Postprint of Overlapping Subspace Clustering Algorithms under Sparse Conditions

Authors: Qiu Yunfei, Fei Bowen, Liu Daqian, Liu Xing

Date: 2018-05-18T00:00:00+00:00

Abstract

Existing subspace clustering algorithms cannot effectively balance the relationship between the denseness of data within subspaces and the sparsity of data across different subspaces, nor can they handle data overlap issues. To address these problems, we propose an Overlapping Subspace Clustering under Sparse Conditions (OSCSC) algorithm. The algorithm constructs a subspace representation model utilizing a mixed-norm representation method combining L1 norm and Frobenius norm, and applies weighted processing to the L1 norm regularization term to enhance the sparsity across different subspaces and the denseness within the same subspace. Subsequently, it performs secondary validation on the partitioned subspaces using an overlapping probability model that conforms to an exponential family distribution to determine the overlap conditions among data from different subspaces, thereby further improving clustering accuracy. Experiments on both synthetic and real datasets demonstrate that the OSCSC algorithm achieves favorable clustering results.

Full Text

Preamble

Novel Algorithm for Overlapping Subspace Clustering Under Sparse Conditions

Qiu Yunfei^a, Fei Bowen^b†, Liu Daqian^c, Liu Xing^a

^a School of Software, ^b School of Business & Management, ^c School of Electronics & Information Engineering
Liaoning Technical University, Huludao, Liaoning 125105, China

Abstract: Existing subspace clustering algorithms cannot effectively balance the density of data within the same subspace and the sparsity of data across different subspaces, nor can they handle data overlap. To address these issues, this

paper proposes an Overlapping Subspace Clustering under Sparse Conditions (OSCSC) algorithm. The algorithm constructs a subspace representation model using a mixed norm representation of the ℓ_1 norm and Frobenius norm, and applies weighted regularization to the ℓ_1 norm term to enhance the sparsity across different subspaces and the density within the same subspace. Subsequently, an overlapping probability model conforming to an exponential family distribution is applied to the partitioned subspaces for secondary verification, determining the overlap status of data across different subspaces to further improve clustering accuracy. Experiments on both synthetic and real-world datasets demonstrate that the OSCSC algorithm achieves favorable clustering performance.

Keywords: overlapping subspace clustering; mixed norm; overlapping probability model; exponential family distribution

Introduction

Clustering analysis is a crucial research area in data mining with widespread applications in machine learning, biomedical analysis, and computer vision. In recent years, data scale has grown rapidly, with increasing data dimensionality. When processing and analyzing such datasets, traditional clustering methods often fail to obtain accurate results due to sparse sample distributions where inter-sample distances become nearly identical. To address issues of large scale and high dimensionality, Agrawal et al. first applied the concept of subspace clustering to clustering problems. Since then, numerous subspace clustering methods have been proposed by scholars and researchers both domestically and internationally.

Existing subspace clustering methods can be broadly categorized into five classes: iterative methods, algebraic methods, statistical methods, matrix factorization-based methods, and spectral clustering-based methods. Among these, spectral clustering-based subspace clustering methods have gained significant attention. These approaches construct similarity matrices based on self-expression models by finding representation coefficients in low-dimensional spaces, then apply spectral clustering algorithms to obtain final clustering results. Both Sparse Subspace Clustering (SSC) and Least Squares Regression (LSR) exhibit this property, assuming each data point in the entire space can be linearly represented by other data points.

SSC solves the subspace clustering model as:

$$\min_{\mathbf{Z}} \|\mathbf{Z}\|_1 \quad \text{s.t.} \quad \mathbf{x}_j = \mathbf{X}\mathbf{z}_j, \quad \mathbf{z}_{jj} = 0$$

where $\|\cdot\|_1$ denotes the ℓ_1 norm and $\mathbf{X}$ is the data matrix. $\mathbf{Z}$ is the coefficient matrix composed of coefficients z_{ij}. For noisy data, SSC can be extended as:

$$\min_{\mathbf{Z}} \|\mathbf{Z}\|_1 + \frac{\lambda}{2}\|\mathbf{X} - \mathbf{X}\mathbf{Z}\|_F^2 \quad \text{s.t.} \quad \mathbf{z}_{jj} = 0$$

LSR establishes the following objective by minimizing the Frobenius norm of the coefficient matrix:

$$\min_{\mathbf{Z}} \|\mathbf{Z}\|_F^2 \quad \text{s.t.} \quad \mathbf{X} = \mathbf{XZ}$$

For noisy data, LSR can be extended as:

$$\min_{\mathbf{Z}} \|\mathbf{Z}\|_F^2 + \frac{\lambda}{2}\|\mathbf{X} - \mathbf{XZ}\|_F^2$$

Based on the symmetry and non-negativity of the similarity matrix, the similarity matrix $\mathbf{W}$ can be defined as:

$$\mathbf{W} = \frac{|\mathbf{Z}| + |\mathbf{Z}^T|}{2}$$

Spectral clustering is then applied to $\mathbf{W}$ to obtain clustering results.

Although these subspace clustering algorithms have improved clustering performance to some extent, they are all hard-partitioning methods that ignore the problem of overlap between data clusters. In practical subspace clustering problems, data from different subspaces exhibit overlapping regions and are not completely independent, causing some data to be incorrectly assigned and affecting clustering precision. Addressing data partitioning uncertainty, Banerjee et al. proposed a Model-based Overlapping Clustering (MOC) algorithm that uses probabilistic models to handle data overlap and determine whether a sample belongs to multiple clusters. Fu et al. proposed a Bayesian Overlapping Subspace Clustering (BOSC) algorithm that discovers overlapping structures by constructing a hierarchical generative model of the data matrix, though this approach does not fully exploit structural relationships between data and is inefficient for high-dimensional datasets.

2. Overlapping Subspace Clustering Under Sparse Conditions

The OSCSC algorithm proposed in this paper integrates subspace clustering and overlapping clustering concepts to address overlapping clustering problems across different subspaces. The algorithm employs an iteratively weighted mixed norm representation of ℓ_1 and Frobenius norms to construct a subspace clustering model, representing high-dimensional data through low-dimensional subspace linear representations and optimizing the model using linearized alternating direction methods. Subsequently, an overlapping probability model estimates data overlap status. Unlike previous hard-partitioning subspace clustering techniques, our algorithm allows a data point to belong to one or multiple subspaces, thereby improving clustering accuracy and reducing errors.

2.1 Weighted Mixed Norm Subspace Representation

2.1.1 Subspace Representation Model The core of subspace clustering lies in subspace model construction. Combining ideas from SSC and LSR, this

paper proposes a mixed norm subspace representation method that ensures inter-class sparsity while enhancing intra-class density. Define the data matrix $\mathbf{X} \in \mathbb{R}^{M\times N}$ containing N column vectors $\{\mathbf{x}_j\}_{j=1}^{N} \in \mathbb{R}^{M}$. Through subspace linear representation, the goal of subspace clustering is to assign each column vector $\mathbf{x}_j$ in $\mathbf{X}$ to its correct subspace. The subspace clustering model can be expressed as:

$$\min_{\mathbf{Z}} \lambda_1 \|\mathbf{Z}\|_1 + \frac{\lambda_2}{2}\|\mathbf{Z}\|_F^2 \quad \text{s.t.} \quad \mathbf{X} = \mathbf{X}\mathbf{Z}, \quad \mathbf{z}_{jj} = 0$$

where $\|\cdot\|_F$ denotes the Frobenius norm and λ_1, λ_2 are trade-off coefficients balancing the relationship between regularization terms. $\mathbf{Z}$ is the coefficient matrix providing conditions for subspace segmentation.

For noisy datasets, the model can be represented as:

$$\min_{\mathbf{Z}} \lambda_1 \|\mathbf{Z}\|_1 + \frac{\lambda_2}{2}\|\mathbf{Z}\|_F^2 + \frac{\lambda_3}{2}\|\mathbf{X} - \mathbf{X}\mathbf{Z}\|_F^2 \quad \text{s.t.} \quad \mathbf{z}_{jj} = 0$$

2.1.2 Weighting Scheme In real-world problems, relationships within datasets are complex, and relying solely on ℓ_1 regularization to ensure subspace sparsity is inadequate. Literature [10, 15] propose weighting methods for ℓ_1 regularization, with extensive experiments demonstrating that iteratively updating weights for the ℓ_1 norm ($\ell_{1,w}$) yields sparser coefficient structures compared to using the ℓ_1 norm alone, bridging the gap between ℓ_1 and ℓ_0 norms and making $\ell_{1,w}$ better approximate ℓ_0. Literature [14] obtains the weighting scheme through iterative updates:

$$w_i^{(t)} = \frac{1}{\epsilon + |x_i^{(t-1)}|}$$

where $w_i^{(t)}$ is the weight for data point x_i at iteration t and ϵ is a control parameter.

This paper applies the iteratively weighted ℓ_1 norm concept to the mixed norm subspace representation model, allowing Equation (10) to be expressed as:

$$\min_{\mathbf{Z}} \lambda_1 \|\mathbf{W} \odot \mathbf{Z}\|_1 + \frac{\lambda_2}{2}\|\mathbf{Z}\|_F^2 + \frac{\lambda_3}{2}\|\mathbf{X} - \mathbf{X}\mathbf{Z}\|_F^2 \quad \text{s.t.} \quad \mathbf{z}_{jj} = 0$$

where $\odot$ denotes element-wise multiplication.

2.1.3 Model Optimization The optimization problem in Equation (13) can be transformed into:

$$\min_{\mathbf{Z},\mathbf{E}} \lambda_1 \|\mathbf{W} \odot \mathbf{Z}\|_1 + \frac{\lambda_2}{2}\|\mathbf{Z}\|_F^2 + \frac{\lambda_3}{2}\|\mathbf{E}\|_F^2 \quad \text{s.t.} \quad \mathbf{X} = \mathbf{X}\mathbf{Z} + \mathbf{E}, \quad \mathbf{z}_{jj} = 0$$

We optimize this problem using the Linearized Alternating Directions Method (LADM) [16]. By introducing Lagrange multiplier $\mathbf{Y}$, we obtain the augmented Lagrangian function:

$$\mathcal{L}(\mathbf{Z},\mathbf{E},\mathbf{Y}) = \lambda_1\|\mathbf{W}\odot\mathbf{Z}\|_1+\frac{\lambda_2}{2}\|\mathbf{Z}\|_F^2+\frac{\lambda_3}{2}\|\mathbf{E}\|_F^2+\langle\mathbf{Y},\mathbf{X}-\mathbf{X}\mathbf{Z}-\mathbf{E}\rangle+\frac{\rho}{2}\|\mathbf{X}-\mathbf{X}\mathbf{Z}-\mathbf{E}\|_F^2 \quad \text{s.t.} \quad \mathbf{z}_{jj}=0$$

where $\rho > 0$ is the penalty parameter.

Using LADM to optimize Equation (15), let k denote the iteration number. Fixing $\mathbf{E}^{(k)}$, we minimize $\mathcal{L}$ with respect to $\mathbf{Z}$:

$$\mathbf{Z}^{(k+1)} = \arg\min_{\mathbf{Z}} \lambda_1\|\mathbf{W}\odot\mathbf{Z}\|_1+\frac{\lambda_2}{2}\|\mathbf{Z}\|_F^2+\langle\mathbf{Y}^{(k)},\mathbf{X}-\mathbf{X}\mathbf{Z}-\mathbf{E}^{(k)}\rangle+\frac{\rho}{2}\|\mathbf{X}-\mathbf{X}\mathbf{Z}-\mathbf{E}^{(k)}\|_F^2 \quad \text{s.t.} \quad \mathbf{z}_{jj}=0$$

This can be approximated by linearization:

$$\mathbf{Z}^{(k+1)} = \arg\min_{\mathbf{Z}} \lambda_1\|\mathbf{W}\odot\mathbf{Z}\|_1+\frac{\lambda_2}{2}\|\mathbf{Z}\|_F^2+\frac{\rho}{2}\|\mathbf{Z}-(\mathbf{Z}^{(k)}-\frac{1}{\rho}\nabla_{\mathbf{Z}}\mathcal{L}(\mathbf{Z}^{(k)}))\|_F^2 \quad \text{s.t.} \quad \mathbf{z}_{jj}=0$$

The solution involves the shrinkage threshold operator $\mathcal{S}_\tau(\cdot)$, defined as:

$$\mathcal{S}_\tau(\mathbf{Q}) = \text{sgn}(\mathbf{Q})\odot\max(|\mathbf{Q}|-\tau,0)$$

Let $\theta = \frac{1.1}{\sigma_{\max}}$ where $\sigma_{\max}$ is the maximum singular value of $\mathbf{X}$. The LADM optimization process proceeds as follows:

Algorithm 1: LADM Optimization for Equation (15) - **Input:** Data matrix $\mathbf{X}$, trade-off coefficients $\lambda_1, \lambda_2, \lambda_3$ - **Output:** Coefficient matrix $\mathbf{Z}$, noise matrix $\mathbf{E}$

Initialize: $\mathbf{Z}^{(0)} = \mathbf{0}$, $\mathbf{E}^{(0)} = \mathbf{0}$, $\mathbf{Y}^{(0)} = \mathbf{0}$, $\rho^{(0)} = 10^{-6}$, $\gamma = 1.6$, $\theta = \frac{1.1}{\sigma_{\max}}$, $\epsilon_1 = 10^{-4}$, $\epsilon_2 = 10^{-5}$

While not converged: 1. Update $\mathbf{Z}^{(k+1)}$ using the shrinkage operator 2. Update $\mathbf{E}^{(k+1)} = \arg\min_{\mathbf{E}} \frac{\lambda_3}{2}\|\mathbf{E}\|_F^2 + \langle\mathbf{Y}^{(k)},\mathbf{X}-\mathbf{X}\mathbf{Z}^{(k+1)}-\mathbf{E}\rangle + \frac{\rho}{2}\|\mathbf{X}-\mathbf{X}\mathbf{Z}^{(k+1)}-\mathbf{E}\|_F^2$ 3. Update Lagrange multiplier: $\mathbf{Y}^{(k+1)} = \mathbf{Y}^{(k)} + \rho(\mathbf{X}-\mathbf{X}\mathbf{Z}^{(k+1)}-\mathbf{E}^{(k+1)})$ 4. Update penalty parameter: $\rho^{(k+1)} = \min(\gamma\rho^{(k)}, 10^{10})$ 5. Check convergence: $\|\mathbf{X}-\mathbf{X}\mathbf{Z}^{(k+1)}-\mathbf{E}^{(k+1)}\|_F/\|\mathbf{X}\|_F < \epsilon_1$ and $\max(\|\mathbf{Z}^{(k+1)}-\mathbf{Z}^{(k)}\|_F/\|\mathbf{X}\|_F, \|\mathbf{E}^{(k+1)}-\mathbf{E}^{(k)}\|_F/\|\mathbf{X}\|_F) < \epsilon_2$

From the optimized coefficient matrix $\mathbf{Z}^*$, we obtain the similarity matrix $\mathbf{W} = \frac{|\mathbf{Z}^*|+|\mathbf{Z}^*|^T}{2}$, then apply the standard segmentation method Ncut [17] to partition the subspaces and obtain the subspace collection $\mathcal{S} = \{S_1, S_2, \ldots, S_L\}$.

2.2 Overlapping Probability Model

Although the weighted mixed norm subspace representation improves intra-subspace density and inter-subspace sparsity, errors persist in subspace clustering. Moreover, this method is a hard-partitioning approach that typically

allows a sample to belong to only one cluster, preventing correction when misclustering occurs. To address this, we employ an overlapping probability model to determine whether data in partitioned subspaces can belong to multiple subspaces.

Given a high-dimensional dataset $\mathbf{X}$, let the obtained L subspace collection be $\mathcal{S} = \{S_1, S_2, \ldots, S_L\}$, where each subspace represents a class. Let $\mathbf{y}_i$ be a data point from subspace S_l. The overlapping probability model is a probabilistic model conforming to an exponential family distribution [18], defined as distributions satisfying:

$$p(\mathbf{y}|\theta) = \exp(\mathbf{T}(\mathbf{y})^T\theta - \varphi(\theta))$$

where $\mathbf{T}(\mathbf{y})$ is the sufficient statistic, θ is the natural parameter, and $\varphi(\theta)$ is the cumulant function.

The conditional probability of the overlapping probability model across different subspaces can be expressed as:

$$p(\mathbf{y}_i|\mathbf{b}_i, \theta) = \prod_{l=1}^{L} p(\mathbf{y}_i|b_{il}, \theta_l)^{b_{il}}$$

where $\mathbf{b}_i$ is a Boolean vector (latent variable) indicating overlap status, with each element $b_{il} \in \{0, 1\}$ corresponding to a subspace. $c(\mathbf{b})$ is a normalization term. Define $\pi(\mathbf{b})$ as the prior of $\mathbf{b}$, where each element follows a Bernoulli distribution Bernoulli(ϕ_l), and the prior distribution follows Beta(α_l, β_l). Substituting into the joint probability yields:

$$p(\mathbf{y}_i, \mathbf{b}_i|\theta, \alpha, \beta) = p(\mathbf{y}_i|\mathbf{b}_i, \theta)\cdot p(\mathbf{b}_i|\alpha, \beta) = \prod_{l=1}^{L}\left[p(\mathbf{y}_i|\theta_l)^{b_{il}} \cdot \frac{\Gamma(\alpha_l + \beta_l)}{\Gamma(\alpha_l)\Gamma(\beta_l)}\phi_l^{\alpha_l - 1}(1 - \phi_l)^{\beta_l - 1}\right]$$

Outliers exist in datasets that do not belong to any cluster. While these values often cannot be ignored, they fall outside the overlapping probability model structure. Thus, Equation (19) can be expressed as:

$$p(\mathbf{y}_i|\theta) = \begin{cases} \pi_0 & \text{if } \mathbf{b}_i = \mathbf{0} \\ \sum_{\mathbf{b}_i \neq \mathbf{0}} p(\mathbf{y}_i|\mathbf{b}_i, \theta)p(\mathbf{b}_i) & \text{otherwise} \end{cases}$$

Each component of the overlapping probability model follows the same exponential family distribution. The conditional probability $p(\mathbf{y}_i|\mathbf{b}_i, \theta)$ has each component following an exponential family distribution with natural parameter $\sum_{l=1}^{L} b_{il}\theta_l$.

Since latent variable $\mathbf{b}$ is a set of Boolean vectors, $\pi(\mathbf{b})$ is its prior where each element follows Bernoulli(ϕ_l), and the prior distribution follows Beta(α_l, β_l). Substituting into Equation (21) yields the joint distribution of the probability model. The goal of overlapping subspace clustering is to determine whether data clusters in different subspaces overlap. By representing high-dimensional

data through low-dimensional subspace linear representation, we obtain subspaces with dense and similar data points. The overlapping probability model then judges overlap between these subspaces. This secondary verification allows misclustered data to be checked against other subspaces, improving clustering accuracy.

2.3 Parameter Estimation

Parameter estimation uses an alternating maximization algorithm [18] to estimate parameters in the subspace overlapping probability model, consisting of two parts: Boolean vector selection and parameter estimation.

Boolean Vector Selection: Given parameter values (α, β, θ), we optimize to obtain maximum log-likelihood using $\mathbf{b}$. Initialize $\mathbf{b}^{(0)}$. For any data point, let $\mathbf{v}_l$ be the initial assignment vector (abbreviated as $\mathbf{v}$), defined as a Boolean vector where the l-th element is 1 and others are 0, forming the set $\{\mathbf{v}_1, \ldots, \mathbf{v}_L\}$. The iterative computation of Boolean vectors is divided into L layers, using a fast heuristic iterative method [13] to compute the optimal solution $\mathbf{b}_l^*$ for each layer. Simulated annealing is employed during Boolean vector selection to escape local optima.

For each layer's selected Boolean vector, set an initial temperature parameter T_0. The Boolean vector can be expressed as $\mathbf{b}_{il}^{(t)}$, with η as a multiplicative factor ensuring temperature parameter T decreases at each iteration, and maximum iterations J. When obtaining a new layer' s Boolean value, iterative judgment is required. Compare the newly searched Boolean value $\mathbf{b}_{il}^{\text{new}}$ with $\mathbf{b}_{il}^{(t)}$. If $f(\mathbf{b}_{il}^{\text{new}}, T) < f(\mathbf{b}_{il}^*, T)$ or maximum iterations are reached, the layer iteration terminates. Select the optimal Boolean vector set $\mathbf{b}_i^*$ for each layer. (Parameters in this paper: $T_0 = 50$, $\eta = 0.67$, $J = 40$).

When $b_{il} = 1$, the sample is considered to belong to its corresponding subspace; when $b_{il} = 0$, it does not. When a Boolean vector contains two or more elements equal to 1, the sample is considered overlapping data that can belong to multiple subspaces.

Parameter Estimation: Given the Boolean vector set $\mathbf{b}$, let n_1 denote the count of 1s in $\mathbf{b}$ and n_0 denote the count of 0s. The optimal Beta distribution parameters satisfy:

$$\alpha_l^* = \frac{n_1}{n_1 + n_0 - 1}, \quad \beta_l^* = \frac{n_0}{n_1 + n_0 - 1}$$

Parameter θ estimation uses the second part of the log-likelihood function from Equation (26) to compute the optimal extreme value. The specific derivation is provided in Appendix 1.

2.4 Algorithm Description

Algorithm 2: OSCSC - **Input:** Dataset $\mathbf{X}$, trade-off coefficient λ, number of clusters L, initial temperature T_0 - **Output:** Clustering results

1. Obtain the weighted ℓ_1 and Frobenius mixed norm subspace representation using Equation (13)
2. Optimize using Algorithm 1 to obtain $\mathbf{Z}^*$ and similarity matrix $\mathbf{W}$
3. Apply Ncut segmentation to obtain subspace collection $\mathcal{S}$
4. Use the overlapping probability model to obtain joint distribution functions
5. For each iteration, optimize $\mathbf{b}$ and estimate parameters (α, β, θ)
6. Use simulated annealing to search for optimal solutions for Boolean vectors selected at each layer
7. Obtain final Boolean vector values $\mathbf{b}^*$ and determine overlap status based on $\mathbf{b}^*$

3. Experimental Results and Analysis

To verify the effectiveness of OSCSC, we compare it with five clustering algorithms: SSC, LSR, RSSC, MOC, and BOSC. All algorithms are implemented in MATLAB R2016a. Experiments use clustering accuracy (AC) [19-20], normalized mutual information (NMI) [21], and running time as evaluation metrics. For reliability, each algorithm is run independently 10 times.

AC is calculated as:

$$\mathrm{AC} = \frac{1}{N}\sum_{i=1}^{N}\delta(s_i, \mathrm{map}(r_i))$$

where N is the total number of samples, $\delta(\cdot)$ equals 1 when its two parameters are equal and 0 otherwise, s_i is the original class label, r_i is the clustering result label, and $\mathrm{map}(\cdot)$ maps clustering results to equivalent original classes.

NMI is calculated as:

$$\mathrm{NMI} = \frac{\sum_{i=1}^{c}\sum_{j=1}^{c} n_{ij}\log\left(\frac{n_{ij}N}{n_i n_j}\right)}{\sqrt{\left(\sum_{i=1}^{c} n_i \log\frac{n_i}{N}\right)\left(\sum_{j=1}^{c} n_j \log\frac{n_j}{N}\right)}}$$

where N is the total number of samples, c is the number of clusters, n_i and n_j are the numbers of samples in cluster i and j, and n_{ij} is the number of shared samples between clusters i and j.

3.1 Synthetic Dataset Results

We generate two synthetic datasets, dataset1 and dataset2, using the method from literature [22]. Dataset1 contains 500 samples, 4 clusters, and 30 dimensions. Dataset2 contains 3,000 samples, 6 clusters, and 80 dimensions. To better approximate real data, we define overlapping samples between different classes

to test algorithms' ability to discover overlapping clusters. Dataset information is shown in Table 1 .

The trade-off coefficient λ balances the ℓ_1 and Frobenius norms. We vary λ to achieve optimal clustering. Figure 1 [Figure 1: see original paper] shows the relationship between λ and clustering accuracy on both datasets, revealing optimal values of $\lambda = 0.8$ and $\lambda = 0.85$, respectively.

Experimental results on both synthetic datasets are shown in Tables 2 and 3 . MOC, BOSC, and OSCSC achieve higher accuracy than hard-partitioning methods (SSC, LSR, RSSC). However, MOC and BOSC cannot fully exploit spatial information when processing large, sparse, high-dimensional datasets, resulting in poor efficiency. OSCSC reduces the difficulty of direct overlapping clustering on sample sets by partitioning high-dimensional space into low-dimensional subspace collections using weighted mixed norm representation, effectively handling datasets of certain scale and dimensionality.

To further verify OSCSC' s effectiveness on noisy data, we test all algorithms with varying noise proportions (10%, 20%, 30%, 40%, 50%) on both synthetic datasets, with noise positions randomly selected. Results in Table 4 show that OSCSC handles various noise levels effectively, with minimal accuracy degradation as noise increases.

3.2 Real-World Dataset Results

We evaluate algorithm performance on six real-world datasets (Table 5): musk, soybean, waveform, and pendigits from UCI; USPS handwritten digits; and AR face dataset. For USPS, we randomly select 100 images per class (1,000 total). For AR, we randomly select 960 images from 80 individuals, downsampled to 32×24.

The trade-off coefficient λ significantly affects results. Figure 2 [Figure 2: see original paper] shows the λ-accuracy relationship across six datasets, with optimal values: musk ($\lambda = 0.75$), soybean ($\lambda = 0.95$), waveform ($\lambda = 0.85$), pendigits ($\lambda = 0.8$), AR ($\lambda = 0.7$), and USPS ($\lambda = 0.8$).

Tables 6 and 7 present AC and NMI results. OSCSC achieves favorable performance across all datasets. SSC, LSR, and RSSC are hard-partitioning methods that cannot correct errors during clustering. MOC, BOSC, and OSCSC are soft-partitioning methods allowing multi-class membership, but MOC and BOSC perform poorly on large-scale, high-dimensional data. OSCSC first partitions data into subspaces using weighted mixed norm representation, ensuring intra-subspace density and inter-subspace sparsity, then applies overlapping probability model verification. This avoids direct matching in high-dimensional space, efficiently discovering overlapping samples and correcting misassignments.

Table 8 shows average running times. LSR is fastest, while OSCSC' s runtime is reasonable compared to MOC and BOSC, especially on large datasets. OSCSC' s efficiency comes from performing overlap judgment on partitioned subspaces

rather than the entire space. However, for datasets with many overlaps (e.g., pendigits) or many classes (e.g., soybean, AR), runtime increases due to extensive overlap processing.

4. Conclusion

The OSCSC algorithm employs a weighted mixed norm representation of ℓ_1 and Frobenius norms to construct a subspace model, representing high-dimensional data through low-dimensional subspace linear combinations to enhance intra-subspace density and inter-subspace sparsity. An overlapping probability model judges overlap within partitioned subspaces, with parameters estimated via alternating maximization and simulated annealing to find global optima, further improving accuracy by correctly assigning data to subspaces. Experiments on synthetic and real datasets demonstrate OSCSC' s favorable performance. Future work will focus on improving the efficiency of overlapping subspace clustering algorithms.

References

[1] Elhamifar E, Vidal R. Sparsity in unions of subspaces for classification and clustering of high-dimensional data [C]// Proc of the 49th Annual Allerton Conference on Communication, Control, and Computing. 2011: 1085-1089.

[2] Peng X, Tang H J, Zhang L, et al. A unified framework for representation-based subspace clustering of out-of-sample and large-scale Data [J]. IEEE Trans on Neural Networks and Learning Systems, 2016, 27 (12): 2499-2512.

[3] 陈爱国, 王士同. 基于多代表点的大规模数据模糊聚类算法 [J]. 控制与决策, 2016, 31 (12): 2122-2130.

[4] Hu H, Lin Z C, Feng J J, et al. Smooth representation clustering [C]// Proc of IEEE International Conference on Computer Vision and Pattern Recognition. 2014: 3834-3841.

[5] Agrawal R, Geherke J, Gunopulos D. et al. Automatic subspace clustering of high dimensional data [J]. Data Mining and Knowledge Discovering, 2005, 11 (1): 5-33.

[6] 王卫卫, 李小平, 冯象初, 等. 稀疏子空间聚类综述 [J]. 自动化学报, 2015, 41 (8): 1373-1384.

[7] Elhamifar E, Vidal R. Sparse subspace clustering: algorithm, theory, and applications [J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 2013, 35 (11): 2765-2781.

[8] Lu C Y, Min H, Zhao Z Q, et al. Robust and efficient subspace segmentation via least squares regression [C]// Proc of the 12th European Conference on Computer Vision. 2012: 347-360.

[9] Liu G C, Lin Z C, Yu Y. Robust subspace segmentation by low-rank representation [C]// Proc of the 27th International Conference on Machine Leaning. 2010: 663-670.

[10] Xu J, Xu K, Chen K, et al. Reweighted sparse subspace clustering [J]. Computer Vision and Image Understanding, 2015 138: 25-37.

[11] 张涛, 唐振民, 吕建勇. 一种基于低秩表示的子空间聚类改进算法 [J]. 电子与信息学报, 2016, 38 (11): 2811-2818.

[12] Baadel S, Thabtah F, LU J. Overlapping clustering: a review [C]// Proc of IEEE Conference on SAI Computing. 2016: 233-237.

[13] Banerjee A, Krumpelman C, Ghosh J, et al. Model-based overlapping clustering [C]// Proc of the 11th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2005: 532-537.

[14] Fu Q, Banerjee A. Bayesian overlapping subspace clustering [C]// Proc of IEEE International Conference on Data Mining. 2009: 776-781.

[15] Candes E J, Wakin M B, Boyd S P. Enhancing sparsity by reweighted l1 Minimization [J]. Journal of Fourier Analysis and Applications, 2008, 14 (5): 877-905.

[16] Panagakis Y, Kotropoulos C. Elastic net clustering applied to pop//rock music structure analysis [J]. Pattern Recognition Letters, 2014, 38 (3): 46-51.

[17] Shi J B, Malik J. Normalized cuts and image segmentation [J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 2000, 22 (8): 888-905.

[18] Fu Q, Multiplication mixture models for overlapping clustering [C]// Proc of IEEE International Conference on Data Mining. 2008: 791-796.

[19] 刘展杰, 陈晓云. 局部子空间聚类 [J]. 自动化学报, 2016, 42 (8): 1238-1245.

[20] Cai D, He X F, Han J W. Non-negative matrix factorization on manifold [C]// Proc of the 8th IEEE International Conference on Data Mining. 2008: 63-72.

[21] 邓赵红, 张丹丹, 蒋亦樟. 基于划分自适应融合的多视角模糊聚类算法 [J]. 控制与决策, 2016, 31 (4): 593-600.

[22] Aggarwal C C, Procopiuc C M, Wolf J L, et al. Fast algorithms for projected clustering [C]// Proc of ACM SIGKDD International Conference on Management of Data. New York: ACM Press, 1999: 61-72.

Appendix: Calculation of Optimal Extreme Value θ_l^*

Using the second part of the log-likelihood function from Equation (26) to estimate parameter θ_l:

$$\theta_l^* = \arg\max_{\theta_l} \sum_{i=1}^{n} \log p(\mathbf{y}_i|\mathbf{b}_i, \theta_l)$$

Substituting Equation (18):

$$\theta_l^* = \arg\max_{\theta_l} \sum_{i=1}^{n} [b_{il} \log p(\mathbf{y}_i|\theta_l) + (1 - b_{il}) \log p(\mathbf{y}_i|\theta_l)]$$

Taking the second derivative with respect to θ_l:

$$\nabla^2_{\theta_l} f(\theta_l) = -\sum_{i=1}^{n} b_{il} \nabla^2_{\theta_l} \varphi(\theta_l)$$

Since $\nabla^2_{\theta_l} \varphi(\theta_l)$ is the cumulant function (always positive), $f(\theta_l)$ is convex. Setting the first derivative to zero yields the optimal extreme value:

$$\theta_l^* = \frac{\sum_{i=1}^{n} b_{il} \mathbf{T}(\mathbf{y}_i)}{\sum_{i=1}^{n} b_{il}}$$

Note: Figure translations are in progress. See original paper for figures.