A Survey on Soft Subspace Clustering

Zhaohong Deng\textsuperscript{1,2}, Kup-Sze Choi\textsuperscript{3}, Jun Wang\textsuperscript{1}, Shitong Wang\textsuperscript{1}

\textsuperscript{1} School of Digital Media, Jiangnan University, Wuxi, Jiangsu, P.R. China
\textsuperscript{2} Department of Biomedical Engineering, University of California, Davis, USA
\textsuperscript{3} The Centre for Smart Health, The Hong Kong Polytechnic University, Hong Kong

Abstract: Subspace clustering (SC) is a promising clustering technology to identify clusters based on their associations with subspaces in high dimensional spaces. SC can be classified into hard subspace clustering (HSC) and soft subspace clustering (SSC). While HSC algorithms have been extensively studied and well accepted by the scientific community, SSC algorithms are relatively new but gaining more attention in recent years due to better adaptability. In the paper, a comprehensive survey on existing SSC algorithms and the recent development are presented. The SSC algorithms are classified systematically into three main categories, namely, conventional SSC (CSSC), independent SSC (ISSC) and extended SSC (XSSC). The characteristics of these algorithms are highlighted and the potential future development of SSC is also discussed.

Keywords: Soft subspace clustering; fuzzy weighting; entropy weighting; fuzzy C-means/K-means model; mixture model
1 Introduction

Despite extensive studies of clustering techniques over the past decades in various application areas like statistics, machine learning and database [1-2], the conventional clustering techniques fall short when clustering is performed in high dimensional spaces [4-6]. A key challenge to most clustering algorithms is that, in many real world problems, data points in different clusters are often correlated with some subsets of features, i.e., clusters may exist in different subspaces or a certain subspace of all features. Therefore, for any given pair of data points within the same cluster, it is possible that the points are indeed far apart from each other in a few dimensions of high dimensional space.

Recent years have witnessed proliferating development of subspace clustering (SC) techniques to overcome this challenge. The goal of SC is to locate clusters in different subspaces or a certain subspace of the original data space. The two main schools of SC algorithms are Hard Subspace Clustering (HSC) and Soft Subspace Clustering (SSC). Research in SC begins with in depth study of HSC methods for clustering high dimensional data. This category of SC algorithms attempts to identify the exact subspaces for different clusters, which can be further divided into bottom-up and top-down subspace search methods [4]. Examples of the former include CLIQUE [8], ENCLUS [9] and MAFIA [10]; and that of the latter are ORCLUS [11], FINDIT [12], DOC [13], $\delta$-Clusters [14], and PROCLUS [15]. Other common HSC algorithms also include HARP [16] and LDR [17]. A detailed review of HSC algorithms can be found in [4-7].

While the goal of HSC is to identify exact subspaces, SSC algorithms performs clustering in high dimensional spaces by assigning a weight to each dimension to measure the contribution of individual dimensions to the formation of a particular cluster. SSC can be considered as an extension of the conventional feature weighting clustering [18-27]. In this paper, SSC algorithms are hierarchically classified into three main categories, namely, (1) conventional SSC (CSSC), (2) independent SSC (ISSC), and extended SSC (ESSC). Here, CSSC refers to conventional feature weighting clustering algorithms, i.e., all clusters share the same subspace and a common weight
vector. On the contrary, the weight vectors in ISSC are different for different clusters. In other words, each cluster has an independent subspace. Thus, ISSC may also be referred to as multiple features weighting clustering. XSSC represents a category of algorithms developed by extending the CSSC or ISSC algorithms with the introduction of new mechanisms to enhance clustering performance. The definitions of these three types of SSC are given in Table 1. A detailed review of the characteristics of the algorithms will be discussed.

The rest of this paper is organized as follows. The classification of the existing SSC algorithms is given in Section 2. The CSSC, ISSC and ESSC algorithms are comprehensively reviewed in Sections 3, 4 and 5 respectively. The perspectives and prospects of SSC are discussed in Section 6 and conclusions are given in Section 7. For clarity, the notations used in this paper are defined in Table 2.

Table 1 Definitions of three categories of SSC algorithms

| SCC algorithms | Descriptions |
|----------------|--------------|
| Conventional SSC (CSSC) algorithms | Classical feature weighting clustering algorithms with all the clusters sharing the same subspace and a common weight. |
| Independent SSC (ISSC) algorithms | Multiple feature weighting clustering algorithms with all the clusters having their own weight vectors, i.e., each cluster has an independent subspace, and the weight vectors are controllable by different mechanisms. |
| Extended SSC (XSSC) algorithms | Algorithms extending the CSSC or ISSC algorithms with new clustering mechanisms for performance enhancement. |

Table 2 Notations commonly used by algorithms discussed in this paper

| Notations | Description |
|-----------|-------------|
| U = [u_{ij}]_{C \times N} | Hard/fuzzy partition matrices |
| \(v_i = [v_{i1}, \ldots, v_{id}]^T\) | Clustering centers matrix, \(v_i\) is the center of the \(i\)th cluster. |
| W = [w_{ij}]_{C \times D} = [w_{11}, \ldots, w_{cd}]^T, w_i = [w_{i1}, \ldots, w_{id}]^T | Weighting matrix \(w_i\), the weight vector associated with \(i\)th cluster. |
| \(m\) | Fuzzy index of fuzzy memberships |
| \(\tau\) | Fuzzy index of fuzzy weighting |
| \(N\) | Number of data/samples |
| \(D\) | Number of features |
2 Classification of SSC

As discussed previously, SSC algorithms can be broadly classified into three main categories, CSSC, ISSC and XSSC. Each of these categories can be further divided into subcategories based on variations in clustering mechanisms, as shown in Table 3. For CSSC, clustering is performed by first identifying the subspace with some strategies and then carrying out clustering in the obtained subspace to partition the data. This is referred to as separated feature weighting where data partitioning involves two separate processes – subspace identification and clustering in subspace. On the other hand, clustering can also be conducted with the two processes performed simultaneously, which is thus referred to as coupled feature weighting. For ISSC, algorithms are developed based on fuzzy C-means (FCM)/K-means model or mixture model, where fuzzy weighting, entropy weighting or other weighting mechanisms are adopted to implement the clustering models. Finally, XSSC algorithms can be subdivided into six subcategories depending on the strategies used to enhance the CSSC and ISSC algorithms, including between-class separation, multi-view learning, evolutionary learning, adoption of new metrics, unbalanced clusters, and other approaches like reliability mechanism and those used for clustering categorical dataset. The classification of SSC algorithms is presented in Table 3. These algorithms will be discussed in the sequent sections.

| CSSC                      | ISSC                      |
|---------------------------|---------------------------|
| Separated feature weighting | Coupled feature weighting |
| [18-21]                   | [22-27]                   |
| [28-33]                   | [34], [35]                |
| [36]                      | [37],[38]                 |
| [39]                      | [40]                      |

| XSSC                      |
|---------------------------|
| Between-class separation  |
| Multi-view learning       |
| Evolutionary learning     |
| New metrics               |
| Unbalanced clusters       |
| Others                    |
| [41-43]                   |
| [44], [45]                |
| [46-48]                   |
| [49], [50]                |
| [51]                      |
| [52-59, 65]               |
3 CSSC

CSSC algorithms can be classified based on the methods of feature weighting approaches adopted. They are thus divided into two categories, one adopting separated feature weighting and the other coupled employing feature weighting. For the former, the weights are determined before clustering is performed; while the weights are learned during the clustering process in the latter.

3.1 Separated Feature Weighting Algorithms

(1) OVW-UAT

The Optimal Variable Weighting for Ultrametric and Additive Tree (OVW-UAT) is a feature weighting strategy developed for hierarchical clustering methods to solve the variable weighting problems [18]. In this approach, two objective functions are developed to determine the weights for trees in ultrametric and additive forms respectively. Once the optimal variable weights are obtained, the resulting inter-object dissimilarities can be applied to any of the existing ultrametric or additive tree fitting procedures. Since hierarchical clustering methods are computationally complex, the OVW-UAT approach cannot handle large data sets efficiently. Makarenkov and Legendre extended the OVW-UAT approach to optimal variable weighting for the k-means clustering [19]. The simulation results showed that the method was effective for identifying important variables, but still not scalable to large data sets.

(2) C-K-means

The Convex K-means (C-K-means) is a method proposed specifically for variable weighting in k-means clustering [20]. This method aims to optimize variable weights in order to achieve the best clustering result by minimizing the generalized Fisher ratio $Q$ – the ratio of the average within-cluster distortion to the average between-cluster distortion. To find the minimum $Q$ value, a set of feasible weight groups are first defined. For each weight group, the k-means algorithm is used to generate a data partition and $Q$ is calculated from the partition. The most desirable cluster is then determined as the partition having the minimum $Q$ value. However,
this method of finding optimal weights from a predefined set of variable weights may not guarantee that the predefined set of weights would always contain the optimal weights. Besides, it is not practical to obtain a predefined set of weights for high dimensional data.

(3) WFCM

The Weighted FCM (WFCM) algorithm is another clustering method grouped under the category of separated feature weighting [21]. In the algorithm, clustering is performed by employing the weighted Euclidean distance as metric that incorporates feature weights into the commonly used Euclidean distance. The algorithm begins by estimating the weight vector using the objective function below and the gradient descent learning technique,

$$
\min E(w) = \frac{2}{N(N-1)^2} \sum_{i=1}^{N} \sum_{j \neq i}^{N} \frac{1}{2} \left( \rho_{ij}^{(w)} (1 - \rho_{ij}) + \rho_{ij} (1 - \rho_{ij}^{(w)}) \right)
$$

$$
\rho_{ij}^{(w)} = \frac{1}{1 + \beta \cdot d_{ij}^{(w)}} , \quad d_{ij}^{(w)} = \sqrt{\sum_{k=1}^{P} w_k^2 (x_{ik} - x_{jk})^2}
$$

where $E(w)$ is the function of weighting variable $w$ for obtaining the optimized weights; $N$ is the number of samples; $d_{ij}^{(w)}$ and $\rho_{ij}^{(w)}$ denote $d_{ij}$ and $\rho_{ij}$ in the original space respectively. Once the weights are determined, WFCM can be implemented by replacing the common Euclidean distance in FCM by the weighted Euclidean distance.

3.2 Coupled Feature Weighting Algorithms

(1) SYNCLUS

The Synthesized Clustering (SYNCLUS) algorithm is developed to deal with variable weighting in k-means clustering [22]. The algorithm is divided into two stages. Starting from an initial set of weights, SYNCLUS first employs k-means clustering to partition data into $k$ clusters, followed by the estimation of a set of new weights for different features by optimizing a weighted mean-square stress-like cost function. The two stages iterate until convergence to an optimal set of weights is achieved. The algorithm is computationally intensive and very time-consuming [3], making it not suitable for handling large data sets.

(2) FWFKM
The Feature Weighted Fuzzy K-means (FWFKM) algorithm performs clustering through an iterative procedure based on the fuzzy k-means algorithm and the supervised ReliefF algorithm [23]. With \( D \) as the number of features, the FWFKM algorithm begins by setting the weights as \( 1/D \) and implementing the fuzzy k-means algorithm with the weighted distances to obtain the initial clustering result and label the data. With the labeled data, the supervised ReliefF algorithm is then used to assign the new weights for every feature. This procedure is conducted iteratively with the weights updated repeatedly until the final clustering result is achieved.

\((3)\) **W-k-means**

Huang et al. proposed the automated variable weighting in k-means type clustering algorithm (W-k-means) by using the following objective function [24],

\[
\min J_{W-k\text{-}means}(U, V, w) = \sum_{i=1}^{C} \sum_{j=1}^{N} \sum_{k=1}^{D} w_k (x_{ik} - v_{ik})^2
\]

s.t. \( u_{ij} \in \{0, 1\} \), \( \sum_{i=1}^{C} u_{ij} = 1 \), \( 0 < \sum_{j=1}^{N} u_{ij} < N \), \( 0 \leq w_k \leq 1 \), and \( \sum_{k=1}^{D} w_k = 1 \).

With the current partition in the iterative k-means clustering process, the W-k-means algorithm calculates a new weight for each variable, i.e., feature, based on the variance of the within-cluster distances. The new weights are used to decide the cluster memberships of objects in the next iteration. The optimal weights are found when the algorithm converges, which can then be used to identify important features for clustering. The features that may indeed be noise to the clustering process can be removed in future analysis. The W-k-means algorithm has received increasing attention, based on which modified algorithms are proposed, such as the fuzzy subspace clustering algorithm [32, 33] to be reviewed in Subsection 4.1.1.

\((4)\) **MWLA**

Cheung and Zeng proposed the Maximum Weighted Likelihood (MWL) learning framework in the context of Gaussian mixture model to identify the clustering structure and the relevant features automatically and simultaneously [25]. The MWL based algorithm (MWLA) is performed by introducing two sets of weight functions – one to reward the significance of each component in the mixture, and the other to discriminate the relevance of each feature of the clustering structure.
Tsai and Chiu proposed the Feature Weight Self-Adjustment Algorithm (FWSA) based on the k-means clustering model [26]. The algorithm adopts the objective function below,
\[\min J_{FWSA}(U, V) = \sum_{i=1}^{C} \sum_{j=1}^{N} u_{ij} \sum_{k=1}^{D} w_{ik} (x_{jk} - v_{ik})^2\]
\[\text{s.t. } u_{ij} \in \{0,1\}, \sum_{i=1}^{C} u_{ij} = 1, \quad 0 < \sum_{j=1}^{N} u_{ij} < N, \quad 0 \leq w_{ik} \leq 1, \quad \text{and} \quad \sum_{k=1}^{D} w_{ik} = 1.\]

Furthermore, the following sub-optimization function is used to adjust the weights
\[\max E(w) = \frac{\sum_{i=1}^{C} \parallel C_i \parallel \sum_{k=1}^{D} w_{ik} (v_{ik} - v)\parallel^2}{J_{FWSA}(w)},\]
where \(\parallel C_i \parallel\) denotes the number of the \(i\)th cluster obtained in the current iteration and \(v_{a} = [v_{a1}, \cdots, v_{ad}]^T\) is the global center of all data objects in the data set. The final clustering results and weights are then obtained iteratively.

For the problems of unsupervised discrete feature selection/weighting, the Model-based Approach for Discrete Data Clustering and Feature Weighting (MA-DDC-FW) [27] is proposed that makes use of a probabilistic approach to assign relevance weights to the discrete features. In the algorithm, the features are considered as random variables modeled by finite discrete mixtures. Bayesian and information-theoretic approaches through stochastic complexity are both employed for the learning of the model. The feasibility and merits of MA-DDC-FW are well demonstrated in difficult problems involving clustering and recognition of visual concepts in image data. The algorithm has also achieved success in text clustering.
4.1 FCM/K-means model based ISSC

4.1.1 Fuzzy Weighting

For ISSC algorithms based on FCM/K-means model and fuzzy weighting mechanism, the parameters \( w_{ik} \) are used for fuzzy weighting with \( \tau \) as the fuzzy indices for weighting, where \( \tau \) can control the distributions of \( w_{ik} \) effectively in the clustering procedure. For example, \( w_{ik} \rightarrow 1/D \) with a very large \( \tau \).

**1) AWFCM**

Keller and Klawonn proposed the Attribute Weighting Fuzzy Clustering (AWFCM) based on FCM model with the objective function below [28].

\[
\min J_{AWFC}(U, V, W) = \sum_{i=1}^{C} \sum_{j=1}^{N} u_{ij} \sum_{k=1}^{D} w_{ik}^p (x_{ik} - v_j)^2 \\
\text{s.t.} \; u_{ij} \in [0,1], \; \sum_{j=1}^{C} u_{ij} = 1, \; 0 < \sum_{j=1}^{N} u_{ij} < N, \; 0 \leq w_{ij} \leq 1, \; \text{and} \; \sum_{k=1}^{D} w_{ik} = 1.
\]

To our knowledge, AWFCM is the first fuzzy weighing ISSC algorithm. Based on AWFCM, improved versions of the algorithm have been proposed, such as EFWSSC algorithm [41].

**2) SCAD**

Frigui and Nasraoui proposed the Simultaneous Clustering and Attribute Discrimination (SCAD) method [29]. Two versions of SCAD algorithms, SCAD-1 and SCAD-2, were proposed with the objective functions as follows,

\[
\min J_{SCAD-1}(U, V, W) = \sum_{i=1}^{C} \sum_{j=1}^{N} u_{ij} \sum_{k=1}^{D} w_{ik}^p (x_{ik} - v_j)^2 + \sum_{i=1}^{C} \sum_{k=1}^{D} w_{ik}^2 \\
\text{s.t.} \; u_{ij} \in [0,1], \; \sum_{j=1}^{C} u_{ij} = 1, \; 0 \leq w_{ij} \leq 1, \; \sum_{k=1}^{D} w_{ik} = 1,
\]

\[
\min J_{SCAD-2}(U, V, W) = \sum_{i=1}^{C} \sum_{j=1}^{N} u_{ij}^{mp} \sum_{k=1}^{D} w_{ik}^p (x_{ik} - v_j)^2 \\
\text{s.t.} \; u_{ij} \in [0,1], \; \sum_{j=1}^{C} u_{ij} = 1, \; 0 \leq w_{ij} \leq 1, \; \sum_{k=1}^{D} w_{ik} = 1,
\]

By comparing the objective functions of AWFCM and SCAD-2, it can be found that they are essentially fuzzy weighting SSC algorithms of the same kind.

**3) AWA**

The Attribute Weighting Algorithm (AWA) developed by Chan et al. [30] employed an objective function similar to that of the W-k-means algorithm [24]. However, the sharing weights
The objective function is expressed as,

\[
\min J_{AWA}(U, V, W) = \sum_{i=1}^{C} \sum_{j=1}^{N} u_{ij} \sum_{k=1}^{D} w_{ik} (x_{ik} - v_{jk})^2
\]

s.t. \( u_{ij} \in \{0, 1\}, \sum_{i=1}^{C} u_{ij} = 1, \ 0 \leq w_{ij} \leq 1, \ \sum_{k=1}^{D} w_{ik} = 1. \)

In fact, it is also clear from the objective function of AWA and that of AWFCM as discussed above [28] that AWA is a hard clustering version of the AWFCM algorithm.

**4) FWKM**

A weakness of AWA is that when some of the attributes have zero standard deviation, the learning rules derived by the algorithm fail to work. The Fuzzy Weighting K-Means (FWKM) algorithm [31] is proposed to overcome this problem. The objective function of FWKM is as follows,

\[
\min J_{FWKM}(U, V, W) = \sum_{i=1}^{C} \sum_{j=1}^{N} u_{ij} \sum_{k=1}^{D} w_{ik} \left[ (x_{ik} - v_{jk})^2 + \sigma \right]
\]

\[
\sigma = \frac{\sum_{j=1}^{N} \sum_{k=1}^{D} (x_{jk} - o_k)^2}{N \cdot D}, \quad o_k = \frac{\sum_{j=1}^{N} x_{jk}}{N}
\]

s.t. \( u_{ij} \in \{0, 1\}, \sum_{i=1}^{C} u_{ij} = 1, \ 0 < \sum_{j=1}^{N} u_{ij} < N, \ 0 \leq w_{ij} \leq 1, \ \text{and} \ \sum_{k=1}^{D} w_{ik} = 1. \)

**5) FSC**

Gan et al. proposed the Fuzzy Subspace Clustering (FSC) algorithm [32, 33] using an objective similar to that of the FWKM algorithm [31] discussed previously. A detailed analysis of the properties of FSC can be found in [33].

\[
\min J_{FSC}(U, V, W) = \sum_{i=1}^{C} \sum_{j=1}^{N} u_{ij} \sum_{k=1}^{D} w_{ik} (x_{ik} - v_{jk})^2 + \varepsilon \sum_{i=1}^{C} \sum_{k=1}^{D} w_{ik}
\]

s.t. \( u_{ij} \in \{0, 1\}, \sum_{i=1}^{C} u_{ij} = 1, \ 0 < \sum_{j=1}^{N} u_{ij} < N, \ 0 \leq w_{ij} \leq 1, \ \text{and} \ \sum_{k=1}^{D} w_{ik} = 1. \)

It is clear from these five fuzzy weighting ISSC algorithms that the fuzzy weighting \( w_{ik}^\tau \) can be regarded as an extension of the classical weighting \( w_{ik} \). The fuzzy index \( \tau \) is usually set such that \( \tau > 1 \) in order to ensure the convergence of the algorithms [28-33]. When \( \tau \rightarrow 1 \), the fuzzy weighting \( w_{ik}^\tau \) is reduced to the classical weighting \( w_{ik} \). Compared with classical weighting, algorithms adopting fuzzy weighting have better elasticity and, meanwhile, can be
easily analyzed using fuzzy optimization techniques.

### 4.1.2 Entropy Weighting

Another category of ISSC algorithms has been developed based on the FCM/K-means model and entropy weighting. Unlike the fuzzy weighting based algorithms described in the previous subsection, the weighting in this category of algorithms are controllable by entropy. Two representative entropy weighting FCM/K-means model based ISSC algorithms are described.

(1) **EWKM**

Jing et al. proposed the Entropy Weighting K-Means (EWKM) clustering algorithm [34] using the following objective function,

$$
\min J_{EWKM}(U, V, W) = \sum_{i=1}^{C} \sum_{j=1}^{N} u_{ij} \sum_{k=1}^{D} w_{ik} (x_{jk} - v_{ik})^2 + \gamma \sum_{i=1}^{C} \sum_{k=1}^{D} w_{ik} \ln w_{ik}
$$

s.t. \( u_{ij} \in \{0,1\}, \sum_{j=1}^{N} u_{ij} = 1, 0 < \sum_{j=1}^{N} u_{ij} < N, 0 \leq w_{ij} \leq 1, \) and \( \sum_{k=1}^{D} w_{ik} = 1, \)

where the second term in the objective function is the Shannon entropy and \( \gamma \) is used to balance its influence on the clustering procedure. By introducing the entropy term, the obtained weight can be effectively controllable by entropy. For example, if \( \gamma \) is very large, the features will be apt to be assigned with equal values. EWKM has become a benchmarking ISSC algorithm and is further extended to develop various XSSC algorithms, such as ESSC algorithm [34].

(2) **LAC**

The weighting in the Local Adaptive Clustering (LAC) algorithm [35] is also controllable by entropy. The objective function can be expressed as

$$
\min J_{LAC}(U, V, W) = \sum_{i=1}^{C} \sum_{k=1}^{D} w_{ik} X_{ik} + \gamma \sum_{i=1}^{C} \sum_{k=1}^{D} w_{ik} \ln w_{ik},
$$

$$
X_{ik} = \left( \sum_{j=1}^{N} u_{ij} (x_{jk} - v_{ik})^2 \right) / \sum_{j=1}^{N} u_{ij}
$$

s.t. \( u_{ij} \in \{0,1\}, \sum_{i=1}^{C} u_{ij} = 1, 0 < \sum_{j=1}^{N} u_{ij} < N, 0 \leq w_{ij} \leq 1, \) and \( \sum_{k=1}^{D} w_{ik} = 1. \)

The objective functions of EWKM and LAC are indeed very similar. The only difference is that
the effect of cluster size is considered in LAC but disregarded in EWKM.

In addition to the entropy weighting LAC in [35], Domeniconi proposed an alternative LAC algorithm with a different objective function [36] as follows

$$\max_{J_{O-LAC} (U, V, W)} = \sum_{i=1}^{C} \sum_{k=1}^{D} w_{ik} \exp(h \cdot X_{ik})$$

$$X_{ik} = \left( \sum_{j=1}^{C} u_{ij} (x_{jk} - v_{ik})^2 \right) / \sum_{j=1}^{C} u_{ij}$$

s.t. $u_{ij} \in \{0, 1\}$, $\sum_{i=1}^{C} u_{ij} = 1$, $0 < \sum_{j=1}^{N} u_{ij} < N$, $0 \leq w_{ij} \leq 1$, and $\sum_{k=1}^{D} w_{ik}^2 = 1$.

Note that the objective function is maximized to solve the solution variables, which is distinct from the entropy-based LAC approach [35] discussed earlier.

### 4.2 Mixture Model based ISSC

In addition to classical FCM/K-means model, probability mixture model is also adopted to develop ISSC algorithms. In this subsection, two representative algorithms are introduced.

**(1) FPC/MPC**

Chen et al. proposed the Fuzzy/Model Projective Clustering (FPC/MPC) algorithm based on the mixture model [37, 38]. In FPC/MPC, each projected dimension is assumed to fit in the Gaussian mixture distribution as follows,

$$F(x_i; \Theta_i) = \sum_{k=1}^{C} \alpha_k G(x_i; \nu_k, \sigma_k)$$

$$G(x_i; \nu_k, \sigma_k) = \frac{1}{\sqrt{2\pi\sigma_k}} \exp \left( -\frac{w_k}{2\sigma_k^2} \right)$$

with $\sum_{k=1}^{C} \sqrt{w_k} = 1$, where $\nu_k, \sigma_k$ denote the means and variances respectively. Furthermore, the following Kullback-Leibler (KL) divergence is minimized for parameter learning

$$R(\hat{\Theta}_j) = \int F(x_i; \Theta_i) \ln \frac{F(x_i; \Theta_i)}{F(x_i; \hat{\Theta}_j)} dx_i.$$  

Based on the above criterion, the objective function for clustering is finally given by

$$J_{FPC/MPC} (U, V, W, \alpha, \sigma, e) = \sum_{i=1}^{C} \sum_{j=1}^{N} \sum_{k=1}^{D} u_{ij} w_{ik} \left( (x_j - \nu_k) \cdot e_k^T \right)^2 - \sum_{i=1}^{C} D \ln \frac{\alpha_i}{\sqrt{2\pi\sigma_i}} \sum_{j=1}^{N} u_{ij} + D \sum_{j=1}^{N} u_{ij} \ln u_{ij}$$
s.t. $u_y \in \{0,1\}$, $\sum_{i=1}^{C} u_{ij} = 1$, $0 < \sum_{j=1}^{N} u_{ij} < N$, $0 \leq w_{ij} \leq 1$, $\sum_{k=1}^{D} \sqrt{w_{ik}} = 1$, $0 \leq \alpha_i \leq 1$ and $\sum_{i=1}^{C} \alpha_i = 1$.

When only axis-aligned subspace is considered, the above objective function can be reduced to

$$J_{\text{FPC/MPC}}(U,V,W,\alpha,\sigma) = \frac{1}{N} \sum_{i=1}^{C} \sum_{j=1}^{N} u_{ij} \left[ -\ln \alpha_i + \sum_{k=1}^{D} \frac{w_{ik}}{2\sigma_i^2} (x_{jk} - v_{ik})^2 - \ln \left( \frac{w_{ik}}{2\sigma_i^2} \right) \right] + \sum_{i=1}^{C} \alpha_i \sum_{j=1}^{N} u_{ij} \ln u_{ij}.$$

It can be seen that the FPC/MPC algorithm also involves the entropy term for the clustering procedure, however, the entropy term here is used to control the partition $u_{ij}$ instead of the feature weight $w_{ij}$.

(2) EWMM

Peng and Zhang proposed the entropy weighting mixture model (EWMM) algorithm [39]. By using an approach similar to that in FPC/MPC, the objective function of EWMM can be formulated as follows,

$$J(U,V,W,\alpha,\sigma) = \frac{1}{N} \sum_{i=1}^{C} \sum_{j=1}^{N} u_{ij} \left[ -\ln \alpha_i + \sum_{k=1}^{D} \frac{w_{ik}}{2\sigma_i^2} (x_{jk} - v_{ik})^2 - \ln \left( \frac{w_{ik}}{2\sigma_i^2} \right) \right] + \sum_{i=1}^{C} \alpha_i \sum_{j=1}^{N} u_{ij} \ln u_{ij}.$$

Furthermore, with the extra control of weighting by using entropy, the final objective function for clustering is given by

$$J_{\text{EWMM}}(U,V,W,\alpha,\sigma) = \frac{1}{N} \sum_{i=1}^{C} \sum_{j=1}^{N} u_{ij} \left[ -\ln \alpha_i + \sum_{k=1}^{D} \frac{w_{ik}}{2\sigma_i^2} (x_{jk} - v_{ik})^2 - \ln \left( \frac{w_{ik}}{2\sigma_i^2} \right) \right] + \sum_{i=1}^{C} \alpha_i \sum_{j=1}^{N} u_{ij} \ln w_{ij}.$$

Note that although both mixture model based algorithms, i.e., Fuzzy/Model algorithm and EWMM algorithm, contain an entropy term, the purposes are very different. The former uses it to control the partition $u_{ij}$ while the latter use it for controlling the weight $w_{ij}$.

When compared with the classical FCM/K-means model based ISSC algorithms, the two mixture model based algorithms are expected to possess stronger adaptation abilities to the data distributions, as evident from that promising clustering results achieved on high dimensional data [39]. However, the mixture model based ISSC algorithms are much more complicated than the
FCM/K-means based algorithms and therefore not very popular in the field of SSC.

### 4.3 Other Model based ISSC

In addition to the FCM/K-Means and mixture model based ISSC algorithms, other ISSC algorithms have also been proposed. Among them, the Clustering Objects on Subsets of Attributes (COSA) is a representative algorithm which was proposed by Friedman and Meulman [40] with the following objective function.

\[
J_{COSA}(U, W) = \sum_{i=1}^{c} \frac{A_i}{n_i} \sum_{j \in i} \left[ \sum_{k=1}^{D} w_{ik} d_{ijk} + \alpha w_{ik} \ln w_{ik} \right] + \alpha \ln(D)
\]

\[
= \sum_{i=1}^{c} \frac{A_i}{n_i} \sum_{k=1}^{D} \left[ w_{ik} \sum_{j \in i} d_{ijk} \right] + \alpha \sum_{i=1}^{c} \frac{A_i}{n_i} \sum_{k=1}^{D} w_{ik} \ln w_{ik} + \alpha \sum_{i=1}^{c} \frac{A_i}{n_i} \sum_{k=1}^{D} \ln(D)
\]

s.t. \(0 \leq w_{ij} \leq 1, \sum_{k=1}^{D} w_{ik} = 1\),

where \(A_i\) are the cluster weights and \(n_i\) is the number of objects assigned to the \(i\)th cluster. COSA is an entropy-based subspace clustering algorithm. As discussed in [34], a shortcoming of COSA is that it may not be scalable to accommodate large data sets.

### 5 XSSC

XSSC algorithms refers to a category of SSC methods developed to improve the performance of CSSC and ISSC algorithms by introducing new learning mechanisms to improve the clustering performance. In the paper, we divide the XSSC algorithms into six subcategories. The techniques applied are reviewed respectively in this subsection.

#### 5.1 Between-cluster Separation

Most SSC algorithms perform clustering by optimizing the within-cluster compactness without making use of the between-cluster information. Recently, algorithms integrating between-cluster separation with within-cluster compactness have been developed. Three representative algorithms are reviewed.

**(1) ESSC**

Based on the EWKM method [34], the Enhanced SSC (ESSC) algorithm was proposed to
improve clustering performance by minimizing the within-cluster compactness in the weighting subspace and maximizing the between-cluster separation simultaneously. The objective function of the ESSC algorithm is as follows,

\[ J_{\text{ESSC}}(U,V,W) = \sum_{i=1}^{C} \sum_{j=1}^{N} u_{ij}^w \sum_{k=1}^{D} w_{ik} (x_{jk} - v_{ik})^2 + \gamma \sum_{i=1}^{C} \sum_{j=1}^{N} w_{ik} \ln w_{ik} - \eta \sum_{i=1}^{C} \sum_{j=1}^{N} u_{ij}^w \sum_{k=1}^{D} w_{ik} (v_{ik} - v_{0k})^2 \]

s.t. \( 0 \leq u_{ij} \leq 1 \), \( \sum_{i=1}^{C} u_{ij} = 1 \), \( 0 < \sum_{j=1}^{N} u_{ij} < N \), \( 0 \leq w_{ij} \leq 1 \), and \( \sum_{k=1}^{D} w_{ik} = 1 \).

The objective function contains three terms – the weighting within-cluster compactness, the entropy of weights and the weighting between-cluster separation. The first and second terms are directly inherited from the objective function of EWKM subspace clustering except that the k-means model is replaced by FCM model. In this objective function, the parameters \( \gamma (\gamma > 0) \) and \( \eta (\eta > 0) \) are used to control the influences of entropy and the weighting between-cluster separation respectively. It is noteworthy to point out that when \( m \to 1 \) and \( \eta = 0 \), the ESSC algorithm is reduced to the EWKM algorithm [34]. Thus, the EWKM algorithm can be regarded as a special case of the ESSC algorithm.

(2) EFWSSC

Using a similar strategy, Guan improved the FSC algorithm [32, 33] by making use of between-cluster separation and proposed the Enhanced Fuzzy Weighting Soft Subspace Clustering (EFWSSC) algorithm [42]. The objective function of the algorithm is

\[ J_{\text{EFWSSC}}(U,V,W) = \sum_{i=1}^{C} \sum_{j=1}^{N} u_{ij}^w \sum_{k=1}^{D} w_{ik} (x_{jk} - v_{ik})^2 + \varepsilon \sum_{i=1}^{C} \sum_{j=1}^{N} w_{ik}^f - \eta \sum_{i=1}^{C} \sum_{j=1}^{N} u_{ij}^w \sum_{k=1}^{D} w_{ik} (v_{ik} - v_{0k})^2 \]

s.t. \( 0 \leq u_{ij} \leq 1 \), \( \sum_{i=1}^{C} u_{ij} = 1 \), \( 0 < \sum_{j=1}^{N} u_{ij} < N \), \( 0 \leq w_{ij} \leq 1 \), and \( \sum_{k=1}^{D} w_{ik} = 1 \).

Similarly, when \( m \to 1 \) and \( \eta = 0 \), the EFWSSC algorithm degenerates into the FSC algorithm, i.e., the FSC algorithm is as a special case of the EFWSSC algorithm.

(3) IEWKM

By integrating the between-cluster separation, Li et al. proposed the Improved Entropy Weighting K-means (IEWKM) with the following objective function [43],
Although the IEWKM algorithm here and the ESSC algorithm [41] are both based on EWKM and they both employ the between-cluster separation, the former is much more difficult to be optimized. Besides, from the viewpoint of optimization, the learning rules for the cluster centers in the IEWKM algorithm lack rigorous derivation [43].

5.2 Multi-view Learning

Multi-view learning is becoming a popular approach in the fields of machine learning for learning on multi-view data. It has been used for SSC recently and two representative multi-view SSC algorithms are introduced as follows.

(1) TW-k-means

Chen et al. proposed the Two-level variable Weighting k-means (TW-k-means) by introducing the view weighting mechanism [44]. It is a multi-view SSC algorithm based on traditional weighting clustering methods. The objective function of TW-k-means is given by

\[
J_{\text{TW-k-means}}(U, V, W) = \frac{\sum_{i=1}^{C} \sum_{j=1}^{N} \sum_{k=1}^{T} u_{ij} \sum_{k'=1}^{D} w_{ik} (x_{jk} - v_{ik})^2}{\sum_{i=1}^{C} \sum_{j=1}^{N} \sum_{k=1}^{T} w_{ik} (v_{ik} - v_{ok})^2} + \gamma \frac{\sum_{i=1}^{C} \sum_{k=1}^{T} w_{ik} \ln w_{ik}}{1}
\]

(15)

In this algorithm, feature weighting is performed in the first level weighting where the importance of features in each view is adjusted. View weighting is then performed in the second level weighting where the influence of each view in the clustering procedure is adjusted.

(2) FG-k-means

Another multi-view SSC algorithm to be introduced is the Feature Groups weighting k-means (FG-k-means) algorithm [45]. It is based on the EWKM algorithm and the corresponding objective function is given by

\[
J_{\text{FG-k-means}}(U, V, W, \bar{W}) = \sum_{i=1}^{C} \sum_{j=1}^{N} \sum_{k=1}^{T} u_{ij} \bar{w}_i \sum_{k'=1}^{D} w_{ik} (x_{jk} - v_{ik})^2 + \gamma \frac{\sum_{i=1}^{C} \sum_{k=1}^{T} w_{ik} \ln w_{ik}}{1} + \gamma' \sum_{i=1}^{C} \bar{w}_i \ln \bar{w}_i
\]
s.t. \( u_j \in \{0,1\}, \sum_{j=1}^C u_j = 1, \ 0 < \sum_{j=1}^N u_j < N, \ 0 \leq w_{ik} \leq 1, \) and \( \sum_{k \in T_t} w_{ik} = 1, \ t = 1, \ldots, T, \ 0 \leq \tilde{w}_i \leq 1, \)
\[ \sum_{i=1}^T \tilde{w}_i = 1. \]

By comparing the FG-k-means and TW-k-means algorithms, it can be seen that former is indeed an extension of the latter from CSSC to ISSC.

### 5.3 Evolutionary Learning

Most of the existing SSC algorithms make use of iterative learning strategy for optimizing the objective functions. However, these algorithms suffer from poor initialization sensitivity and local optimization of solutions. In order to overcome the deficiencies, evolutionary learning technique has been introduced to optimize the objective functions of SSC. Several representative XSSC algorithms developed based on evolutionary learning are reviewed below.

**(1) Coevolutionary SSC**

Gangrski et al. proposed two SSC algorithms based on coevolution learning [46]. The first algorithm was inspired by the Lamarck theory and used the distance-based cost function defined in the AWC algorithm [30] as the fitness function. The second algorithm employed a fitness function based on a new partitioning quality measure. The experimental results in [46] highlighted the benefits of using coevolutionary feature weighting methods to improve the knowledge discovery process.

**(2) PSOVW**

Lu et al. proposed the Particle Swarm Optimizer for Variable Weighting (PSOVW) algorithm by using the particle swarm optimizer as the evolutionary strategy for SSC. The following objective function is employed for variable optimization [47],

\[
J_{PSOVW}(U,V,W) = \sum_{i=1}^C \sum_{j=1}^N u_j \sum_{k \in T_t} \frac{w_{ik}}{D_{ik}} (x_{ik} - v_{ik})^2
\]

s.t. \( u_j \in \{0,1\}, \sum_{j=1}^C u_j = 1, \ 0 \leq w_{ij} \leq 1. \)

By transforming the original constrained variable weighting problem into a problem with bound constraints, i.e., using a normalized representation of variable weights, the particle swarm
optimizer can easily minimize the objective function to search for the global optima for the variable weighting problem. Experimental results show that the PSOVW algorithm greatly improves the clustering performance and the clustering results are much less dependent on the initial cluster centers.

(3) MOEA-SSC

The Multi-Objective Evolutionary Approach for SSC (MOEA-SSC) is a multi-objective evolutionary algorithm developed by Xia et al. that makes use of new encoding and operators [48]. Here, two objective functions are adopted for SSC, i.e.,

$$J_{IN}(U, V, W) = \sum_{i=1}^{C} \sum_{j=1}^{N} u_{ij} w_{ik} (x_{ik} - v_{ik})^2$$

$$J_{Add}(U, V, W) = \sum_{i=1}^{C} \frac{A_i}{\sum_{i=1}^{D} (v_{ik} - v_{ik})^2} + \sum_{i=1}^{C} \sum_{k=1}^{D} w_{ik} \ln w_{ik}$$

with $A_i = \frac{\sum_{k=1}^{D} w_{ik} \delta_k}{\sum_{i=1}^{D} \delta_i}$, $\delta_k = \begin{cases} 1 & \text{if } w_{ik} > 1/D \\ 0 & \text{Otherwise} \end{cases}$

s.t. $0 \leq u_{ij} \leq 1$, $\sum_{i=1}^{C} u_{ij} = 1$, $0 < \sum_{j=1}^{N} u_{ij} < N$, $0 \leq w_{ij} \leq 1$, and $\sum_{k=1}^{D} w_{ik} = 1$,

where $J_{IN}$ denotes the within-cluster dispersion, and $J_{Add}$ contains both the information of the negative weight entropy and the separation between clusters. Similar to other evolutionary learning based SSC algorithms, MOEA-SSC algorithm is less dependent on the initial cluster centers.

5.4 Adaptive Metric

In SSC, Euclidean distance is commonly used as the metric for defining the distance between the data points and the cluster centers in each dimension. In order to improve clustering performance, a number of modified metrics have been proposed. Two of the metrics are introduced below, along with the corresponding XSSC algorithms thus developed.

(1) Kernel Metric

Shen et al. develop an improved version of fuzzy weighting SSC algorithm, namely, the
Weighted Fuzzy Kernel-Clustering Algorithm (WFKCA), by proposing a new metric in the Mercer kernel space, i.e., Mercer kernel distance. This metric is used instead of the Euclidean distance in the original space for each dimension. The objective function of WFKCA can be formulated as follows,

$$J_{WFKCA}(U,V,W) = \sum_{i=1}^{C} \sum_{j=1}^{N} \sum_{k=1}^{D} w_{ik}^{j} \left\| \phi(x_{jk}) - \phi(v_{ik}) \right\|^2$$

where \( \phi(x_{jk}) \) is the mapped feature vector of feature \( x_{jk} \) and \( k(x_{jk},x_{jk}) \). With Mercer kernel distance as metric, WFKCA is able to improve the clustering performance and adaptive abilities of the algorithm when compared with the traditional Euclidean distance based algorithms.

**2) Minkowski Metric**

Amorim and Mirkin also attempted to improve fuzzy weighting SSC algorithms by employing an alternative metric, where the Minkowski metric was used to replace Euclidean distance to develop the Minkowski metric Weighted K-Means (MWK-Means) algorithm. The objective function of MWK-Means algorithm is as follows,

$$J_{MWK-Means}(U,V,W) = \sum_{i=1}^{C} \sum_{j=1}^{N} \sum_{k=1}^{D} w_{ik}^{j} \left\| x_{jk} - v_{ik} \right\|^p,$$

where the traditional Euclidean distance is replaced by the Minkowski metric with \( p \) as the parameter. By comparing MWK-Means with AWA [30], we can easily see that when \( p = 2 \), AWA is indeed a special case of MWK-Means.

**5.5 Imbalanced Clusters**

For data with imbalanced cluster, a XSSC algorithm, called the Weighted LAC (WLAC), is proposed based on elite selection of weighted clusters [51]. The objective function of WLAC can be expressed as follows,

$$J_{WLAC}(U,V,W,d) = \sum_{i=1}^{C} d_i \sum_{k=1}^{D} w_{ik} \lambda_i + \gamma_1 \sum_{i=1}^{C} \sum_{k=1}^{D} w_{ik} \ln w_{ik} + \gamma_2 \sum_{i=1}^{C} d_i \ln d_i,$$
\[ X_{ik} = \left( \frac{\sum_{j=1}^{N} u_{ij} (x_{ik} - v_{ik})^2}{\sum_{j=1}^{N} u_{ij}} \right)^{1/2} \]

s.t. \( u_{ij} \in \{0,1\} \), \( \sum_{j=1}^{C} u_{ij} = 1 \), \( 0 < \sum_{j=1}^{N} u_{ij} < N \), \( 0 \leq w_{ij} \leq 1 \), and \( \sum_{k=1}^{D} w_{ik} = 1 \),

\[ 0 \leq d_i \leq 1, \sum_{i=1}^{C} d_i = 1, \]

where \( d_i \) is the co-efficient representing the diameter of the \( i \)th cluster and is used to balance the influence of imbalanced clusters. In particular, strategies such as ensemble learning have also been studied to reduce the influence of the parameters \( \gamma_1 \) and \( \gamma_2 \) in the WLAC algorithm.

### 5.6 Other XSSC Algorithms

In addition to the five subcategories of XSSC algorithms discussed above, many other XSSC algorithms have also been reported recently [52-59] and are described briefly as follows.

**(1) Category Data**

While the IEWKM algorithm is first proposed for category and numeric mixture data [52], algorithms developed specifically for this type of data have also been made available. For example, a modified version of EWKM [53] was proposed by Ahmad and Dey for handling category and numeric mixture data. Bai et al. also proposed a modified SSC algorithm based on AWA and the use of an improved metric for category data [54].

**(2) Ensemble Learning**

Ensemble learning is a useful technique to enhance clustering performance by fusing different clustering results on the same data set. Domeniconi and Al-razgan proposed a graph-partitioning-based clustering ensemble approach called Weighted Similarity Partitioning Algorithm (WSPA) to overcome the problem of parameter sensitivity of LAC algorithm [55]. The WSPA is able to combine multiple clustering results obtained from different runs of the LAC SSC algorithm.

Besides, Gullo et al. proposed the Projective Clustering Ensembles (PCE) algorithms for ensemble learning of soft clustering [56]. The PCE algorithm was developed with two different formulations, namely, single-objective PCE and two-objective PCE. The former was implemented as an EM-like algorithm and called EM-PCE. The latter employed techniques similar to those in the realm of multi-objective evolutionary algorithms, and thus known as MOEA-PCE algorithm. Experimental evaluation shows that MOEA-PCE generally produces
higher quality projective consensus clustering results, while it is not as efficient as the EM-PCE algorithm.

(3) Data Reliability

Boonigoen et al. proposed a novel approach to SSC based on the measure of data reliability, named as reliability-based SSC algorithms [57]. This approach is advantageous in that it is applicable to various clustering algorithms, while existing wrappers are only suitable to FCM/K-means model and/or mixture model. Research has been conducted to make the reliability-based SSC algorithms more efficient and feasible for dealing with large data sets. Evaluation on gene expression data shows that the algorithms can improve their corresponding baseline techniques and outperform important soft and crisp subspace clustering methods.

6 Future Development

Following a comprehensive review of various SSC methods, the potential future development of SSC is discussed in this section. In particular, the discussions will be made from the perspectives of new SSC algorithms based on transfer learning and multi-view learning.

6.1 Transfer Learning

The effectiveness of traditional clustering algorithms depends on the sufficiency of data and information. Most of the algorithms fail when the data available is insufficient for performing the clustering tasks, which is not uncommon in real-world applications. This presents a major challenge for traditional clustering algorithms. A promising approach to deal with the issue is to capitalize transfer learning techniques to improve the performance of the clustering algorithms. Transfer learning and domain adaptive learning [60-64] are very attractive approaches in the fields of machine learning for their applicability in the real-world modeling and learning tasks, such as classification, regression and clustering, particularly in situations where the data of the current scene (also commonly named as the target domain) are insufficient, or are difficult to be used for a certain task, while some information of a related scene (also commonly named as the source domain) is available for the task.

The conceptual diagram in Fig. 1 illustrates the improvement in clustering performance by exploiting transfer learning. The inset in the left shows that the data in the target domain, delineating the situation where it is difficult to obtain satisfactory clustering result with the data using traditional no-transfer learning based clustering algorithms. However, if a reference scene,
i.e., the source domain, is available and the information is taken into consideration (right of Fig. 1), more promising clustering result can be obtained. For example, if it known that there are two related domains and the number of clusters in these two domains is expected to be the same, then the induced knowledge in the source domain, such as the cluster centers, can be employed to guide the clustering process in the target domain. It can be seen from this example that the realization of effective knowledge/information transfer mechanisms from the source domain to the target domain is the key to develop transfer clustering algorithms. This issue requires special attention and should be properly addressed in the development of effective transfer learning strategy for SSC. For example, the knowledge induced in the source domain, such as the cluster centers and/or the subspace of the clustering results, should be used appropriately to boost the clustering performance in the target domain.

Despite the potential of transfer learning and the extensive study for classification and regression tasks in recent years, the application of transfer learning for clustering remains very scarce, especially for high dimensional data. Therefore, research towards the development of transfer leaning based SSC algorithms is promising. It is expected to yield effective algorithms for high dimensional data clustering analyses where transfer learning is useful to enhance clustering performance.

6.2 Enhanced Multi-view Learning

Another direction of further development in SSC is to leverage multi-view learning for advanced clustering algorithms. Multi-view data are becoming popular in many real-world modeling tasks, such as multi-view visual recognition. The research presented in [44, 45] indicates that multi-view clustering is a very effective approach to improve clustering
performance for multi-view learning scenes.

However, the existing multi-view SSC algorithms are yet to meet the requirement of the real-world data mining tasks since the multi-view learning mechanisms adopted are still very simple and not effective enough for clustering the high dimensional datasets such as gene expression datasets and text dataset. The development of effective multi-view learning mechanisms is thus an important research area and will make significant contribution to multi-view clustering analysis of high dimensional data.

6.3 Scalable Learning for Large Dataset

Another challenge to SSC is the fast and scalable learning abilities for the large datasets. As reviewed for the existing SSC algorithms, most of them are ineffective for clustering the large datasets due to high computational complexity. This problem is hampering further extensive applications of SSC. In order to overcome this challenge, some existing techniques about the scalable and fast learning on large datasets in classical clustering algorithm, can be integrated into the SSC to design the effective algorithms. Related work can be seen in [66]. Some new advancement for the learning on the large datasets, such as the minimal enclosing ball approximation technique [67-71], can also be taken into account to develop the scalable and fast SSC algorithms.

7 Conclusions

In this paper, a comprehensive survey of SSC has been presented. A wide variety of existing algorithms are systematically classified into three main categories, i.e., CSSC, ISSC and XSSC. These three categories of SSC algorithms, along with the different subcategories have been reviewed and discussed in detail. We foresee that transfer learning and multi-view learning will play an important role in the development of SCC in the future. A thorough understanding of SSC algorithms and insights into the advancement of SSC can be obtained through this survey paper.

Acknowledgement

This work was supported in part by the National Natural Science Foundation of China (61170122, 61272210), the Ministry of education program for New Century Excellent Talents (NCET-120882), and the General Research Fund of Hong Kong RGC (PolyU 5134/12E).
References

[1] R. Xu, D. Wunsch, Survey of clustering algorithms, IEEE Trans. Neural Networks 16(3) (2005) 645-678.
[2] A.K. Jain, Data clustering: 50 years beyond K-means, Pattern Recognition Letters 31(8) (2010) 651-666.
[3] Y. Xiao, J. Yu, Partitive clustering (K-means family), Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery 2(3) (2012) 209-225.
[4] L. Parsons, E. Haque, H. Liu H, Subspace clustering for high dimensional data: a review, ACM SIGKDD Explorations Newsletter 6(1) (2004) 90-105.
[5] K. Sim, V. Gopalkrishnan, A. Zimek A, et al, A survey on enhanced subspace clustering, Data Mining and Knowledge Discovery 26(2) (2013) 332-397.
[6] H.P. Kriegel, P. Kröger, A. Zimek, Clustering high-dimensional data: A survey on subspace clustering, pattern-based clustering, and correlation clustering, ACM Transactions on Knowledge Discovery from Data 3(1) (2009) Article No. 1.
[7] H.P. Kriegel, P. Kröger, A. Zimek, Subspace clustering, Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 2(4) (2012) 351-364.
[8] R. Agrawal, J. Gehrke, D. Gunopulos and P. Raghavan, Automatic subspace clustering of high dimensional data for data mining applications, Proceedings of ACM SIGMOD Int'l Conf. Management of Data, 1998, pp. 94-105.
[9] C.H. Cheng, A.W. Fu, Y. Zhang, Entropy-based subspace clustering for mining numerical data, Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, 1999, pp. 84-93.
[10] S. Goil, H. Nagesh, A. Choudhary, MAFIA: Efficient and scalable subspace clustering for very large data sets, Proceedings of the 5th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 1999, pp. 443-452.
[11] C.C. Aggarwal and P.S. Yu, Finding generalized projected clusters in high dimensional spaces, in: Proc. ACM SIGMOD Int'l Conf. Management of Data, 2000, pp. 70-81.
[12] K.G. Woo, J.H. Lee, M.H. Kim, et al, FINDIT: a fast and intelligent subspace clustering algorithm using dimension voting, Information and Software Technology 46(4) (2004) 255-271.
[13] C.M. Procopiuc, M. Jones, P.K. Agarwal, et al, A Monte Carlo algorithm for fast projective clustering, Proceedings of the 2002 ACM SIGMOD international conference on Management of data, ACM, 2002, pp. 418-427.
[14] J. Yang, W. Wang, H. Wang, et al, δ-clusters: Capturing subspace correlation in a large data set, Proceedings. 18th International Conference on Data Engineering, IEEE, 2002: 517-528.
[15] C.C. Aggarwal, J.L. Wolf, P.S. Yu, et al, Fast algorithms for projected clustering, ACM SIGMOD Record, 28(2) (1999) 61-72.
[16] K.Y. Yip, D.W. Cheung, M.K. Ng, Harp: A practical projected clustering algorithm, IEEE Trans. Knowledge and Data Engineering 16(11) (2004) 1387-1397.
[17] K. Chakrabarti, S. Mehrotra, Local dimensionality reduction: A new approach to indexing high dimensional spaces, Proceedings of the 26th International Conference on Very Large Data Bases. 2000, pp. 89-100.
[18] G. De Soete, Optimal variable weighting for ultrametric and additive tree clustering, Quality and Quantity, 20(2-3) (1986) 169-180.
[19] V. Makarenkov, P. Legendre, Optimal variable weighting for ultrametric and additive trees and K-means partitioning: Methods and software, Journal of Classification 18(2) (2001) 245-271.
[20] D.S. Modha, W.S. Spangler, Feature weighting in k-means clustering, Machine learning 52(3) (2003) 217-237.
[21] X. Wang, Y. Wang, L. Wang, Improving fuzzy c-means clustering based on feature-weight learning, Pattern Recognition Letters 25(10) (2004) 1123-1132.
[22] W.S. DeSarbo, J.D. Carroll, L.A. Clark, et al, Synthesized clustering: A method for amalgamating alternative clustering bases with differential weighting of variables, Psychometrika 49(1) (1984) 57-78.
[23] J. Li, X. Gao, L. Jiao, A new feature weighted fuzzy clustering algorithm, In: Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing. Springer Berlin Heidelberg, 2005, pp. 412-420.
[24] J.Z. Huang, M.K Ng, H. Rong, et al, Automated variable weighting in k-means type clustering, IEEE Trans. Pattern Analysis and Machine Intelligence 27(5) (2005) 657-668.
[25] Y. Cheung, H. Zeng, A maximum weighted likelihood approach to simultaneous model selection and feature weighting in Gaussian mixture, In: Artificial Neural Networks–ICANN 2007, Springer Berlin Heidelberg, 2007, pp. 78-87.
[26] C.Y. Tsai, C.C. Chiu, Developing a feature weight self-adjustment mechanism for a K-means clustering algorithm, Computational statistics & data analysis, 52(10) (2008) 4658-4672.
[27] N. Bouguila, A model-based approach for discrete data clustering and feature weighting using MAP and stochastic complexity, IEEE Trans. Knowledge and Data Engineering 21(12) (2009) 1649-1664.
[28] A. Keller, F. Klauwonn, Fuzzy clustering with weighting of data variables, International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems 8(06) (2000) 735-746.
[29] H. Frigui, O. Nasraoui, Unsupervised learning of prototypes and attribute weights, Pattern recognition 37(3) (2004) 567-581.
[30] E.Y. Chan, W.K. Ching, M.K. Ng, et al, An optimization algorithm for clustering using weighted dissimilarity measures, Pattern recognition 37(5) (2004) 943-952.
[31] L. Jing, M.K. Ng, J. Xu, et al, Subspace clustering of text documents with feature weighting k-means algorithm, In: Advances in Knowledge Discovery and Data Mining, Springer Berlin Heidelberg, 2005, pp. 802-812.
[32] G. Gan, J. Wu, Z. Yang, A fuzzy subspace algorithm for clustering high dimensional data, In: Advanced Data Mining and Applications. Springer Berlin Heidelberg, 2006, pp. 271-278.
[33] G. Gan, J. Wu, A convergence theorem for the fuzzy subspace clustering (FSC) algorithm, Pattern Recognition 41(6) (2008) 1939-1947.
[34] L. Jing, M.K. Ng, J.Z. Huang, An entropy weighting k-means algorithm for subspace clustering of high-dimensional sparse data, IEEE Trans. Knowledge and Data Engineering 19(8) (2007) 1026-1041.
[35] C. Domeniconi, D. Gunopulos, S. Ma, et al, Locally adaptive metrics for clustering high dimensional data, Data Mining and Knowledge-Based Systems 14(1) (2007) 63-97.
[36] C. Domeniconi, D. Papadopoulos, D. Gunopulos, et al, Subspace clustering of high dimensional data, SIAM Int. Conf. on Data Mining, 2004.
[37] L. Chen, Q. Jiang, S. Wang, A probability model for projective clustering on high dimensional data, Eighth IEEE International Conference on Data Mining, 2008, IEEE, 2008, pp. 755-760.
[38] L. Chen, Q. Jiang, S. Wang, Model-based method for projective clustering, IEEE Trans. Knowledge and Data Engineering 24(7) (2012) 1291-1305.
[39] L. Peng, J. Zhang, An entropy weighting mixture model for subspace clustering of high-dimensional data, Pattern Recognition Letters 32(8) (2011) 1154-1161.
[40] J.H. Friedman, J.J. Meulman, Clustering objects on subsets of attributes (with discussion), Journal of the Royal Statistical Society: Series B (Statistical Methodology) 66(4) (2004) 815-849.
[41] Z. Deng, K.S. Choi, F.L. Chung, et al, Enhanced soft subspace clustering integrating within-cluster and
between-cluster information, Pattern Recognition 43(3) (2010) 767-781.

[42] Q. Guan, The study on enhanced soft subspace clustering techniques, Master Dissertation, 2011.

[43] T. Li, Y. Chen, An improved k-means algorithm for clustering using entropy weighting measures, the 7th World Congress on Intelligent Control and Automation, IEEE, 2008, pp. 149-153.

[44] X. Che, X. Xu, J. Huang, et al, TW-k-means: Automated Two-level Variable Weighting Clustering Algorithm for Multi-View 25(4) (2013) 932-944.

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

[46] P. Gançarski, A. Blansche, A. Wania, Comparison between two coevolutionary feature weighting algorithms in clustering, Pattern Recognition 41(3) (2008) 983-994.

[47] Y. Lu, S. Wang, S. Li, et al, Particle swarm optimizer for variable weighting in clustering high-dimensional data, Machine learning 82(1) (2011) 43-70.

[48] L. Zhu, L. Cao, J. Yang, Multiobjective evolutionary algorithm-based soft subspace clustering, 2012 IEEE Congress on Evolutionary Computation, IEEE, 2012, pp. 1-8.

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

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

[51] H. Parvin, B. Minaei-Bidgoli, A clustering ensemble framework based on elite selection of weighted clusters, Advances in Data Analysis and Classification (2013) 1-28.

[52] T. Li, Y. Chen, A weight entropy k-means algorithm for clustering dataset with mixed numeric and categorical data, Fifth International Conference on Fuzzy Systems and Knowledge Discovery, IEEE, 2008, vol. 1, pp. 36-41.

[53] A. Ahmad, L. Dey, A K-means type clustering algorithm for subspace clustering of mixed numeric and categorical datasets, Pattern Recognition Letters 32(7) (2011) 1062-1069.

[54] L. Bai, J. Liang, C. Dang, et al, A novel attribute weighting algorithm for clustering high-dimensional categorical data, Pattern Recognition 44(12) (2011) 2843-2861.

[55] C. Domeniconi, M. Al-Razgan, Weighted cluster ensembles: Methods and analysis, ACM Transactions on Knowledge Discovery from Data 2(4) (2009) Article No. 17.

[56] F. Gullo, C. Domeniconi, A. Tagarelli, Projective clustering ensembles, Data Mining and Knowledge Discovery 26(3) (2013) 452-511.

[57] T. Boongoen, C. Shang, N. Iam-On, et al, Extending data reliability measure to a filter approach for soft subspace clustering, IEEE Trans. Systems, Man, and Cybernetics, Part B: Cybernetics 41(6) (2011) 1705-1714.

[58] G. Guo, S. Chen, L. Chen, Soft subspace clustering with an improved feature weight self-adjustment mechanism, International Journal of Machine Learning and Cybernetics 3(1) (2012) 39-49.

[59] J Wang, S. Wang S, Z. Deng, et al, Double indices induced FCM clustering and its integration with fuzzy subspace clustering, 2012 IEEE International Conference on Fuzzy Systems, IEEE, 2012, pp. 1-8.

[60] S.J. Pan, Q. Yang, A survey on transfer learning, IEEE Trans. Knowledge and Data Engineering 22(10) (2010) 1345-1359.

[61] X. Gao, X. Wang, X. Li, et al, Transfer latent variable model based on divergence analysis, Pattern Recognition 44(10) (2011) 2358-2366.

[62] I.H. Jhuo, D. Liu, D.T. Lee, et al, Robust visual domain adaptation with low-rank reconstruction, 2012 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2012, pp. 2168-2175.
[63] L. Duan, D. Xu, S.F. Chang, Exploiting web images for event recognition in consumer videos: A multiple source domain adaptation approach, 2012 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2012, pp. 1338-1345.

[64] R. Gopalan, R. Li, R. Chellappa, Domain adaptation for object recognition: An unsupervised approach, 2011 IEEE International Conference on Computer Vision, IEEE, 2011, pp. 999-1006.

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

[66] T.C. Havens, J.C. Bezdek, C. Leckie, L.O. Hall and M. Palaniswami, Fuzzy c-means for very large data, IEEE Trans. Fuzzy Systems 20(6) (2012) 1130-1146.

[67] I.W. Tsang, J.T. Kwo and P.M. Cheung, Core vector machines: fast SVM training on large data sets, Journal of Machine Learning Research 6 (2005) 363-392.

[68] I.W. Tsang, A. Kocsor, and J. T. Kwok, Large-scale maximum margin discriminant analysis using core vector machines, IEEE Trans. Neural Networks 19(4) (2008) 610-624.

[69] Z. H. Deng, F. L. Chung and S. T. Wang, FRSDE: fast reduced set density estimator using minimal enclosing ball approximation, Pattern Recognition 41(4) (2008) 1363-1372.

[70] F. L. Chung, Z. H. Deng, and S. T. Wang, From minimum enclosing ball to fast fuzzy inference system training on large datasets, IEEE Trans. Fuzzy Systems 17(1) (2009) 73-184.

[71] Z. H. Deng, K. S. Choi, F. L. Chung, S.T. Wang, Scalable TSK fuzzy modeling for very large datasets using minimal enclosing ball approximation, IEEE. Trans. Fuzzy systems 19(2) (2011) 210-226.