal. The result demonstrates that integrating distance metric learning into soft subspace clustering can guarantee stable clustering quality. Similar conclusions can be drawn for CKS-EWFC-F.

4.3 Distance metric learning in CKS

To assess the advantage of distance metric learning in CKS, the two proposed algorithms CKS-EWFC-K and CKS-EWFC-F were implemented respectively with one basis kernel only to investigate the effect of basis kernels on the clustering performance. The single basis kernel version of CKS-EWFC-K and CKS-EWFC-F are denoted as KEWFC-K and KEWFC-F respectively.

For the sake of space, the details of the experiments conducted with CKS-EWFC-K are only given here. The optimal parameters with which CKS-EWFC-K achieved the best clustering result were first found by grid searching strategy, which were then applied to KEWFC-K. The performance of KEWFC-K in terms of RI and NMI with different basis kernels is shown in Table 11 and Table 12 respectively. The experiments for KEWFC-F were conducted in the same way and the corresponding results are shown in Table 13 and Table 14.

It can be seen from Table 11 and Table 12 that with one fixed basis kernel, the performance of KEWFC-K was generally inferior to that of CKS-EWFC-K. Given a fixed basis kernel, while KEWFC-K was able to give better clustering result for some

datasets, its performance was rather poor in other cases. This illustrates that "one basis kernel fitting all datasets" is not possible because the data items on each feature can have different relationships which could not be evaluated with a single dissimilarity measure. Meanwhile, CKS-EWFC-K could always obtain satisfactory results for different datasets, showing that the development of soft subspace fuzzy clustering in CKS can improve the clustering results by automatically selecting the effective kernels along with each feature. This is advantageous in practice since soft subspace fuzzy clustering in CKS can be utilized without the need to identify the most basis kernel in advance. Another observation from the experiments is that, with the same parameters settings, the best average clustering result of CKS-EWFC-K was always determined by the best results of KEWFC-K over different basis kernels. The result illustrates that the introduction of effective kernel into the basis kernel set can improve the clustering quality of CKS-EWFC-K. Similar conclusions can be obtained from CKS-EWFC-F.

Table 9 Clustering performance in terms of RI

Dataset	Measure	CKS-EWFC-K	CKS-EWFC-F	EWKM	FWKM	AWA	FSC	LAC	MPC
Australian	Mean	0.7183	**0.7204**	0.5977	0.5493	0.5512	0.5027	0.5293	0.6152
	Std	0.0783	**0.0888**	0.0334	0.0000	0.1079	0.0048	0.0103	0.0000
	p-value (CKS-EWFC-K)			2.0575e-03	7.6859e-05	1.7245e-03	1.0378e-05	3.9393e-05	2.4378e-03
	p-value (CKS-EWFC-F)			9.0972e-03	7.6859e-05	2.6668e-03	2.0905e-04	5.5209e-04	1.2236e-02
Breast Tissue	Mean	**0.8100**	0.7457	0.7242	0.6097	0.7637	0.7353	0.7170	0.2928
	Std	**0.0227**	0.0258	0.0240	0.0411	0.0320	0.0235	0.0219	0.2725
	p-value (CKS-EWFC-K)			1.6233e-06	3.2718e-07	4.0497e-04	1.7193e-04	8.3489e-07	1.9488e-04
	p-value (CKS-EWFC-F)			7.3438e-02	3.2718e-07	0.2757	0.4086	1.8589e-02	7.4513e-04
Bupa	Mean	**0.5154**	0.5047	0.5107	0.4997	0.5034	0.5052	0.5145	0.4987
	Std	**0.0024**	0.0008	0.0004	0.0002	0.0003	0.0017	0.0023	0.0000
	p-value (CKS-EWFC-K)			3.4153e-04	1.0851e-08	1.0607e-07	7.6199e-07	2.3089e-04	3.8346e-09
	p-value (CKS-EWFC-F)			2.3321e-06	1.0851e-08	3.8450e-02	0.3931	1.0124e-09	5.9967e-07
Heart	Mean	**0.6971**	0.6816	0.6579	0.5768	0.5606	0.5407	0.6724	0.5826
	Std	**0.0000**	0.0088	0.0640	0.0060	0.0601	0.0662	0.0487	0.0906
	p-value (CKS-EWFC-K)			8.4682e-02	3.1771e-13	5.1600e-05	3.8109e-05	0.1439	3.1266e-03
	p-value (CKS-EWFC-F)			0.3034	3.1771e-13	1.7295e-04	1.1248e-04	0.6270	5.7254e-03
Iris	Mean	0.9267	0.8737	0.8667	0.9003	**0.9464**	0.9381	0.8622	0.8889
	Std	0.0000	0.0000	0.0112	0.0207	**0.0040**	0.0241	0.0739	0.0614
	p-value (CKS-EWFC-K)			3.9787e-08	2.9472e-03	**8.4857e-08**	0.1709	2.2130e-02	8.3453e-02
	p-value (CKS-EWFC-F)			7.8325e-02	2.9472e-03	**7.7188e-13**	1.4593e-05	6.3332e-1	0.4539
Parkinsons	Mean	**1.0000**	0.9799	0.6606	0.5632	**1.0000**	0.6280	0.6117	0.6270
	Std	**0.0000**	0.0636	0.1193	0.0089	**0.0000**	0.0022	0.0000	0.0000
	p-value (CKS-EWFC-K)			8.5495e-06	9.6665e-17	NaN	1.6045e-021	0.0000	0.0000
	p-value (CKS-EWFC-F)			2.5712e-05	9.6665e-17	0.3434	2.8247e-08	1.9886e-08	2.8810e-08
Pima Indians Diabetes	Mean	**0.6153**	0.5574	0.5507	0.5388	0.5390	0.5390	0.5444	0.5450
	Std	**0.0012**	0.0050	0.0000	0.0282	0.0000	0.0000	0.0022	0.0000
	p-value (CKS-EWFC-K)			3.6419e-17	1.0173e-05	8.1238e-18	8.1238e-18	8.3332e-16	1.7106e-17
	p-value (CKS-EWFC-F)			3.8989e-03	1.0173e-05	3.9841e-06	3.9841e-06	5.3075e-05	7.5242e-05
Vehicle	Mean	0.6647	0.6641	0.6561	0.6482	**0.6715**	0.6675	0.6471	0.6535
	Std	0.0220	0.0019	0.0096	0.0393	**0.0106**	0.0098	0.0138	0.0222
	p-value (CKS-EWFC-K)			0.2949	0.2593	**0.4114**	0.7394	4.1792e-02	0.3881
	p-value (CKS-EWFC-F)			1.7017e-02	0.2593	**5.8058e-02**	0.3399	3.4618e-03	0.1779
wdbc	Mean	**0.8968**	0.8605	0.8365	0.8365	0.7984	0.7515	0.8423	0.8394
	Std	**0.0000**	0.0031	0.0063	0.0000	0.0502	0.1170	0.0000	0.0000
	p-value (CKS-EWFC-K)			2.1793e-10	0.0000	1.5858e-04	3.4811e-03	0.0000	0.0000
	p-value (CKS-EWFC-F)			1.1396e-08	0.0000	3.0998e-04	1.2923e-03	3.2166e-04	3.9313e-04
Wine	Mean	**0.9461**	0.8964	0.6835	0.7103	0.8582	0.8058	0.8316	0.8899
	Std	**0.0000**	0.0649	0.0061	0.0206	0.0411	0.0316	0.0646	0.0139
	p-value (CKS-EWFC-K)			3.3567e-16	4.6474e-11	8.2594e-05	1.9727e-07	3.3142e-04	4.0710e-05
	p-value (CKS-EWFC-F)			8.1897e-02	4.6474e-11	3.0590e-03	0.1093	4.2088e-02	0.5219

Note: NaN means “Not-a-Number”, which is returned by MATLAB and means that the result is meaningless.

Table 10 Clustering performance in terms of NMI

Dataset	Measure	CKS-EWFC-K	CKS-EWFC-F	EWKM	FWKM	AWA	FSC	LAC	MPC
Australian	Mean	**0.3686**	0.3744	0.1425	0.1043	0.0909	0.0169	0.1294	0.1693
	Std	**0.1345**	0.1512	0.0496	0.0000	0.1730	0.0095	0.0452	0.0000
	p-value (CKS-EWFC-K)			1.0089e-03	1.5658e-04	1.5197e-03	1.5621e-05	6.3061e-04	1.1453e-03
	p-value (CKS-EWFC-F)			4.9048e-03	1.5658e-04	2.5388e-03	2.7267e-04	3.5035e-03	6.3997e-03
Breast Tissue	Mean	**0.4856**	0.2733	0.3189	0.3409	0.3995	0.3179	0.2577	0.1097
	Std	**0.0218**	0.0317	0.0173	0.0261	0.0467	0.0118	0.0127	0.2315
	p-value (CKS-EWFC-K)			1.8098e-04	4.5030e-04	1.1745e-02	3.1117e-05	1.1330e-06	1.9645e-03
	p-value (CKS-EWFC-F)			6.0724e-03	4.5030e-04	1.2560e-04	1.4512e-03	0.2.339	6.2476e-02
Bupa	Mean	**0.0196**	0.0007	0.0105	0.0134	0.0060	0.0102	0.0103	0.0063
	Std	**0.0000**	0.0001	0.0000	0.0000	0.0000	0.0011	0.0020	0.0000
	p-value (CKS-EWFC-K)			0.7420	2.3382e-02	5.7077e-04	0.5400	0.1137	8.7981e-04
	p-value (CKS-EWFC-F)			1.9475e-14	2.3382e-02	3.3855e-12	4.4851e-10	3.4198e-07	2.1692e-12
Heart	Mean	**0.3062**	0.2836	0.2827	0.1118	0.1023	0.0649	0.2828	0.1284
	Std	**0.0000**	0.0130	0.1047	0.0090	0.0867	0.1014	0.0767	0.1386
	p-value (CKS-EWFC-K)			0.4963	1.5395e-13	3.9637e-05	3.5981e-05	0.3605	2.8664e-03
	p-value (CKS-EWFC-F)			0.9800	1.5395e-13	1.2292e-04	1.0146e-04	0.9785	5.0177e-03
Iris	Mean	0.8513	0.7419	0.7416	0.7882	**0.8584**	0.8525	0.7923	0.7979
	Std	0.0000	0.0000	0.0291	0.0292	**0.0074**	0.0246	0.0268	0.0518
	p-value (CKS-EWFC-K)			8.2314e-07	7.7187e-05	**1.4348e-02**	0.8817	6.5652e-05	9.8465e-03
	p-value (CKS-EWFC-F)			0.9689	7.7187e-05	**2.8000e-12**	1.8233e-07	2.1660e-04	7.6477e-03
Parkinsons	Mean	**1.0000**	0.9554	0.3206	0.0905	**1.0000**	0.3059	0.2973	0.0000
	Std	**0.0000**	0.1409	0.0142	0.0611	**0.0000**	0.0000	0.0000	0.0000
	p-value (CKS-EWFC-K)			1.2106e-16	4.4258e-12	NaN	1.7736e-143	1.5866e-143	0.0000
	p-value (CKS-EWFC-F)			2.3173e-07	4.4258e-12	0.3434	1.4477e-07	1.2908e-07	4.9132e-09
Pima Indians Diabetes	Mean	**0.1306**	0.0771	0.0297	0.0619	0.0204	0.0204	0.0313	0.0000
	Std	**0.0021**	0.0476	0.0000	0.0435	0.0000	0.0000	0.0000	0.0000
	p-value (CKS-EWFC-K)			1.0617e-16	5.9774e-04	4.7988e-17	4.7988e-17	1.2260e-16	1.0494e-17
	p-value (CKS-EWFC-F)			5.1118e-04	5.9774e-04	6.8405e-05	6.8405e-05	7.5620e-04	2.6234e-06
Vehicle	Mean	0.1848	0.1881	**0.2001**	0.1949	0.1860	0.1765	0.1720	0.1704
	Std	0.0097	0.0306	**0.0098**	0.0104	0.0278	0.0580	0.0170	0.0320
	p-value (CKS-EWFC-K)			**2.3871e-03**	3.9968e-03	6.1807e-02	0.4138	0.1618	0.2603
	p-value (CKS-EWFC-F)			**7.3708e-06**	3.9968e-03	2.2730e-03	0.1838	1.2629e-02	8.4625e-02
wdbc	Mean	**0.6833**	0.6312	0.5944	0.5638	0.4737	0.3932	0.5808	0.5603
	Std	**0.0000**	0.0050	0.0031	0.0000	0.0929	0.2006	0.0000	0.0000
	p-value (CKS-EWFC-K)			1.2679e-14	0.0000	5.6469e-05	1.3682e-03	0.0000	4.1680e-14
	p-value (CKS-EWFC-F)			3.7287e-09	0.0000	3.5898e-03	6.2230e-03	2.6307e-16	6.3507e-16
Wine	Mean	**0.8464**	0.7465	0.4645	0.4216	0.7097	0.6403	0.6460	0.6682
	Std	**0.0000**	0.0996	0.0021	0.0123	0.0782	0.0579	0.0870	0.0238
	p-value (CKS-EWFC-K)			8.3252e-022	2.2796e-15	3.6492e-04	1.3241e-06	4.6272e-05	1.7715e-02
	p-value (CKS-EWFC-F)			3.7708e-02	2.2796e-15	4.2538e-03	0.1369	8.9099e-02	1.4013e-05

Note: NaN means “Not-a-Number”, which is returned by MATLAB and means that the result is meaningless.

Table 11 Clustering performance of CKS-EWFC-K and KEWFC-K in terms of RI

Dataset	Measure	CKS-EWFC-K	KEWFC-K (CKS-EWFC-K with single basis kernel)								
			K1	K2	K3	K4	K5	K6	K7	K8	K9
Australian	Mean	**0.7183**	0.7169	0.6768	0.6860	0.6992	0.6945	0.6996	0.7036	0.6961	0.6994
	Std	**0.0783**	0.0778	0.0029	0.0017	0.0009	0.0009	0.0008	0.0020	0.0019	0.1050
	p-value		0.3434	0.0127	0.0226	0.0461	0.0359	0.0469	0.3661	0.0397	0.0365
Breast Tissue	Mean	**0.8100**	0.7433	0.7991	0.7910	0.8011	0.8028	0.7990	0.8009	0.8029	0.7722
	Std	**0.0227**	0.0262	0.0194	0.0245	0.0189	0.0208	0.0260	0.0272	0.0225	0.0173
	p-value		1.1700e-04	0.2016	0.0111	0.1133	0.1768	3.3165e-03	0.0352	0.4191	9.9645e-04
Bupa	Mean	**0.5154**	0.5026	0.5037	0.4994	0.5136	0.5096	0.5122	0.5121	0.5113	0.5031
	Std	**0.0024**	0.0000	0.0000	0.0009	0.0025	0.0008	0.0000	0.0003	0.0019	0.0000
	p-value		4.1011e-08	9.0839e-08	4.8043e-09	0.0355	1.9859e-04	2.5709e-03	2.2192e-03	1.1183e-03	5.9795e-08
Heart	Mean	**0.6971**	0.6731	0.6952	0.6888	0.6860	0.6906	0.6901	0.6897	0.6948	0.5850
	Std	**0.0000**	0.0047	0.0044	0.0029	0.0023	0.0032	0.0024	0.0032	0.0025	0.0207
	p-value		5.7607e-08	0.2220	8.5381e-06	1.1820e-07	1.2405e-04	7.9999e-06	4.5024e-05	0.0150	3.5567e-08
Iris	Mean	0.9267	0.8923	0.7735	0.8123	0.8820	0.8882	0.9044	0.8879	0.8843	**0.9495**
	Std	0.0000	0.0000	0.0131	0.0726	0.0730	0.0934	0.0665	0.0627	0.0603	**0.0000**
	p-value		0.0000	3.8095e-11	7.5929e-04	0.0844	0.2250	0.3162	0.0820	0.0531	0.0000
Parkinsons	Mean	**1.0000**	1.0000	0.6021	0.5975	0.5929	0.5929	0.6027	0.6104	0.6095	1.0000
	Std	**0.0000**	0.0000	0.0000	0.0000	0.0000	0.0000	0.0073	0.0160	0.0464	0.0000
	p-value		NaN	5.1806e-144	0.0000	0.0000	0.0000	3.7490e-17	5.2775e-14	7.1559e-10	NaN
Pima Indians Diabetes	Mean	0.6153	**0.6154**	0.5466	0.5466	0.5458	0.5458	0.5465	0.5466	0.5474	0.5592
	Std	0.0012	**0.0004**	0.0000	0.0000	0.0000	0.0000	0.0003	0.0000	0.0000	0.0008
	p-value		0.7762	2.1000e-17	2.1000e-17	1.8934e-17	1.8934e-17	4.7955e-17	2.1000e-17	2.3340e-17	2.2635e-16
Vehicle	Mean	0.6647	0.3698	0.6604	0.6645	0.6622	0.6631	**0.6659**	0.6597	0.6522	0.5977
	Std	0.0220	0.1066	0.0138	0.0127	0.0043	0.0030	**0.0064**	0.0178	0.0212	0.0780
	p-value		1.2384e-05	0.5438	0.9695	0.7275	0.8145	0.8860	0.6426	0.2093	0.0209
wdbc	Mean	**0.8968**	0.8394	0.8905	0.8999	0.8946	0.8968	0.8937	0.8934	0.8955	0.8541
	Std	**0.0000**	0.0000	0.0000	0.0000	0.0015	0.0000	0.0000	0.0027	0.0022	0.0000
	p-value		0.0000	0.0000	0.0000	1.3230e-03	NaN	0.0000	3.2291e-03	0.1046	0.0000
Wine	Mean	**0.9461**	0.8972	0.9114	0.9301	0.9036	0.8855	0.8787	0.8834	0.8779	0.9133
	Std	**0.0000**	0.0000	0.0030	0.0031	0.0000	0.0026	0.0021	0.0064	0.0020	0.0036
	p-value		0.0000	4.2011e-11	5.4523e-08	0.0000	9.2926e-14	3.8626e-15	1.9369e-10	3.1581e-15	3.5946e-10

Note: NaN means “Not-a-Number”, which is returned by MATLAB and means that the result is meaningless.

Table 12 Clustering performance of CKS-EWFC-K and KEWFC-K in terms of NMI

Dataset	Measure	CKS-EWFC-K	KEWFC-K (CKS-EWFC-K with single basis kernel)								
			K1	K2	K3	K4	K5	K6	K7	K8	K9
Australian	Mean	**0.3686**	0.3666	0.2689	0.2848	0.3075	0.2996	0.3081	0.3149	0.3015	0.3394
	Std	**0.1345**	0.1336	0.0048	0.0027	0.0016	0.0015	0.0014	0.0036	0.0033	0.1785
	p-value		0.3644	0.0433	0.0406	0.0786	0.0138	0.0189	0.0236	0.0153	0.3642
Breast Tissue	Mean	0.4856	0.4788	0.3879	0.3824	0.4166	0.4233	0.4229	0.4323	0.4452	**0.5230**
	Std	0.0218	0.0163	0.0518	0.0539	0.0525	0.0595	0.0542	0.0621	0.0603	**0.0120**
	p-value		0.1106	0.0190	3.3523e-03	0.0758	0.1553	0.0280	0.3194	0.9557	0.2154
Bupa	Mean	**0.0196**	0.0000	0.0105	0.0003	0.0094	0.0026	0.0036	0.0033	0.0025	0.0003
	Std	**0.0000**	0.0000	0.0000	0.0003	0.0030	0.0007	0.0000	0.0005	0.0014	0.0001

	p-value		1.0189e-06	0.0742	2.2464e-06	0.0496	3.8051e-05	2.9126e-05	2.0677e-05	4.7842e-06	1.2711e-06
Heart	Mean	**0.3062**	0.2648	0.3016	0.2911	0.2875	0.2959	0.2953	0.2946	0.3038	0.1309
	Std	**0.0000**	0.0075	0.0071	0.0047	0.0039	0.0053	0.0038	0.0050	0.0038	0.0214
	p-value		3.0271e-08	0.0719	3.3098e-06	1.1278e-07	1.6610e-04	7.4844e-06	4.1460e-05	0.0766	9.1922e-10
Iris	Mean	0.8513	0.8058	0.5649	0.6366	0.7767	0.7998	0.8081	0.7828	0.7806	**0.8642**
	Std	0.0000	0.0000	0.0251	0.1205	0.0801	0.0850	0.0588	0.0529	0.0488	**0.0000**
	p-value		0.0000	4.8092e-11	3.1886e-04	0.0163	0.0175	0.0452	2.7012e-03	1.3219e-03	0.0000
Parkinsons	Mean	**1.0000**	1.0000	0.3212	0.3160	0.3109	0.3109	0.3218	0.3303	0.3130	1.0000
	Std	**0.0000**	0.0000	0.0000	0.0000	0.0000	0.0000	0.0082	0.0180	0.0965	0.0000
	p-value		NaN	2.1672e-143	2.0237e-143	1.8930e-143	1.8930e-143	8.4611e-19	1.1916e-15	3.1952e-09	NaN
Pima Indians Diabetes	Mean	0.1306	**0.1381**	0.0096	0.0096	0.0052	0.0052	0.0091	0.0096	0.0139	0.0617
	Std	0.0021	**0.0007**	0.0000	0.0000	0.0000	0.0000	0.0014	0.0000	0.0000	0.0012
	p-value		0.8812	2.0731e-17	2.0731e-17	1.5058e-17	1.5058e-17	3.7446e-16	2.0731e-17	2.8740e-17	6.9406e-15
Vehicle	Mean	**0.1848**	0.1344	0.1496	0.1528	0.1493	0.1493	0.1612	0.1622	0.1489	0.1372
	Std	**0.0097**	0.0475	0.0218	0.0213	0.0076	0.0097	0.0281	0.0291	0.0190	0.0367
	p-value		0.2021	0.3842	0.0522	0.0375	0.0383	0.0729	0.0668	0.0466	0.0213
wdbc	Mean	0.6833	0.5665	0.6678	**0.6914**	0.6776	0.6827	0.6751	0.6744	0.6797	0.5948
	Std	0.0000	0.0000	0.0000	**0.0000**	0.0035	0.0000	0.0000	0.0066	0.0052	0.0000
	p-value		0.0000	0.0000	0.0000	1.7892e-03	0.0000	0.0000	3.6430e-03	0.1232	0.0000
Wine	Mean	**0.8464**	0.7532	0.7639	0.8023	0.7673	0.7328	0.7204	0.7256	0.7090	0.8056
	Std	**0.0000**	0.0000	0.0062	0.0070	0.0000	0.0054	0.0058	0.0108	0.0024	0.0054
	p-value		0.0000	1.1941e-11	8.9298e-09	0.0000	1.8809e-13	1.5331e-13	5.6230e-11	2.7890e-17	1.8232e-09

Note: NaN means “Not-a-Number”, which is returned by MATLAB and means that the result is meaningless.

Table 13 Clustering performance of CKS-EWFC-F and KEWFC-F in terms of RI

Dataset	Measure	CKS-EWFC-F	KEWFC-F (CKS-EWFC-F with single basis kernel)								
			K1	K2	K3	K4	K5	K6	K7	K8	K9
Australian	Mean	0.7204	/	NaN	0.7104	**0.7518**	0.7419	0.7497	0.7081	0.6561	/
	Std	0.0888		NaN	0.0000	**0.0000**	0.0198	0.0000	0.1019	0.1292	
	p-value			NaN	NaN	**NaN**	0.3910	0.0000e+000	8.4362e-04	0.1600	
Breast Tissue	Mean	0.7457	/	**0.7497**	0.7380	0.7370	0.7398	0.7459	0.7473	0.7496	/
	Std	0.0258		**0.0150**	0.0168	0.0209	0.0242	0.0246	0.0278	0.0307	
	p-value			**0.4360**	0.1562	5.5632e-02	2.8708e-03	0.8816	0.3691	0.1630	
Bupa	Mean	**0.5047**	/	0.5043	0.5038	0.5028	0.5027	0.5021	0.5022	0.5019	/
	Std	**0.0008**		0.0000	0.0003	0.0008	0.0006	0.0010	0.0006	0.0004	
	p-value			NaN	0.1048	6.0164e-03	3.8790e-03	2.3148e-03	1.1672e-03	1.9949e-04	
Heart	Mean	**0.6816**	/	NaN	NaN	NaN	NaN	0.6788	0.6744	0.6744	/
	Std	**0.0088**		NaN	NaN	NaN	NaN	0.0000	0.0000	0.0000	
	p-value			NaN	NaN	NaN	NaN	0.3910	0.1075	2.8869e-02	
Iris	Mean	**0.8737**	/	0.8732	0.8737	0.8737	0.8737	0.8737	0.8732	0.8732	/
	Std	**0.0000**		0.0018	0.0000	0.0000	0.0000	0.0000	0.0018	0.0018	
	p-value			0.3434	NaN	NaN	NaN	NaN	0.3434	0.3434	
Parkinsons	Mean	**0.9799**	/	NaN	NaN	0.6846	0.8474	0.9267	0.9199	0.8751	/
	Std	**0.0636**		NaN	NaN	0.0000	0.1414	0.1277	0.1263	0.1481	
	p-value			NaN	NaN	0.0000e+000	5.4716e-02	0.1188	8.3235e-02	3.5778e-02	
Pima Indians Diabetes	Mean	**0.5574**	/	NaN	0.5541	0.5530	0.5537	0.5526	0.5531	0.5534	/
	Std	**0.0050**		NaN	0.0000	0.0007	0.0035	0.0004	0.0014	0.0026	

	p-value			NaN	NaN	5.1242e-04	0.1319	2.2108e-02	3.5276e-02	1.0660e-03	
Vehicle	Mean	**0.6641**	/	0.6615	0.6577	0.6537	0.6611	0.6622	0.6579	0.6413	/
	Std	**0.0019**		0.0103	0.0091	0.0096	0.0047	0.0094	0.0140	0.0186	
	p-value			0.4343	6.4099e-02	1.0825e-02	7.6389e-02	0.5802	0.2203	3.0512e-03	
wdbc	Mean	**0.8605**	/	0.8085	0.8395	0.8139	0.7109	0.6827	0.6791	0.6816	/
	Std	**0.0031**		0.0353	0.0019	0.1113	0.0895	0.0053	0.0081	0.0000	
	p-value			1.3069e-03	1.0843e-07	0.2286	5.1892e-04	1.3108e-14	1.6006e-13	2.0749e-17	
Wine	Mean	**0.8964**	/	0.8303	0.8610	0.9031	0.8616	0.8066	0.7653	0.8624	/
	Std	**0.1649**		0.1144	0.0854	0.1800	0.1697	0.1458	0.1260	0.1635	
	p-value			0.8445	0.1089	0.9802	8.8254e-03	1.6116e-03	2.2083e-04	0.1018	

~~Note: NaN means "Not-a-Number", which is returned by MATLAB and means that the result is meaningless.~~

Table 14 Clustering performance of CKS-EWFC-F and KEWFC-F in terms of NMI

Dataset	Measure	CKS-EWFC-F	KEWFC-K (CKS-EWFC-F with single basis kernel)								
			K1	K2	K3	K4	K5	K6	K7	K8	K9
Australian	Mean	0.3744	/	NaN	0.3570	**0.4279**	0.4032	0.4249	0.3542	0.2658	/
	Std	0.1512		NaN	0.0000	**0.0000**	0.0495	0.0000	0.1732	0.2195	
	p-value			NaN	NaN	**NaN**	0.3910	0.0000e+000	9.9243e-03	0.1621	
Breast Tissue	Mean	0.2733	/	0.2375	0.2400	0.2544	0.2584	0.2714	**0.2835**	0.3010	/
	Std	0.0317		0.0267	0.0277	0.0200	0.0245	0.0368	**0.0381**	0.0423	
	p-value			4.7300e-02	4.0805e-02	7.5387e-02	0.1724	0.8702	**0.1532**	6.5733e-03	
Bupa	Mean	**0.0007**	/	0.0000	0.0000	0.0002	0.0002	0.0005	0.0004	0.0006	/
	Std	**0.0001**		0.0000	0.0000	0.0003	0.0002	0.0006	0.0003	0.0002	
	p-value			NaN	0.2298	0.7978	0.7638	0.2052	0.1816	1.6346e-02	
Heart	Mean	**0.2836**	/	NaN	NaN	NaN	NaN	0.2795	0.2729	0.2729	/
	Std	**0.0130**		NaN	NaN	NaN	NaN	0.0000	0.0000	0.0000	
	p-value			NaN	NaN	NaN	NaN	0.3910	0.1073	2.8754e-02	
Iris	Mean	**0.7419**	/	0.7405	0.7419	0.7419	0.7419	0.7419	0.7405	0.7405	/
	Std	**0.0000**		0.0045	0.0000	0.0000	0.0000	0.0000	0.0045	0.0045	
	p-value			0.3434	NaN	NaN	NaN	NaN	0.3434	0.3434	
Parkinsons	Mean	**0.9554**	/	NaN	NaN	0.4127	0.6774	0.8355	0.8183	0.7418	/
	Std	**0.1409**		NaN	NaN	0.0000	0.2475	0.2233	0.2233	0.2634	
	p-value			NaN	NaN	0.0000e+000	3.7928e-02	4.9168e-02	3.2564e-02	2.0605e-02	
Pima Indians Diabetes	Mean	**0.0771**	/	NaN	0.0526	0.0509	0.0521	0.0503	0.0510	0.0516	/
	Std	**0.0476**		NaN	0.0000	0.0012	0.0061	0.0008	0.0023	0.0048	
	p-value			NaN	NaN	9.0248e-02	0.4051	0.2084	0.2495	0.1391	
Vehicle	Mean	**0.1881**	/	0.1442	0.1363	0.1314	0.1453	0.1576	0.1562	0.1490	/
	Std	**0.0306**		0.0273	0.0179	0.0176	0.0130	0.0268	0.0286	0.0201	
	p-value			0.6475	4.2562e-02	1.3126e-02	0.5576	0.3836	0.4982	0.9156	
wdbc	Mean	**0.6312**	/	0.5466	0.5969	0.5455	0.3616	0.2926	0.2767	0.1201	/
	Std	**0.0050**		0.0509	0.0031	0.1880	0.1421	0.0414	0.0212	0.0000	
	p-value			5.8648e-04	1.0797e-07	0.1926	2.1082e-04	1.1407e-09	2.2919e-12	2.9981e-18	
Wine	Mean	**0.7465**	/	0.6521	0.7128	0.7487	0.6986	0.6371	0.5850	0.6695	/
	Std	**0.1996**		0.1457	0.1118	0.2280	0.2117	0.1952	0.1726	0.1819	
	p-value			0.7573	9.5024e-02	0.9576	3.1216e-02	1.1381e-02	1.5234e-03	8.2630e-02	

~~Note: NaN means "Not-a-Number", which is returned by MATLAB and means that the result is meaningless.~~

4.4 Scalability

Experiments were conducted to investigate the scalability of both CKS-EWFC-K and CKS-EWFC-F with respect to the basis kernel number. The datasets listed in Table 7 were used to test the two algorithms with different number of basis kernels. Since KEWFC-K was a special case of CKS-EWFC-K with one basis kernel only, KEWFC-K was first executed with each basis kernel to evaluate the suitability of CKS-EWFC-K on each dataset. A kernel resulting in higher clustering quality with KEWFC-K indicated that the kernel was more suitable for CKS-EWFC-K. In the experiment, basis kernels were added into the basis kernel set one by one in descending order of suitability for CKS-EWFC-K, i.e. a kernel that is more suitable was added first. Given a basis kernel set, CKS-EWFC-K was executed 10 times on each dataset. Similarly, experiments were also conducted with CKS-EWFC-F in the same way. Table 15 shows the order of addition of the basis into the basis kernel set for different datasets. The parameters used in CKS-EWFC-K and CKS-EWFC-F for different datasets are given in Table 16.

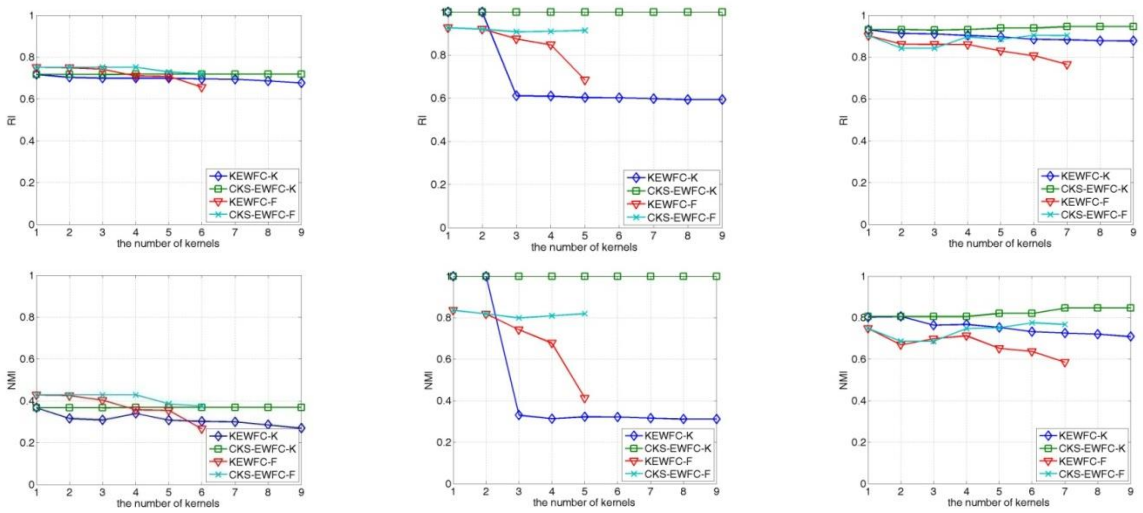
Table 15 details on orders in which the kernels were added into CKS-EWFC-K and CKS-EWFC-F

algorithms	datasets	Order of addition
CKS-EWFC-K	Australian	K1, K7, K6, K9, K4, K8, K5, K3, K2
	Parkinsons	K1, K9, K7, K8, K6, K2, K3, K4, K5
	Wine	K3, K9, K2, K4, K1, K5, K7, K6, K8
CKS-EWFC-F	Australian	K4, K6, K5, K3, K7, K8
	Parkinsons	K6, K7, K8, K5, K4
	Wine	K4, K 8, K5, K 3, K2, K6, K7

Table 16 parameters for CKS-EWFC-K and CKS-EWFC-F on different datasets

algorithms		m	γ	η
CKS-EWFC-K	Australian	1.2	1	10000
	Parkinsons	1.2	1	1000
	Wine	1.2	50	1000
CKS-EWFC-F	Australian	1.2	500	10
	Parkinsons	1.2	1	500
	Wine	1.2	500	5

The clustering performance of CKS-EWFC-K and CKS-EWFC-F versus the number of basis kernels was plotted in Fig. 2, where the performance of KEWFC-K and KEWFC-F with the newly added basis kernel was also plotted in the same figure. It can be observed from the figure that, although the newly added kernels were more and more ineffective for CKS-EWFC-K, the clustering quality of both CKS-EWFC-K and CKS-EWFC-F remained steady, demonstrating their immunity to ineffective kernels.



Australian Parkinsons Wine

Fig.2 Scalability of CKS-EWFC-K and CKS-EWFC-F with respect to kernel number

4.5 Noisy datasets

The robustness of the proposed algorithms was evaluated with noisy datasets. In the experiments, the proposed algorithms were first tested with noisy datasets generated by introducing uniformly distributed noise within the range [*-dev*, *dev*] into the *Australian* Dataset of the UCI repository. The noisy datasets were generated with *dev* taking values within the range from 0.2 to 2. The robustness of the algorithms against the increasingly noisy datasets is shown in Fig. 3. It can be seen that the clustering quality of the algorithms degraded with the amount of noise added to the dataset. Nevertheless, the performance of the proposed CKS-EWFC-K and CKS-EWFC-F always performed better than the other algorithms, clearly demonstrating that they were more robust against noise in the clustering process.

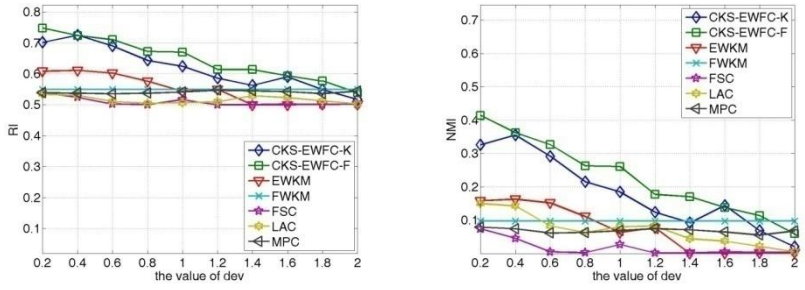


Fig.3 Performance of soft subspace clustering algorithms on noisy dataset

The proposed algorithms were also tested with segmented noisy texture images to evaluate their performance in handling large-scale dataset. The images used in the experiment were synthesized using five texture patterns drawn from the Brodatz texture database [36]. The resolution of the images was 100×100 pixels. Fig.4(a) shows the texture image and Fig.4(b) shows the ideal segmentation results.

In order to verify the robustness of the proposed algorithms, Gaussian noise with zero mean and different standard deviations (σ=0.05, 0.10, 0.15, 0.20, 0.25, 0.30) was added to generate the six noisy texture images shown in Fig.4(c)-(h). To generate data applicable to the clustering algorithms, Gabor filter [37] was applied to extract features from the texture images. A filter bank with 4 orientations (one every 45°) and five frequencies (starting from 0.46) was created. A 20-dimensional feature vector for every pixel of the images was thus extracted and datasets containing 10000 20-dimensional feature vectors were created accordingly.

In the experiment, the performance of the proposed algorithms in image segmentation was compared with that of other clustering algorithms. The optimal parameter settings of the algorithms were found by using the grid searching strategy. For each parameter setting, the algorithms were executed 10 times. Fig.5 shows the average clustering performance in terms of RI and NMI over the six noisy texture images, with the algorithms executed under the optimal parameter settings. The results show that the proposed algorithms outperformed the other algorithms in the image segmentation application, which further demonstrates the merit of integrating distance metric learning into the process of soft subspace clustering.

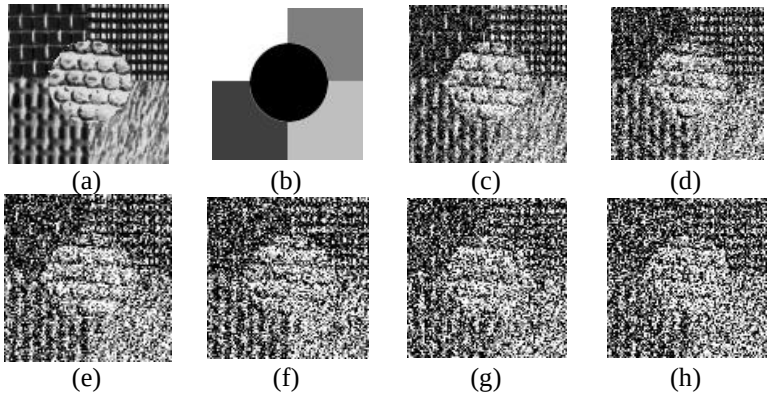


Fig.4 Noisy texture images

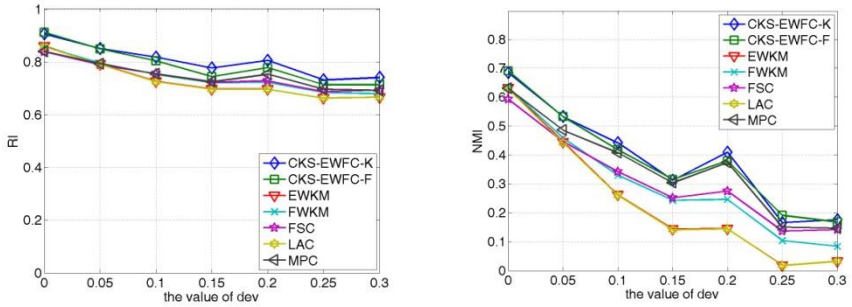


Fig.5 Clustering performance on noisy texture images with different amount of noises added.

5. CONCLUSIONS AND FUTURE WORK

The existing soft subspace clustering algorithms often utilize one distance function only to evaluate the similarity between data items on each feature, making them incapable of handling complex datasets. In this regard, the mechanism of distance metric learning in CKS is investigated. The work of the paper includes: (1) the construction of CKS by the linear combination

of a set of basis kernel mappings and the mapping of the prototype in feature space into the CKS by a class of mappings; (2) the development of the CKS-EWFC-K and CKS-EWFC-F algorithms using a novel learning criterion that combines the framework of entropy weighting soft subspace fuzzy clustering and distance metric learning in CKS; (3) the comprehensive experiments conducted to evaluate the performance of the proposed algorithms. The results show that the proposed algorithms can adaptively learn the distance functions suitable for the datasets during the clustering process, and the overall performance is better than that of other classical algorithms. These characteristics are appealing for various real-world applications.

This study will be further extended to improve the performance of other soft subspace clustering algorithms. For example, CKS versions of existing fuzzy weighting subspace clustering algorithms [11] and multiple-view clustering algorithms in CKS can be developed. In addition, to make the algorithms more practical for real world applications, it is important to conduct a theoretical study on the setting of the parameters of the algorithms so that guidelines can be provided to facilitate the identification of the parameters.

ACKNOWLEDGEMENTS

This work was supported in part by the Hong Kong Polytechnic University under Grant G-UA68, the National Natural Science Foundation of China under Grants 61103128, 61272210 and 61300151, the Fundamental Research Funds for the Central Universities (JUSRP51321B), the Natural Science Foundation of Jiangsu Province under Grant BK20130155 and BK20130160, the University Natural Science Research Project in Jiangsu Province (13KJB520001), and the Hong Kong Research Grants Council (PolyU 5134/12E).

References

[1] G. Gan, C. Ma, J. Wu, *Data Clustering: Theory, Algorithms, and Applications* (Asa-Siam Series on Statistics and Applied Probability), Society for Industrial & Applied Mathematics, USA, 2007.

[2] A. K. Jain, M. N. Murty, P. J. Flynn, Data Clustering: A review, ACM Computing Surveys, 31(3) (1999) 264-323.

[3] X. Li, H-S Wong, S. Wu. A fuzzy minimax clustering model and its applications, Information Sciences, 186(1) (2012) 114-125.

[4] J. Valente de Oliveira, W. Pedrycz, Advances in Fuzzy Clustering and Its Applications, Wiley and Sons, Chichester, West Sussex, England, 2007.

[5] X. Wang, Y. Wang, L. Wang, Improving fuzzy c-means clustering based on feature-weight learning, Pattern Recognition Letters, 25(10)(2004) 1123–1132.

[6] R. Xu, D. Wunsch, Survey of clustering algorithms, IEEE Transactions on Neural Networks, 16(3) (2005) 645-678.

[7] L. Zhu, F. L. Chung, S. Wang. Generalized fuzzy c-means clustering algorithm with improved fuzzy partitions, IEEE Transactions on Systems, Man, and Cybernetics, Part B, 39(3) (2009) 578-591.

[8] Y. Chan, W. Ching, M. K. Ng, J. Z. Huang. An optimization algorithm for clustering using weighted dissimilarity measures, Pattern Recognition, 37(5) (2004) 943-952.

[9] L. Chen, Q. Jiang, S. Wang. Model-based method for projective clustering, IEEE Transactions on Knowledge and Data Engineering, 24(7) (2012) 1291-1305.

[10] Z. Deng, K-S Choi, F-L Chung, S. Wang, Enhanced soft subspace clustering integrating within-cluster and between-cluster information, Pattern Recognition, 43(3) (2010) 767–781.

[11] G. Gan, J. Wu, A convergence theorem for the fuzzy subspace clustering (FSC) algorithm, Pattern Recognition, 41(6) (2008) 1939-1947.

[12] L. Jing, M. K. Ng, J. Xu, J. Z. Huang. Subspace clustering of text documents with feature weighting k-means algorithm, in: Proceedings of the Ninth Pacific-Asia Conference on Knowledge Discovery and Data Mining, 2005, pp.802–812.

[13] L. Jing, M. K. Ng, J. Z. Huang, An Entropy Weighting k-Means Algorithm for Subspace Clustering of High-Dimensional Sparse Data, IEEE Transactions on Knowledge and Data Engineering, 19(8) (2007) 1026-1041.

[14] F. Li, J. Yang, J. Wang, "A transductive framework of distance metric learning by spectral dimensionality reduction," in: Proceedings of 24th International conference on Machine learning. Penn Plaza, Suite 701 NewYork, NY, USA, pp.513-520, 2007.

[15] Y. Pan, J. Wang, Z. Deng. Alternative soft subspace clustering algorithm, Journal of Information and Computational Science, 10(12) (2013) 3615–3624.

[16] H. Shen, J. Yang, S. Wang, X. Liu. Attribute weighted mercer kernel based fuzzy clustering algorithm for general non-spherical datasets, Soft Computing, 10(11) (2006) 1061-1073.

[17] J. Wang, S. Wang, F. L. Chung, Z. Deng. Fuzzy partition based soft subspace clustering and its applications in high dimensional data, Information Sciences, 246(10) (2013) 133-154.

[18] L. Zhu, L. Cao, J. Yang, Soft subspace clustering with competitive agglomeration, in: 2011 IEEE International Conference on Fuzzy Systems, 2011, pp.27-30.

[19] L. Zhu, L. Cao, J. Yang, J. Lei. Evolving soft subspace clustering, Applied Soft Computing, 14(1) (2014) 210-228.

[20] J. H. Friedman, J. J. Meulman. Clustering objects on subsets of attributes, Journal of the Royal Statistical Society: Series B, 66(4) (2004) 815-849.

[21] E. P. Xing, A. Y. Ng, M. I. Jordan, S. J. Russell, Distance metric learning, with application to clustering with side-information, in: Advances in Neural Information Processing Systems 15, 2002, pp. 505-512.

[22] J. V. Davis, B. Kulis, P. Jain, S. Sra, and I. S. Dhillon, Information-theoretic metric learning, in: Proceedings of International Conference on Machine Learning, 2007, pp. 209-216.

[23] Y. Ying, K. Huang, C. Campbell, Sparse Metric Learning via Smooth Optimization, in: Advances in Neural Information Processing Systems 22, 2009, pp. 2214-2222.

[24] H. Chang, D. Y. Yeung, Locally linear metric adaptation for semi-supervised clustering, in: Proceedings of the 21st International Conference on Machine Learning, 2004, pp.153-160.

[25] J. Chen, Z. Zhao, J. Ye, H. Liu. Nonlinear Adaptive Distance Metric Learning for Clustering, in: Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining, 2007, pp.123-132.

[26] C. Lu, G. Feng, J. Jiang, P. Wang, Metric learning: A general dimension reduction framework for classification and visualization, in: Proceedings of 19th International Conference on Pattern Recognition, 2008, pp.1–4.

[27] S. Okada, T. Nishida. Online incremental clustering with distance metric learning for high dimensional data, in: Proceedings of the 2011 International Joint Conference on Neural Networks (IJCNN), San Jose, California, USA, 2011, pp.2047-2054.

[28] L. Yang, R. Jin. Distance metric learning: A comprehensive survey, Available: http://www.cs.cmu.edu/~liuy/frame_survey_v2.pdf, 2006.

[29] J. Ye, Z. Zhao, H. Liu. Adaptive distance metric learning for clustering, in: 2007 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2007, pp.1-7.

[30] R. C. de Amorim, B. Mirkin. Minkowski metric, feature weighting and anomalous cluster initializing in K-Means clustering, Pattern Recognition, 45(3) (2012) 1061-1075.

[31] B. SchAolkopf, A. J. Smola, Learning with Kernels. Cambridge, MA: MIT Press, 2002.

[32] D. Graves, W. Pedrycz, Kernel-based fuzzy clustering and fuzzy clustering: A comparative experimental study, Fuzzy Sets and Systems,161(4) (2010) 522–543.

[33] H-C Huang, Y-Y Chuang, C-S Chen. Multiple kernel fuzzy clustering, IEEE Transactions on Fuzzy Systems, 20(1) (2012) 120-134.

[34] W. M. Rand, Objective criteria for the evaluation of clustering methods, Journal of the American Statistical Association, 66(1971) 846-850.

[35] J. Demsar, Statistical comparisons of classifiers over multiple datasets, Journal of Machine Learning Research, 7(2006) 1-30.

[36] T. Randen, Brodatz Texture, http://www.ux.uis.no/~tranden/brodatz.html.

[37] V. Kyrki, J. K. Kamarainen, H. Kalviainen, Simple Gabor feature space for invariant object recognition, Pattern Recognition Letter, 25(3) (2004) 311-318.

[38] C. Domeniconi, D. Papadopoulos, D. Gunopulos, S. Ma, Subspace clustering of high dimensional data, in: Proceedings of the SIAM International Conference on Data Mining, 2004.

[39] X. Chen, Y. Ye, X. Xu, J. Z. Huang, A feature group weighting method for subspace clustering of high-dimensional data, Pattern Recognition, 45(1) (2012) 434-446.

[40] G. Zhong, K. Huang, C-L Liu, Low rank metric learning with manifold regularization, in: Proceedings of International Conference on Data Mining, 2011, pp. 1266-1271.