IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received May 13, 2021, accepted June 3, 2021, date of publication June 8, 2021, date of current version June 16, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3087575

Active Block Diagonal Subspace Clustering

ZIQI XIE AND LIHONG WANG
School of Computer and Control Engineering, Yantai University, Yantai 264005, China

Corresponding author: Lihong Wang (wanglh@ytu.edu.cn)

This work was supported by the National Natural Science Foundation of China under Grant 62072391 and Grant 61773331.

ABSTRACT Subspace clustering aims to find clusters in the low-dimensional subspaces for high-dimensional data. Subspace clustering with Block Diagonal Representation (BDR) maintains the number of connected components of the graph by Laplacian rank constraint, and the learned affinity matrix shows a block diagonal structure, which will achieve a good segmentation for the dataset by spectral clustering. However, the subspaces of real data may overlap and the learned affinity matrix may be imprecise. In this work, we propose an Active learning framework for BDR(ABDR) to acquire and incorporate prior knowledge to improve the subspace clustering performance. An active selection strategy is designed to acquire labels of the informative data points from both the skeleton of clusters and the boundaries of clusters, and then the labeled data are converted into pairwise constraints, which are incorporated into BDR. The optimization of the new objective function is given and the convergence of ABDR is discussed. Experimental results on three images datasets(MNIST, ORL and COIL-20) and one UCI dataset(ISOLET) demonstrate the effectiveness of ABDR on complex clustering tasks and show that ABDR is superior to multiple state-of-the-art active clustering and learning techniques.

INDEX TERMS Active learning, subspace clustering, block diagonal representation, pairwise constraints, self-expressiveness, convergence.

I. INTRODUCTION

As a preliminary task in the field of data mining and machine learning, data clustering detects the underlying structures of a given dataset and divides it into a number of clusters. However, traditional clustering technologies often perform poorly in real applications, especially for high-dimensional data, such as text data clustering [1], image segmentation [2] and face recognition [3]. One of the reasons is that high-dimensional data often lie in low-dimensional subspaces instead of being uniformly distributed across the ambient space [4]. Subspace clustering or subspace segmentation aims to segment data samples into different clusters, where each cluster is related to a subspace [5]. In the past decade, lots of researches focus on constructing a block-diagonal affinity matrix because of the strong self-expressiveness property as well as robustness to the outliers. For example, Sparse Subspace Clustering (SSC) [4] presents a sparse representation to select a few points from the same subspace. SSC can handle noise and outliers by incorporating the model of the data into the sparse optimization program. Lu *et al.* [6] showed that the l_1-minimization in SSC makes not only the between-cluster connections sparse, but also the inner-cluster connections sparse, thus the clustering results obtained by spectral clustering may not be correct. Low-Rank Representation (LRR) [7], [8] enforces low-rank constraint on the affinity matrix to achieve block-diagonal property and encouraging performance. However, low-rank constraint cannot always generate block-diagonal representation matrix so that it might fail to uncover the intrinsic multiple subspace structure within the data [5], [6]. Recently, Block Diagonal Representation (BDR) for subspace clustering algorithm has been widely concerned due to its good clustering performance [6], [9]. The BDR algorithm explores the subspace representation and adds Laplacian rank constraints to the objective function to maintain the number of connected components of the graph. Ideally, the affinity matrix has a k-block diagonal structure, each of which is a self-expressive coefficient sub-matrix between data points within the same subspace. In this case, spectral clustering on the affinity matrix will segment the data into reasonable clusters. BDR, as well as most of subspace clustering algorithms, assumes that subspaces are independent of each other and the data are noise free [10]. However, for complex clustering tasks, the subspaces of real data may

The associate editor coordinating the review of this manuscript and approving it for publication was Qilian Liang.

be highly overlapped and corrupted by noise and (or) outliers, and it is difficult to get a subspace-preserving affinity matrix [10]. In this case, the clustering results of BDR are inconsistent with the expected clusters, thus cause high clustering errors [11].

In order to improve the subspace clustering performance, semi-supervised methods may be used to guide the clustering process by allowing the user to provide external semantic knowledge, generally in the form of data labels [12]–[14] or pairwise constraints on elements in the data [15]–[17]. The pairwise constraints include must-links (two samples are known to be in the same cluster) and cannot-links (two samples are known to be in different clusters) [18], [19]. These efforts on semi-supervised clustering have shown that, when the constraints are selected well, incorporating pairwise constraints can significantly improve the clustering results.

In this paper, we propose an Active learning framework for BDR (ABDR) to challenge complex datasets by incorporating additional information into BDR. ABDR has three advantages over previous active subspace clustering and semi-supervised subspace clustering.

Firstly, the baseline is promising. As a novel algorithm, BDR has outperformed most other subspace clustering algorithms, such as SSC and LRR, which motivates us to use BDR as a baseline for semi-supervised learning.

Secondly, the active selection strategy is novel and efficient. We select informative data points instead of pairwise constraints and use their labels to provide more information for constrained clustering. Specifically, we select data points from both the skeleton of clusters and the boundaries of clusters. The constraints that provide information about the skeleton of clusters can be very useful since they allow clustering algorithms to better discover complex-shaped clusters [20]. We define the skeleton of clusters as certain points and the boundaries of clusters as uncertain points via l_1-norm of each cluster, and provide a chance for each cluster to query its own certain and uncertain points. The selected informative points are locally-based, which take a closer look at the structure of the dataset. Experiments show that our strategy outperforms the compared state-of-the-art active strategies on test datasets with different subspace structures.

Finally, the proposed constrained subspace clustering incorporates the semi-supervised information into the clustering process in a new way. First of all, we convert the known data labels into pairwise constraints, and then use the cannot-links to zero the corresponding entries in the representation matrix, leaving the entries of must-links to be determined by ABDR. ABDR optimizes simultaneously the constrained representation matrix and the objective function that concerns the reconstructed error and the Laplacian rank. Our semi-supervised learning method can be used in other constrained subspace clustering algorithms, such as constrained SSC [21], to incorporate the cannot-links by zeroing the entries of the sparse coefficient matrix to break the connections between points from different clusters, other than directly adjust the affinity matrix for spectral clustering. The effectiveness of our strategy is verified by comparing it against other strategies on complex clustering tasks.

The main contributions of this paper are as follows:

- We propose an Active learning framework for BDR, ABDR, which combines an active selection strategy and a constrained BDR clustering algorithm. The proposed selection strategy acquires valuable data labels in batch and converts them into pairwise constraints for constrained clustering.
- We propose a constrained BDR algorithm, whose objective function is constrained by cannot-link constraints. Must-links are used to enhance the affinity matrix for final spectral clustering.
- We compare ABDR against baselines and state-of-the art active clustering techniques on three image datasets (MNIST [22], ORL [23] and COIL-20 [24]), and one UCI machine learning dataset ISOLET [25]. Our results (see Section IV) show that given the same number of labels queried, our method performs significantly better than existing state-of-the-art techniques.

The rest of this paper is organized as follows: The block diagonal representation-based subspace clustering algorithm is reviewed in Section II. ABDR is proposed in Section III. The proposed algorithm is evaluated experimentally in Section IV and discussed in Section V. Conclusions are given in Section VI.

II. RELATED WORK

A. BDR

BDR adds Laplacian rank constraint, named the k-block diagonal regularizer, to the objective function to maintain the number of connected components of the graph for a block diagonal affinity matrix [6], [9].

$$\min_{Z,B} \frac{1}{2} \| X - XZ\|_F^2 + \frac{\beta}{2} \|Z - B\|_F^2 + \gamma \|B\|_{k} ,$$
$$s.t. \; diag\,(B) = 0, \quad B \geq 0, \; B = B^{\mathsf{T}}, \tag{1}$$

where, Z is the representation matrix, $X = [x_1, x_2, \ldots, x_n]$ is the dataset with each column as an object, and k is the cluster number.

For any matrix B, the k-block diagonal regularizer is defined as the sum of the first k smallest eigenvalues of L_B, i.e.,

$$\|B\|_k = \sum_{i=n-k+1}^{n} \lambda_i \left(L_B\right), \tag{2}$$

where, $L_B = Diag(B1) - B$ is the Laplacian matrix of B, and $\lambda_i, i = 1, \ldots, n$, are the eigenvalues of L_B in descending order. The multiplicity k of eigenvalue 0 of L_B equals the number of connected components of the spectral graph of B [26]. When $\|B\|_k = 0$, the matrix B shows a k-block diagonal structure. In view of subspaces, when the subspaces of the data are independent of each other and the data are noise free, the affinity matrix B obtained by BDR has the k-block diagonal structure, i.e.,

$$B = \begin{bmatrix} B_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & B_k \end{bmatrix}.$$

BDR-Z [6] is the BDR algorithm, in which spectral clustering is conducted on the affinity matrix W, $W = \left(|Z| + \left|Z^{\mathsf{T}}\right|\right)/2$.

To enhance the block diagonal structure, some studies focus on global constraints by adding regularization terms to the objective functions to optimize the representation matrix as well as the regularization terms. Li *et al.* [27] proposed Structured Sparse Subspace Clustering (S^{3}C) for learning both the affinity and the segmentation matrix. The regularization term in the objective function is the structure induced by a norm that depends on the segmentation matrix. Both the structured sparse representation and the segmentation can be found simultaneously by optimizing the objective function via a combination of ADMM (an Alternating Direction Method of Multipliers) with spectral clustering [27]. Similarly, Liu *et al.* [28] proposed SBDR (Structured Block Diagonal Representation) subspace clustering by adding the same structure as the regularization term to the objective function, and using the structure matrix obtained by the spectral clustering to facilitate a better initialization for the representation matrix. Based on SBDR, Xie and Wang [29] proposed a constrained SBDR subspace clustering algorithm. The objective function of SBDR is re-optimized under an additional semi-supervised constraint, in which the representative coefficients between both must-linked and cannot-linked nodes are set to 0 to clean the noise in the representation matrix. Wang *et al.* [11] proposed a Robust Block Diagonal Representation learning (RBDR) for subspace clustering. They used a penalty matrix to adaptively weight the reconstruction error and directly constrained the affinity matrix in the objective function. Thus, RBDR can handle noise without prior knowledge. Xu *et al.* [30] proposed a latent BDR (LBDR) model to perform the subspace clustering on a nonlinear structure, which jointly learns an autoencoder and a BDR matrix. The autoencoder learns features from the nonlinear samples and then uses the learned features as a new dictionary for a linear model with block-diagonal regularization to ensure good performance for spectral clustering. Additionally, Guo *et al.* [31] proposed a block diagonal representation for multi-view subspace clustering (MSCBDR). This model concerns a block diagonal regularizer with the complementarity of multi-view information, which is different from our main concern. Generally, the clustering performance is corrupted by a small number of ''hard'' data points, whose local structures in the affinity matrix cannot be correctly fixed by global constraints. Therefore, some researchers explore the applications of constrained subspace clustering.

B. CONSTRAINED SUBSPACE CLUSTERING

Constrained clustering is a semi-supervised learning method which focuses on enhancing the quality of the partition by utilizing pairwise constraints [32]. To improve the clustering performance, prior knowledge or constraints are incorporated into the clustering process to optimize the local structure of the affinity matrix. Compared with the global regularization, the local constraints can improve the clustering performance pointedly. Pairwise constraints, as local side information, are popularly used in constrained clustering algorithms. As a step of BDR, spectral clustering is employed to obtain the segmentation of the dataset. Some efforts on constrained spectral clustering incorporate must-links and cannot-links provided by the user or obtained by the active learning into the affinity matrix before spectral clustering [32]–[38]. Specifically, the affinity matrix entries of the must-linked pairs are set to 1 while the entries of cannot-linked ones are set to -1 [38] or 0 [39]. However, experiments show that constrained clustering with random constraint selection may degrade the clustering performance of the basic $k-$means algorithm [40], [41], and the performance of spectral clustering depends heavily on the quality and quantity of constraints [20], [39].

As for constrained subspace clustering, more efforts pay attention to the constrained (semi-supervised) LRR [17], [42], [43] than constrained SSC [21]. Fang *et al.* [42] proposed a robust semi-supervised subspace clustering method based on non-negative LRR (NNLRR). The supervision information, such as the label information, is explicitly incorporated in an optimization problem to guide the affinity matrix construction and subspace clustering. Wang *et al.* [17] proposed a constrained low-rank representation (CLRR) for robust semi-supervised subspace clustering based on a novel constraint matrix constructed. CLRR incorporates supervision information as hard constraints for enhancing the discriminating power of optimal low-rank representation of data. Fang *et al.* [43] proposed a symmetry constrained latent low rank representation with converted nuclear norm (SLLRRC) algorithm for robust subspace clustering. SLLRRC both enhances the sparsity of the coefficient matrix and guarantees weight consistency for each pair of data samples when seeking the low rank representation. Huang *et al.* [21] proposed a unified framework for hyperspectral image (HSI) clustering, which incorporates spatial information and label information in a SSC model for a more precise similarity matrix. The spatial information is included through a joint sparsity constraint on the coefficient matrix of each local region and the available label information is incorporated by zeroing the entries of the sparse coefficient matrix corresponding to the data points from different classes. To the best of our knowledge, our previous work [29] is the first one on constrained BDR. Specifically, this work incorporates pairwise constraints into the structured subspace clustering with block diagonal representation (SBDR). In this paper, we incorporate pairwise constraints into the subspace clustering with BDR, and explore the constrained BDR without the help of structure regularizer in the SBDR.

C. ACTIVE CONSTRAINT LEARNING

It is widely accepted that not all pairwise constraints are equally important. Good constraints can greatly improve clustering performance, while bad constraints will likely result in very little improvement, or even degrade clustering performance [16], [44]. Querying high quality constraints for clustering is a difficult task due to the lack of an appropriate criterion for measuring the quality of constraints. Abin [44] proposed a method to track the quality of candidate constraints in embedding spaces instead of the input data space, and the proposed method outperforms existing algorithms for constraint selection when used in conjunction with two well-known clustering methods.

Active learning, which aims at searching and utilizing supervised information to guide the clustering, is used for constraint selection. The efforts on active constraint learning can be mainly grouped into two categories: pairwise constraint learning and node label learning. In the spectral graph theory, each node in the graph represents a data point, thus, we use nodes and data points alternatively in the following sections.

(1) Pairwise constraint learning. Due to the difficulty of providing category labels for large multi-class problems, many works used pairwise constraints to improve the clustering performance. The work of Klein *et al.* [45] may be the first work to show the benefit of carefully chosen constraints for constrained clustering [20]. Farthest First Query Selection (FFQS) [46] was proposed for constraints collection based on the min-max method, which includes two steps: Explore and Consolidate. A set of cannot-link constraints is identified to form a skeleton of the clusters and the farthest data point from existing skeleton is chosen to query in Explore steps. Consolidate step randomly picks a point outside the skeleton and queries it against each point in the skeleton, until the user stops querying the constraints. Liu *et al.* [16] used entropy to measure uncertainty, and selected the most uncertain data point based on the intermediate clustering results, then exploited the pairwise constraints of the selected points to guide SSC (Sparse Subspace Clustering). References [33] and [35] combined cluster assignment vectors with the ground-truth labels and picked the pair of nodes with the maximal expected error reduction. In 2011, Biswas and Jacobs [34] optimized the active strategy by taking into account the size and distance of the cluster where the data pairs are located, avoiding the selection of extreme constraints. In 2014, Biswas and Jacobs [36] used Jaccard's coefficient (JCC) as a measure of similarity between two clustering results, and selected the pair of nodes which produced the maximal expected change in the clustering. Xie and Wang [29] proposed an active learning strategy for block diagonal subspace clustering, which iteratively selects a point inside each block as a representative point of the block and a point outside the block as a suspected missing one to form a pairwise constraint. Besides, Abin and Wu [20] proposed a method, we call it DQIC (Density-based approach for Querying Informative Constraints), to estimate density and impurity of data points on different adjacency distances and calculate centrality for each data point by applying a density tracking approach on the obtained densities. The obtained information is then used to select a set of high-quality constraints. In summary, the active strategies above maximally reduce the uncertainty of the whole dataset by querying on pairwise constraints.

(2) Node label learning. Xiong *et al.* [37] proposed local and global nonparametric structure models based on node rather than pair uncertainty. One of the reasons is that, if an uncertain pair contains two uncertain nodes, a constraint between these nodes will not extrapolate well beyond these two nodes. The other one is that pair selection has an inherently higher complexity and limited scalability due to the presence of n^2 constraints for every n nodes [37], [38]. Aiming at reducing the misclassification of boundary points in dichotomy, Xu *et al.* [47] defined nodes far away from the cluster boundary as definite points and the ones near the cluster boundary as uncertain points, respectively, which are obtained by analyzing spectral eigenvectors. Xiong *et al.* [38] proposed an uncertainty reduction model URASC (Uncertainty Reducing Active Spectral Clustering) for informative sample selection, and estimated the uncertainty reduction potential of each sample in the dataset. The pairwise with respect to only the best candidate sample are queried and the constraints are used to modify the similarity matrix and then the spectral clustering is employed to give the segmentation of the dataset. Liu *et al.* [48] defined a constraint strategy called Partition Level Side information to directly obtain the ground-truth labels of a small proportion of data, expecting to uncover the data distribution structure from these known data.

Following [37], we propose an active node label learning strategy for BDR. We pay attention to node label learning for another reason, i.e., the node labels contain more information than that of pairwise constraints, as shown in Figure 1. The queries on node labels usually deduce more constraints than the same number of pairwise queries.

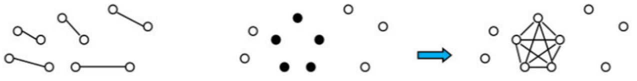

(a) 5 must-links. (b) 5 nodes with the same label (c) 10 must-links.

FIGURE 1. 5 queries on node labels and pairwise constraints.

III. THE PROPOSED ABDR

A. ALGORITHM FRAMEWORK

We propose an active constrained subspace clustering based on BDR to incorporate the acquired pairwise constraints into the clustering process. The real-world data usually contain noise and (or) outliers, and the data subspaces are overlapped. Due to the sensitivity of the clustering performance to the constraints, it is challenging to find valuable data nodes and use them efficiently (e.g., convert into pairwise constraints in this paper) to improve the clustering performance.

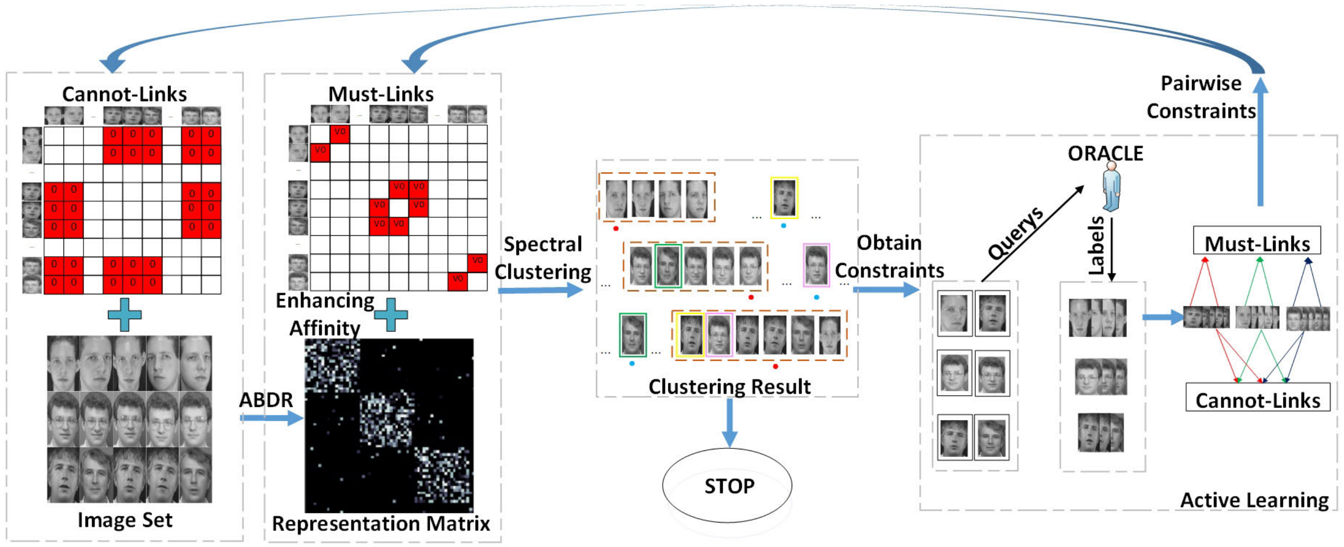


FIGURE 2. The framework of ABDR.

First of all, we run BDR-Z on the unlabeled data (e.g., Image set) to initialize the affinity matrix W, which is used to segment the data into k-clusters by spectral clustering. Based on the segmentation, we use the proposed active selection strategy to select some nodes from each cluster and query their labels, and then add these new labeled data to the set of known data, and convert them into pairwise constraints.

Next, as shown in Figure 2, we input the dataset and cannot-links into our ABDR, and enhance the learned affinity matrix W by must-links, and then employ the spectral clustering on W to generate the segmentation. Now, we can stop the program and output the segmentation as the clustering result. If we want to refine the clustering result further, we continue to select new nodes based on the segmentation and query their labels and then update pairwise constraints again. We will explain the proposed framework in the following sections.

B. THE OBJECTIVE FUNCTION OF ABDR

The objective function of ABDR is as follows:

$$\begin{aligned}&\min J(Z,B)=\frac{1}{2}\|X-XZ\|_F^2+\frac{\beta}{2}\|Z-B\|_F^2+\gamma\|B\|_k\,,\\&s.t.\ diag\,(B)=0,\quad B\geq 0,\ B=B^{\mathsf{T}},\ Z_C=0,\end{aligned}\tag{3}$$

where, C is the set of cannot-links, and $Z_C=0$ requires the entries of cannot-linked nodes in the representation matrix Z to be 0, i.e.,

$$Z_{ij}=0,\ (x_i,x_j)\in C.$$

The constraint $Z_C=0$ is designed to reduce the noise in the representation matrix Z, since the representation coefficient of x_i and x_j must be zero by the self-expressiveness property of subspaces if x_i and x_j are cannot-linked [10]. Meanwhile, the representation coefficients of must-linked nodes are calculated by the optimization process. We use must-links to enhance the affinity matrix W as follows, and then W is input into spectral clustering to obtain the clustering result.

$$\begin{aligned}A&=\left(|Z|+\left|Z^{\mathsf{T}}\right|\right)/2,\\W&=\begin{cases}\max\left(v_0,A_{ij}\right) & \left(x_i,x_j\right)\in M,\\A_{ij} & otherwize,\end{cases}\end{aligned}\tag{4}$$

where, v_0 is a constant, and M is the set of must-links.

Pairwise constraints are usually used to adjust the affinity matrix for spectral clustering, however, we use the cannot-links to clean the corresponding entries in the representation matrix Z in the iterations of ABDR, leaving the entries of must-links in Z to be determined by ABDR, and then employ the must-links to enhance the affinity matrix W for spectral clustering. Since the matrix Z of the dataset X is a representation of X, the clearer and more accurate the block diagonal structure of Z is, the easier the spectral clustering finds a correct segmentation matrix for X. On one hand, we zero the representation coefficients of cannot-linked points to prevent them to be clustered in the same cluster; on the other hand, the connections of must-linked points are enhanced in the affinity matrix to make it easy to cluster them together by the spectral clustering.

C. OPTIMIZATION OF THE OBJECTIVE FUNCTION

We rewrite $\|B\|_k$ as a convex optimization problem [6],

$$\begin{aligned}\|B\|_k&=\min_V\ \langle L_B,V\rangle\\&\qquad s.t.\ 0\preceq V\preceq I,\quad tr\,(V)=k.\end{aligned}$$

Then, the objective function can be rewritten as

$$\begin{aligned}&\min_{Z,B}\frac{1}{2}\|X-XZ\|_F^2+\frac{\beta}{2}\|Z-B\|_F^2\\&\qquad+\gamma\,\langle Diag\,(B\mathbf{1}-B,V)\rangle\,,\end{aligned}$$

$$\begin{aligned} s.t.\ diag\left(B\right)=0,\quad & B\geq 0,\ B=B^{\mathsf{T}},\ Z_C=0, \\ 0\preceq V\preceq I,\quad & tr\left(V\right)=k. \end{aligned} \tag{5}$$

ADMM is adopted to solve V, Z and B as follows:

Step 1: Update V with fixed Z and B. V is updated as

$$\begin{aligned} V^{t+1} &= \arg\min_{V} \left\langle Diag\left(B1\right), V\right\rangle \\ & s.t.\ 0\preceq V\preceq I,\quad tr\left(V\right)=k. \end{aligned}$$

The solution of V is given as follows [6]:

$$V^{t+1}=UU^{\mathsf{T}}, \tag{6}$$

where, $U\in R^{n*k}$ consists of the k eigenvectors associated with the k smallest eigenvalues of L_B.

Step 2: Update B with fixed Z and V [6].

$$B^{t+1}=\left[\left(\hat{A}+\hat{A}^{\mathsf{T}}\right)/2\right]_{+}, \tag{7}$$

where, $\left[A\right]_{+}=\max\left(A,0\right), A=Z-\frac{\gamma}{\beta}\left(diag\left(V\right)1^{\mathsf{T}}-V\right)$, $\hat{A}=A-Diag\left(diag\left(A\right)\right).$

Step 3: Update Z with fixed B and V.

We partition Z into Z_C and $Z_{\overline{C}}$, $Z=Z_C+Z_{\overline{C}}$.

$$\begin{aligned} Z_C(i,j) &= \begin{cases} Z_{i,j} & \left(x_i,x_j\right)\in C, \\ 0 & otherwize. \end{cases} \\ Z_{\overline{C}}\left(i,j\right) &= \begin{cases} 0 & \left(x_i,x_j\right)\in C, \\ Z_{i,j} & otherwize. \end{cases} \end{aligned} \tag{8}$$

According to the Augmented Lagrangian Method (ALM) [49], Z can be updated by solving the following problem:

$$\begin{aligned} f\left(Z_C,Z_{\overline{C}}\right) = \frac{1}{2}\left\|X-XZ\right\|_F^2+\frac{\beta}{2}\left\|Z-B\right\|_F^2 \\ +\frac{\mu}{2}\left\|Z_C+\frac{1}{\mu}\Lambda\right\|_F^2. \end{aligned} \tag{9}$$

Following [29], [49], (9) can be rewritten as

$$\begin{aligned} & f(Z_C,Z_{\overline{C}}) \\ &= \frac{1}{2}tr((\mathrm{X\text{-}XZ_C\text{-}xZ_{\overline{C}}})^{\mathsf{T}}(\mathrm{X\text{-}XZ_C\text{-}xZ_{\overline{C}}})) \\ &\quad +\frac{\beta}{2}tr((Z_C-B_C)^{\mathsf{T}}(Z_C-B_C)) \\ &\quad +\frac{\beta}{2}tr((Z_{\overline{C}}-B_{\overline{C}})^{\mathsf{T}}(Z_{\overline{C}}-B_{\overline{C}})) \\ &\quad +\frac{\mu}{2}tr((Z_C+\frac{1}{\mu}\Lambda)^{\mathsf{T}}(Z_C+\frac{1}{\mu}\Lambda)). \end{aligned}$$

Take the derivative of Z_C and set it to 0, we derive

$$Z_C=(XX^{\mathsf{T}}+(\beta+\mu)I)^{-1}(-X^{\mathsf{T}}XZ_{\overline{C}}+L), \tag{10}$$

where, $L=XX^{\mathsf{T}}+\beta B_C-\Lambda$, and I is the identity matrix.

Similarly, we have

$$Z_{\overline{C}}=(XX^{\mathsf{T}}+\beta I)^{-1}(-X^{\mathsf{T}}XZ_C+D), \tag{11}$$

where, $D=\beta B_{\overline{C}}+XX^{\mathsf{T}}$.

Substitute (11) into (10), then we derive

$$Z_C=Y^{-1}R, \tag{12}$$

where, $Y=\beta X^{\mathsf{T}}X(X^{\mathsf{T}}X+\beta I)^{-1}+(\beta+\mu)I$ and $R=L-X^{\mathsf{T}}X(X^{\mathsf{T}}X+\beta I)^{-1}D$.

Consequently, we get $Z_{\overline{C}}$ by substituting (12) into (11), then $Z=Z_C+Z_{\overline{C}}$.

According to the ALM, Z can be obtained by iteratively updating μ and Λ. In this paper, we set the number of iterations (MAXno) to 100, $\rho=1.8$ and $\mu=1.5$ for ALM. See **Algorithm 1** for details of updating Z, and **Algorithm 2** for ABDR.

The convergence condition of ABDR is

$$max\{\left\|Z^{t+1}-Z^t\right\|_\infty, \left\|B^{t+1}-B^t\right\|_\infty\}\leq\varepsilon.$$

As suggested, $\varepsilon=10^{-3}$ [6]. We set the number of iterations $S=1$ for ABDR in the following experiments.

D. THE PROPOSED ACTIVE SELECTION STRATEGY

Most of the methods in the field of constraint learning select constraints from the boundaries of clusters, however,

Algorithm 1 Update Z

```
Input: $X \in R^{d*n}$, $C$, $\beta$.
Output: $Z \in R^{n*n}$.
Initialization: $1 < \rho < 2$, $\mu > 0$, $\Lambda = 0_{n*n}$.
for $t = 1$:MAXno do
    Update $Z_C$ by (12);
    Update $Z_{\overline{C}}$ by (11);
    $\Lambda \leftarrow \Lambda + \mu Z_C$;
    $\mu \leftarrow \mu\rho$;
end for
$Z = Z_C + Z_{\overline{C}}$.
return $Z$.
```

Algorithm 2 ABDR

```
Input: $X \in R^{d*n}$, $\gamma > 0$, $\beta > 0$, $\varepsilon > 0$.
Output: the segmentation of $X$.
Initialization: $t = 0$, $V^t = 0_{n*n}$, $B^t = 0_{n*n}$, $Z^t = 0_{n*n}$, $M = \varnothing$, $C = \varnothing$.
Run BDR-Z to initialize $W$ and the segmentation;
for $s = 1 : S$ do
    Update $M$ and $C$ by active selection;
    while not converged do
        Update $V^{t+1}$ by (6);
        Update $B^{t+1}$ by (7);
        Update $Z^{t+1}$ by Algorithm 1;
        $t = t + 1$;
    end while
    Enhance $W$ by (4);
    Input enhanced $W$ to spectral clustering for segmentation.
end for
return the segmentation.
```

the constraints that provide information about the skeleton of clusters can also be very useful since they allow clustering algorithms to better discover complex-shaped clusters [20]. In this paper, we select data points from both the skeleton of clusters and the boundaries of clusters. The points from the skeleton of clusters are concerned as certain points, while the ones from the boundaries of clusters are uncertain.

Firstly, we sort the rows and columns of the affinity matrix W according to the data segmentation. Ideally, the sorted W has a block diagonal structure and each block is a self-expressive coefficient matrix of data points from the same subspace.

Formally, assume that W_1 is a block on the diagonal of W, and the data subset in the corresponding block(cluster) is X_1, then we define the *Representative Strength* (RS) of $x_i \in X_1$ as the l_1 -norm of the corresponding column vector in W_1,

$$RS(x_i) = \sum_{x_j \in X_1} |w_{ji}|.$$

Secondly, we calculate the RS values for all points in X_1 and sort them in ascending order. As shown in Figure 3, let $\langle x_1, x_2, \ldots, x_{n1} \rangle$ be the sorted queue of X_1.

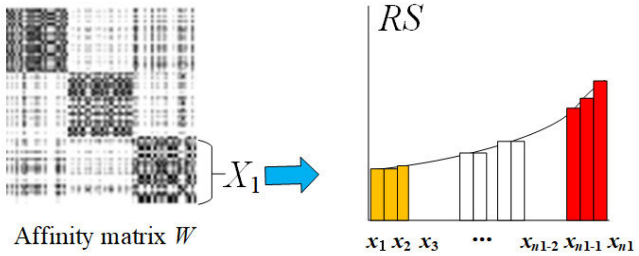

Affinity matrix W
X_1
RS
x_1 x_2 x_3 ··· $x_{n1\text{-}2}$ $x_{n1\text{-}1}$ x_{n1}

FIGURE 3. The proposed selection strategy.

Specifically,

$$\begin{aligned} x_1 &= \arg\min_{x_i \in X_1} RS(x_i), \\ x_2 &= \arg\min_{x_i \in X_1, x_i \neq x_1} RS(x_i), \\ x_{n_1} &= \arg\max_{x_i \in X_1} RS(x_i), \\ x_{n_1-1} &= \arg\max_{x_i \in X_1, x_i \neq x_{n-1}} RS(x_i). \end{aligned} \tag{13}$$

The point with the highest RS value, i.e., x_{n_1}, is most closely connected with the points in cluster X_1, and can be regarded as the certain point of cluster X_1. On the other hand, the point x_1 that loosely connects with the points in cluster X_1 is the uncertain one.

Thirdly, following the concept of p-partition level side information [48], we select a small proportion $p \in (0, 1)$ of unlabeled data in X. We pick up required number of data points from two ends of the sorted l_1-queue of each block. For example, we pick x_{n_1} and x_{n_1-1} with the first two highest RS values as the skeleton of clusters and the equal number of points with the lowest RS, i.e., x_1 and x_2, as boundaries of clusters, and then query their labels.

Next, we convert the labeled data to pairwise constraints in the trivial way, i.e., x_i and x_j are must-linked ($(x_i, x_j) \in M$) if they have the same label, and cannot-linked ($(x_i, x_j) \in C$) otherwise.

$$\begin{aligned} (x_i, x_j) \in M &\quad \textit{if } label(x_i) = label(x_j), \\ (x_i, x_j) \in C &\quad \textit{if } label(x_i) \neq label(x_j). \end{aligned}$$

Finally, the pairwise constraints are incorporated into ABDR to get the final clustering result. Generally, ABDR with the updated constraints runs iteratively. For simplification, we select and query the unlabeled data in batch, and the program stops after only one iteration in the following experiments. The overall flow of ABDR is summarized in Figure 4.

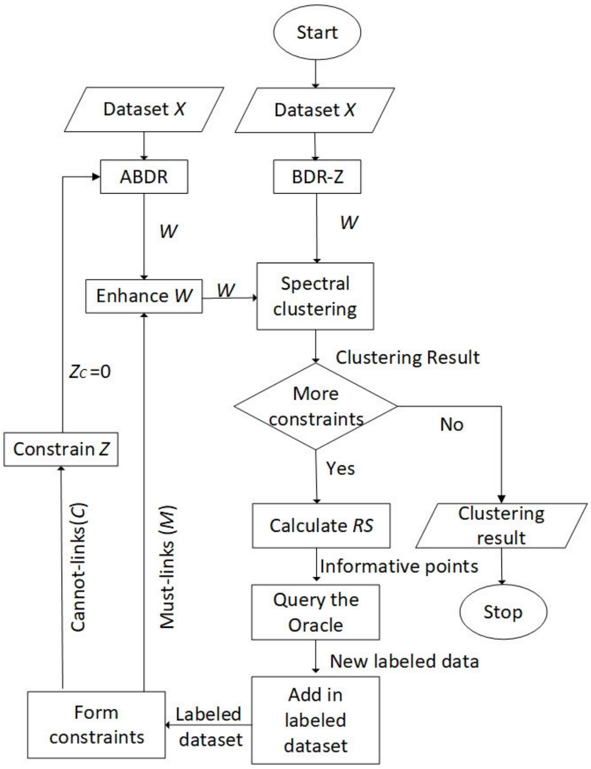


FIGURE 4. The flow chart of ABDR.

E. OVERVIEW OF THE PROPOSED METHOD

We give the overview of the proposed ABDR from seven perspectives of interactive clustering [50].

1) ON WHAT LEVEL IS THE INTERACTION HAPPENING

ABDR presents preliminary BDR clustering results to the user and then gives them the freedom to guide the subsequent interactive process. ABDR asks the users for true labels of the selected informative data using ideas from active learning, and then uses these labels to generate must-link/cannot-link constraints, which works as a constraint. ABDR does not query must-link/cannot-link constraints directly.

2) WHICH INTERACTIVE OPERATIONS ARE INVOLVED

It is the ABDR program that makes corrections of the preliminary BDR clusters by adding constraints. Specifically, active learning finds some informative data and queries the user for the labels, and then uses these labels to generate must-link/cannot-link constraints to adjust the representation matrix Z. The iteration of ABDR by adding these constraints in its objective function will correct some error clusters and lead to a better clustering result.

3) HOW USER FEEDBACK IS INCORPORATED

The interactive clustering process continuously communicates information to users and takes feedback from them, and the interactive process requires a mechanism to incorporate the user's feedback [50]. In ABDR, the user's feedback consists of answering the query with labels of informative data that are suggested by active learning strategy. ABDR is a constrained BDR in which user's feedback constrains the clustering by adjusting the representation matrix Z.

4) HOW INTERACTIVE CLUSTERING IS EVALUATED

We use both accuracy and normalized mutual information (NMI) as evaluation metrics.

5) WHICH DATA

We evaluate the proposed ABDR on four real datasets, i.e., MNIST, ORL, COIL-20 and ISOLET.

6) WHICH CLUSTERING METHODS HAVE BEEN USED

ABDR is a subspace clustering algorithm focused on clustering high-dimensional data, simultaneously, it is a constrained method as mentioned above.

7) WHAT OUTLINED CHALLENGES THERE ARE

a: TECHNICAL IMPROVEMENTS

ABDR needs to improve computational efficiency on large datasets. This challenge is inherited from BDR and the time cost for updating the representation matrix Z decreases the computational efficiency further. As a step of ABDR, spectral clustering is of crucial importance to the segmentation matrix, however, the spectral clustering is time-consuming [51] and unstable [52] for complex clustering tasks. We observe that when the cluster number is larger than 10, the spectral clustering may give a bad clustering result even the affinity matrix is good. Efforts on scalability and robustness of spectral clustering for large-scale datasets with limited resources [53] will be helpful for the improvement of ABDR.

b: METHODOLOGY DEVELOPMENT

The proposed ABDR needs to compare against other active constrained clustering algorithms, with the same active learning strategy and different clustering algorithms, or different active learning strategies and the same constrained BDR. We compare ABDR against four active learning strategies with the same constrained BDR, and compare the proposed active learning strategy against URASC [38] with the same spectral clustering, see the experiments for details.

IV. EXPERIMENTS

A. DATASETS

Three image datasets MNIST [22], ORL [23], COIL-20 [24] and one UCI dataset ISOLET [25] are selected to evaluate the proposed algorithm. The sample images from the image datasets are shown in Figure 5, and the details of 12 test datasets are summarized in Table 1.

TABLE 1. Details of 12 test datasets.

Dataset	Dimension	Size	Classes	Source
Dig6	784	600	6	MNIST
Dig8		800	8	
Dig10		1000	10	
ORL20	1024	200	20	ORL
ORL30		300	30	
ORL40		400	40	
COIL5	1024	360	5	COIL-20
COIL10		720	10	
COIL15		1080	15	
ISOLET6	617	360	6	ISOLET
ISOLET12		720	12	
ISOLET18		1080	18	

1) MNIST

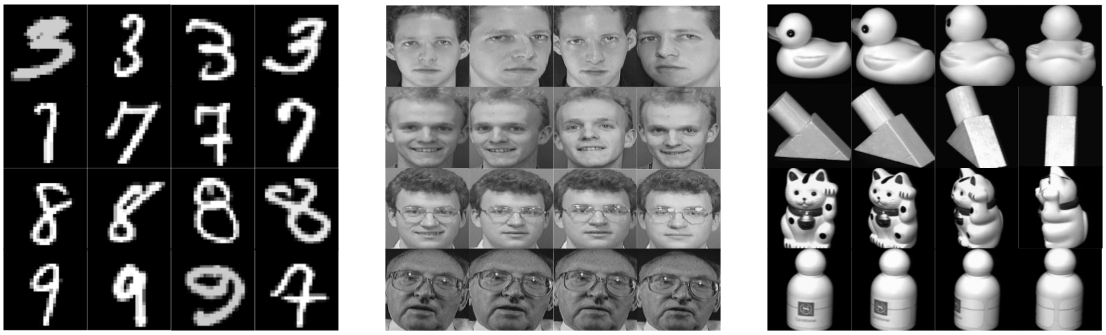
MNIST contains grey handwritten images of 10 digits from 0 to 9 and each image is of size 28*28. We randomly select 6, 8, and 10 digits with 100 images for each digit to form the test datasets Dig6, Dig8 and Dig10, respectively. Each original image is vectorized as a vector of length 784, and then the vector is normalized to have a unit length as suggested [6]. In the following experiments, each test dataset has 20 versions to be used in 20 runs, which is the same with another three datasets, and we use *MNIST related datasets* to refer to Dig6, Dig8 and Dig10 simultaneously.

2) ORL

ORL contains 400 face images of 40 subjects with 10 images each subject. To test the clustering performance of ABDR on ORL, three datasets ORL20, ORL30 and ORL40 are constructed. ORL20 and ORL30 consist of randomly selected 20 and 30 subjects with 10 images each subject, respectively, and ORL40 contains 40 subjects and 400 images. The face images are resized into 32*32, then vectorized as a vector and normalized to have a unit length.

3) COIL-20

COIL-20 is composed of 20 objects with 72 images per object. For each object, a camera takes an image by every 5 degrees when rotated through 360 degrees. Each original image with 128*128 pixels is resized to 32*32 and then turned into a unit vector. COIL5, COIL10 and COIL15 are generated, each of which contains all the images of randomly selected 5, 10, and 15 objects, respectively.



(a) MNIST.

(b) ORL.

(c) COIL-20.

FIGURE 5. Sample images from three image datasets.

4) ISOLET

ISOLET dataset consists of speech samples of 26 English letters, which were collected from 150 speakers who had spoken the name of the alphabet twice, and each of samples has 617 dimensions. We only use speech samples from 30 speakers, so we have 26 categories of samples, 60 of which are in each category. We pick $k \in \{6, 12, 18\}$ categories to randomly generate ISOLET6, ISOLET12 and ISOLET18, respectively.

It is challenging to conduct the subspace clustering on these complex high-dimensional datasets, especially for large multi-class problems.

B. EVALUATION METRIC

The accuracy (ACC) and Normalized Mutual Information (NMI) are used to evaluate the clustering results of ABDR and other algorithms.

Let $P = [p_1, p_2, \ldots, p_n], Q = [q_1, q_2, \ldots, q_n]$ represent the real labels and output labels of all data, respectively. ACC is defined as follows [6]:

$$ACC(P, Q) = \frac{1}{n} \sum_{i=1}^{n} \delta(p_i, map(q_i)),$$

where, p_i and q_i represent the real label and the output label of the ith data point, respectively, $\delta(x, y) = 1$ if $x = y$, and $\delta(x, y) = 0$ otherwise. $map()$ function matches the real labels with the output labels of the algorithm.

The higher ACC, the better clustering performance.

NMI is also a commonly used evaluation metric in clustering analysis [11],

$$NMI(P, Q) = \frac{I(P, Q)}{[S(P) + S(Q)]/2},$$

where, $I(P, Q)$ represents the interaction between P and Q, and S stands for entropy,

$$I(P, Q) = \sum_{i=1}^{R} \sum_{j=1}^{T} P(p_i \cap q_j) log \frac{P(p_i \cap q_j)}{P(p_i)P(q_j)},$$

$$S(P) = -\sum_{i=1}^{R} P(p_i) log P(p_i).$$

$P(c)$ is the probability that the data points belong to the cluster c. R and T represent the number of real labels and output labels of data, respectively.

The value range of NMI is [0, 1], and the higher the NMI, the better the performance.

C. BASELINES AND STATE-OF-THE-ART METHODS

To evaluate our ABDR framework and proposed active learning strategy, we test the following set of methods, including two baselines and multiple state-of-the-art active clustering and learning techniques.

- BDR-Z [6]: A baseline algorithm.
- SBDR [28]: Structured subspace clustering with block diagonal representation, as another baseline without active learning.
- Rand: Randomly sample given number of unlabelled data points.
- URASC [38]: An active spectral clustering algorithm based on uncertainty reduction.
- URASC + AL: Unlabeled data points are selected by our active learning method and the deduced constraints are fed to the spectral clustering algorithm of URASC.
- ABDR + UR: Unlabeled data points are selected by URASC and the deduced constraints are fed to ABDR.
- DQIC [20]: A density-based approach for querying informative constraints.
- LNSM [37]: An active learning framework which computes node uncertainty by local and nonparametric structure models.

D. EXPERIMENTAL SETUP

We take BDR-Z [6] and SBDR [28] as baselines to highlight the improvement of the constrained clustering. In addition, we evaluate the performance of the proposed active selection strategy as well as that of state-of-the-art selection strategies. To do so, we arm ABDR with another four selection strategies based on points, i.e., the uncertainty reduction model (ABDR + UR) [38], the random strategy (Rand), the local nonparametric structure model (LNSM) [37] and the approach based on density(DQIC) [20]. The five strategies

TABLE 2. W/T/L, average ranks and *p*-values for different metrics. Best results are high-lighted in bold, and the second-best ones are shown in italics.

	Metric	URASC	URASC+AL	ABDR+UR	Rand	LNSM	DQIC	ABDR	p-value
ACC	W/T/L	2/1/57	0/0/60	3/1/56	4/1/55	7/1/52	*13/1/46*	**28/1/31**	
	Avg. ranks	6.175	6.067	4.033	3.708	3.258	*2.650*	**2.000**	< 2.2e-16
NMI	W/T/L	0/0/60	0/0/60	1/0/59	8/0/52	*11/0/49*	4/0/56	**36/0/24**	
	Avg. ranks	6.233	5.750	4.833	3.500	*2.533*	3.300	**1.850**	< 2.2e-16

are evaluated on the test datasets in Section IV.A. To allow a fair comparison of the techniques, all evaluations use the same experimental protocol, i.e., the same random datasets, the same number of queried nodes as well as the same parameters in Table 4. The tuning of parameters β, γ, and δ are given in section V.B, and the sampling proportion p is selected for four datasets based on the idea of ensuring that each cluster has data points to be queried, and the queries are as few as possible. In addition, the constrained clustering algorithms based on ABDR (ABDR + UR, Rand, LNSM, DQIC and ABDR) are compared against URASC based ones (URASC and URASC + AL).

E. EXPERIMENTAL RESULTS

For each pair <dataset, algorithm> with a given sampling proportion p, the averaged ACCs and NMIs over 20 runs are shown in Tables 5-6 in the Appendix, respectively. The best value in each case is shown in bold. As expected, the best values of constrained clustering in most cases (59/60 for ACC, and 42/60 for NMI) are better than that of baselines.

1) W/T/L

W/T/L (Wins/Ties/Losses) records the number of times that the algorithm achieves the best value /equals the best value /is not the best value. From Table 2, we observe that, in view of W/T/L, ABDR performs significantly better than other methods on the test datasets. Specifically, ABDR wins on 28 and 36 test cases in terms of ACC and NMI, respectively, which are higher than that of other algorithms. Hence, ABDR is more advantageous in getting the best performance. In addition, the results of constrained clustering based on ABDR, i.e., ABDR + UR, Rand, LNSM, DQIC and ABDR, outperform the clustering results based on spectral clustering used by URASC, even if the points are selected by our active learning strategy in URASC + AL.

2) FRIEDMAN TEST AND NEMENYI MULTIPLE COMPARISON TEST

We use Friedman rank test [54] for the statistical comparison of these techniques over the 12 test datasets. Table 2 presents the average Friedman ranks summarized from Tables 5-6. Lower ranks are better, and the best performing algorithm is the one presenting the lowest average rank. ABDR is the algorithm with the lowest average ranks in terms of ACC and NMI (average rank = 2.000 and 1.850, respectively), and DQIC and LNSM are the second lowest ones (average rank = 2.650 and 2.533, respectively) in terms of ACC and NMI, respectively. In terms of average rank, we observe that our proposed ABDR is better than the others.

To compare the seven algorithms (i.e., URASC, URASC + AL, ABDR + UR, Rand, LNSM, DQIC and ABDR), we evaluate the following hypothesis H_0 using Friedman test.

Null Hypothesis H_0: The seven algorithms do not show any significant difference when used for clustering on the datasets.

We calculate p-value for each test, and the hypothesis is checked at $\alpha = 0.05$ significance level, as shown in Table 2.

In terms of both ACC and NMI, Friedman test results are significant at $\alpha = 0.05$. Thus, we reject Null hypothesis H_0. Namely, the seven algorithms perform significantly different from each other at the test datasets.

In addition, we conduct pairwise comparisons using Nemenyi multiple comparison test [55] for ACC and NMI. We calculate p-value for each test, and the hypothesis is checked at $\alpha = 0.05$ significance level, as shown in Table 3.

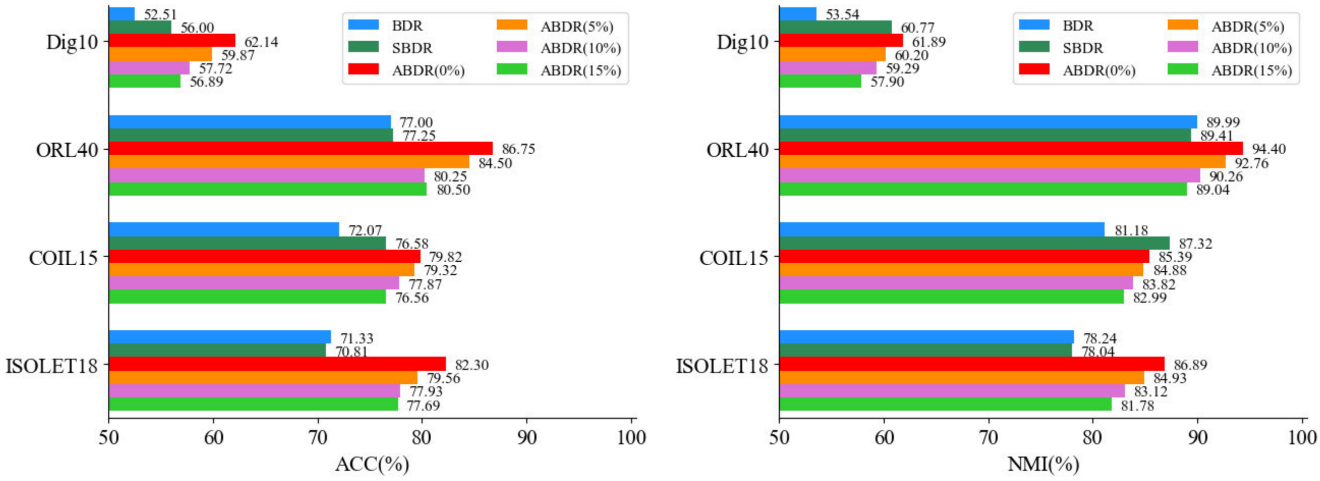
From Table 3, we observe that ABDR is significantly different from all compared algorithms except DQIC in terms of ACC, and significantly different from all compared algorithms except LNSM in terms of NMI. Moreover, four constrained clustering algorithms based on ABDR (Rand, LNSM, DQIC and ABDR) are all significantly different from URASC based ones (URASC and URASC + AL). In terms of W/T/L and statistical tests, we can draw a conclusion that the proposed framework of ABDR is better than that of URASC.

F. PERFORMANCE FOR THE NOISY LABEL CASE

In the proposed active learning strategy, we query the oracle for the ground truth labels of selected informative points. In practice, however, the human experts may provide wrong labels for the queried data points [36]. We test the robustness of ABDR by adding 5%, 10% and 15% noise to the queried labels, and compare the performance of ABDR against baselines. We employ Dig10 with 6% sampling proportion as a representative of MNIST, since more instances and larger test sampling proportion will capture more wrong labels than the other cases for MNIST. Similarly, ORL40 with 30%, COIL15 with 6% and ISOLET18 with 25% sampling proportion are chosen to test ABDR. The averaged performance over 20 runs on the four test datasets is shown in Figure 6, where ABDR(0%) denotes the ABDR without label noise. We observe that ABDR performs consistently in the presence of label noise over the datasets. As the fraction of mislabelled

TABLE 3. p-values of using Nemenyi multiple comparison test for ACC and NMI.

Metric		URASC	URASC+AL	ABDR+UR	Rand	LNSM	DQIC
ACC	URASC+AL	0.9981	-	-	-	-	-
	ABDR+UR	2.4e-07	5.3e-06	-	-	-	-
	Rand	1.4e-09	4.7e-08	0.9826	-	-	-
	LNSM	4.0e-13	2.3e-11	0.4372	0.9155	-	-
	DQIC	7.9e-14	5.9e-14	0.0082	0.1024	0.7190	-
	ABDR	7.0e-14	7.1e-14	5.3e-06	0.0003	0.0240	0.6509
NMI	URASC+AL	0.8844	-	-	-	-	-
	ABDR+UR	0.0071	0.2323	-	-	-	-
	Rand	8.8e-11	2.4e-07	0.0128	-	-	-
	LNSM	6.7e-14	9.2e-14	1.1e-07	0.1776	-	-
	DQIC	2.2e-12	1.1e-08	0.0019	0.9987	0.4511	-
	ABDR	2.6e-14	6.5e-14	8.6e-13	0.0006	0.5940	0.0044



(a) ACC

(b) NMI

FIGURE 6. Performance for the noisy label case.

data points increased, the performance of ABDR decreases over the datasets. In terms of ACC, ABDR is shown to be significantly better or at least as good as the baselines for datasets, even with 15% class label noise. However, NMI is more sensitive to the label noise, and ABDR with 15% label noise performs worse on Dig10 and COIL15 than the baseline SBDR.

V. DISCUSSIONS

A. COMPUTATIONAL COMPLEXITY OF ABDR

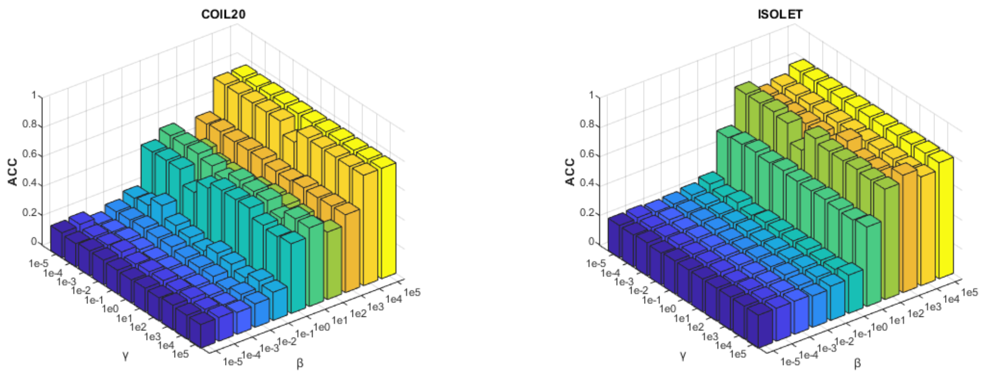
As shown in **Algorithm 2**, ABDR consists of two parts, i.e., BDR-Z in line 2 and S iterations to refine the clustering results in lines 3-13. Due to the spectral clustering for data segmentation and the eigenvalue decomposition in its iterations for getting matrices Z and B, the time complexity of BDR-Z is $O(tn^3)$, where t is the iterative times and n is the number of points. Besides, S iterations in lines 3-13 are composed of three main parts, i.e., active learning, iterations for updating matrices V, B and Z, and the spectral clustering in line 12. The complexity for active learning is $O(nP)$, where P is the number of points to be selected. If the updating process needs K iterations to converge, the computational complexity of this step is $O(KMn^3)$, where M is the iteration number MAXno in **Algorithm 1**. For spectral clustering, the computational complexity is $O(n^3)$. In summary, the total computational complexity of ABDR is $O((t + SKM)n^3)$. Spectral clustering is the main reason of high time complexity, which needs to be investigated in the future work.

B. PARAMETER ANALYSIS

In this section, we tune the parameters used in the experiments. The comparison algorithms URASC, LNSM and DQIC use the parameters suggested by their authors.

1) THE PARAMETER TUNING FOR BDR-Z AND SBDR

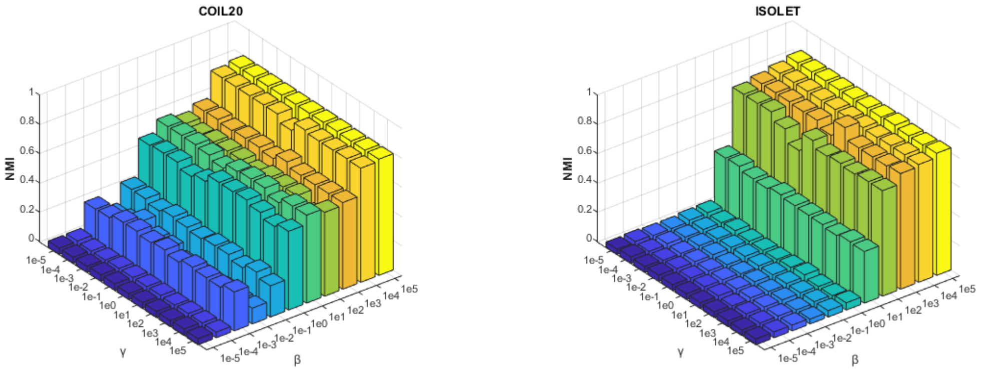
As suggested [6], [28], we conduct BDR-Z and SBDR on the datasets related to MNIST and ORL with parameters $\beta = 70$, $\gamma = 10^{-1}$, which have achieved outstanding



(a) COIL-20.

(b) ISOLET.

FIGURE 7. The averaged ACCs of BDR-Z on datasets COIL-20 and ISOLET.



(a) COIL-20.

(b) ISOLET.

FIGURE 8. The averaged NMIs of BDR-Z on datasets COIL-20 and ISOLET.

clustering results. Besides, the parameter δ in SBDR is suggested to take the value 10^{-1} and 10^{-5} on MNIST and ORL, respectively [28]. To tune the parameters β and γ for datasets COIL-20 and ISOLET, we conduct BDR-Z on the test datasets COIL10 and ISOLET12 by searching the interval $[10^{-5}, 10^{5}]$, the averaged ACCs and NMIs over 20 runs are shown in Figures 7 and 8, respectively. We observe that BDR-Z performs well when β is relatively large, so we set $\beta = 10^5$, $\gamma = 10^5$ and $\beta = 10^5$, $\gamma = 1$ for BDR-Z and SBDR for COIL-20 and ISOLET *related datasets*, respectively.

With fixed β and γ, we run SBDR on COIL10 and ISOLET12 with $\delta \in \{10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 1.0, 10^2, 150, 10^3, 1500\}$. Figure 9 shows that, when δ is equal to 150, the averaged ACCs of SBDR are highest.

Since BDR-Z is the initial step of the proposed ABDR and the foundation of the new objective function, we use the same parameter values of BDR-Z in ABDR. The parameters are summarized in Table 4.

TABLE 4. Parameter values.

Dataset	β	γ	δ	v_0	$p(\%)$
MINST	70	10^{-1}	10^{-1}	0.2	{2,3,4,5,6}
ORL	70	10^{-1}	10^{-5}	0.2	{10,15,20,25,30}
COIL-20	10^5	10^5	150	1.0	{2,3,4,5,6}
ISOLET	10^5	1	150	0.05	{5,10,15,20,25}

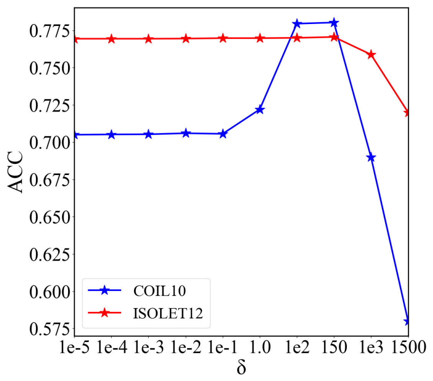
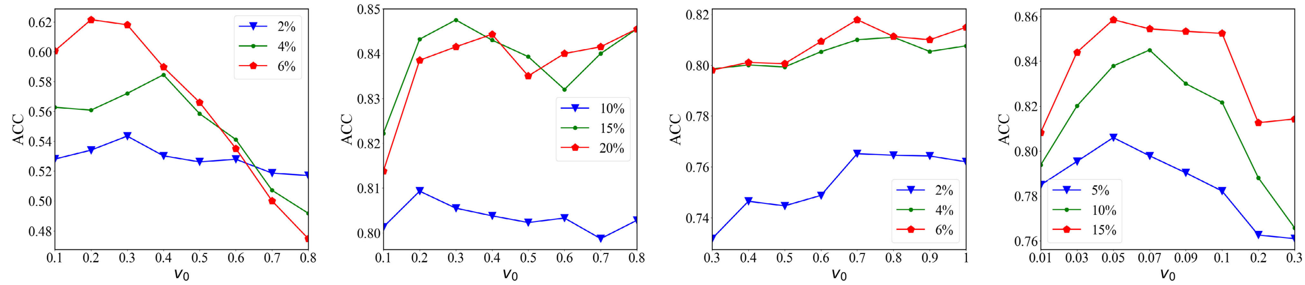


FIGURE 9. The averaged ACCs of SBDR on datasets COIL-20 and ISOLET.

2) THE EFFECT OF v_0

The parameter v_0 is used to enhance the affinity between data points with must-links. Some works set v_0 to 1 [16], [56], [57], or 0.001 for text clustering [46]. Excessive enhancement of affinity will have effects, but not necessarily a good effect, especially for complex clustering tasks.

We record the averaged ACCs of ABDR on four test datasets with β and γ set as Table 4 and v_0 set in the interval



(a) Dig10. (b) ORL30. (c) COIL10. (d) ISOLET12.

FIGURE 10. The clustering accuracy of ABDR with different proportion *p* (e.g., 2%, 4%, and 6% data are actively learned for Dig10).

[0.01, 1.0], as shown in Figure 10. We choose the suitable v_0 for each dataset and show it in Table 4.

C. CONVERGENCE OF ABDR

Wang *et al.* proposed the convergence property of a constrained spectral clustering algorithm as follows [35]:

As the constraint matrix $Q(t)$ at time t approaches the ground truth (complete) constraint matrix Q^*, the output of the constrained clustering algorithm $u(t)$ will converge to the ground truth cluster assignment u^*:

$$\lim_{Q(t)\to Q^*} u(t) = u^*.$$

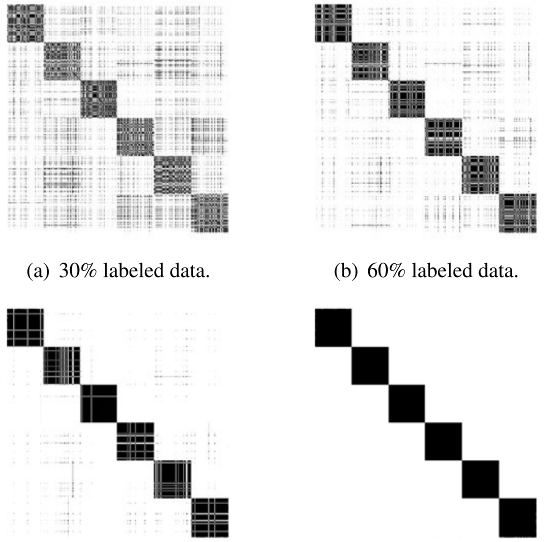
This property ensures that the active learning framework will converge to the ground truth cluster assignment as more constraints are revealed by the oracle [35]. For ABDR, the cluster assignment is obtained by spectral clustering on the affinity matrix, however, spectral clustering is unstable [52] and may give a bad clustering result even the affinity matrix is good for complex clustering tasks. As a weak property, we concern the convergence of the affinity matrix, since a correct affinity matrix is the premise of the correct cluster assignment.

We define the convergence property of the affinity matrix as follows:

As the constraint matrix $Q(t)$ at time t approaches the ground truth (complete) constraint matrix Q^*, the affinity matrix $W(t)$ of the constrained clustering algorithm will converge to the ground truth affinity matrix W^*:

$$\lim_{Q(t)\to Q^*} W(t) = W^*.$$

As for ABDR, the objective function in Section III.B is under the constraint $Z_C = 0$, which zeros the representative coefficients between the known cannot-linked nodes, thus the part of representative matrix Z outside the diagonal blocks is cleaned with the increase of the cannot-links. As more labeled data are revealed by the oracle, the affinity matrix W shows a clear block diagonal. Figure 11 shows the convergence of ABDR on Dig6. Following [16], the affinity matrix W used here is obtained by calculating the Euclidean distance between feature vectors of spectral clustering.



(a) 30% labeled data. (b) 60% labeled data.

(c) 90% labeled data. (d) 100% labeled data.

FIGURE 11. Visualization of affinity matrix W on Dig6.

D. DISCUSSION ON CONSTRAINT METHODS

In this section, we discuss three methods for incorporating pairwise constraints.

1) $Z_M = v_0, Z_C = 0$

Set appropriate fixed value v_0 at the must-linked entries, and 0 at the cannot-linked entries of Z to replace the representative coefficients generated by BDR. This method can enhance the connections between must-linked points, however, it may destroy the natural affinity between data points without pairwise constraints. For an instance, let $X = [x_1, x_2, \ldots, x_n]$, we can get the representation matrix Z_{n*n}, such that, $X = XZ$. Assume that an instance $x_1 = Xz_1$, where, $z_1 = [0, z_{21}, z_{31} \ldots z_{n1}]$ is the self-expressive vector of x_1, and there is a must-link constraint between data instances x_1 and x_2, hence z_{21} is set to v_0. Generally, v_0 is different from the true affinity between x_1 and x_2, and the values of $z_{31}, \ldots, z_{n1}$ are distorted to be adapted to the value of z_{21}, then the biased affinity matrix may lead to the decline of clustering quality. Therefore, this incorporating method has a very high requirement for the quality and quantity of constraints and the setting of v_0.

TABLE 5. ACC of different algorithms and strategies.

Datasets	BDR-Z	SBDR	p(%)	URASC	URASC+AL	ABDR+UR	Rand	LNSM	DQIC	Ours
Dig6	0.6493	0.6390	2	0.5992	0.6038	0.6524	0.6591	0.6576	0.6453	**0.6852**
			3	0.6065	0.6117	0.6559	0.6493	0.6925	0.6668	**0.7183**
			4	0.6065	0.6168	0.6370	0.7137	0.7006	0.6758	**0.7428**
			5	0.6167	0.6531	0.6243	0.6990	0.6902	0.6993	**0.7742**
			6	0.6299	0.6585	0.6203	0.7751	0.7127	0.7340	**0.8068**
Dig8	0.5550	0.5761	2	0.4388	0.3909	0.5711	0.5445	0.5507	**0.5802**	0.5594
			3	0.4756	0.4073	0.5878	0.5546	0.5746	**0.5890**	0.5806
			4	0.4983	0.4133	0.5940	0.5674	0.5837	**0.6039**	0.6022
			5	0.5128	0.4455	0.5713	0.5644	0.6048	**0.6334**	0.6230
			6	0.5255	0.4384	0.5723	0.6301	0.5952	**0.6431**	0.6406
Dig10	0.5251	0.5600	2	0.3658	0.338	**0.5420**	0.5184	0.5158	0.5377	0.5364
			3	0.3788	0.3410	0.5472	0.5227	0.5184	0.5464	**0.5601**
			4	0.4101	0.3490	0.5474	0.5361	0.5369	0.5576	**0.5700**
			5	0.4490	0.3641	0.5194	0.5218	0.5468	0.5625	**0.6051**
			6	0.4840	0.3738	0.5258	0.5931	0.5691	0.5981	**0.6214**
ORL20	0.8322	0.8445	10	0.7898	0.7760	0.8422	0.8330	**0.8612**	0.8478	0.8457
			15	0.7863	0.7905	0.8480	0.8405	**0.8792**	0.8520	0.8557
			20	0.8235	0.7910	0.8670	0.8450	**0.8917**	0.8520	0.8662
			25	0.8060	0.8433	0.8730	0.8447	0.8955	0.8738	**0.9232**
			30	0.8183	0.8275	0.8878	0.8680	0.9137	0.8838	**0.9200**
ORL30	0.7995	0.7992	10	0.7285	0.7425	0.7957	0.7903	**0.8102**	0.8085	0.7975
			15	0.7233	0.7507	0.8050	0.7837	**0.8387**	0.8077	0.8233
			20	0.7445	0.7515	0.8075	0.7987	0.8292	0.8227	**0.8307**
			25	0.7445	0.7833	0.8182	0.8045	0.8547	0.8327	**0.8925**
			30	0.7610	0.7749	0.8377	0.8132	0.8678	0.8470	**0.8842**
ORL40	0.7700	0.7725	10	0.7425	0.7125	0.7400	0.7725	**0.7975**	0.7400	0.7700
			15	0.7375	0.7775	0.8125	**0.8175**	0.7975	0.8000	0.8125
			20	0.7000	0.7450	0.8075	0.7800	**0.8400**	**0.8400**	0.7950
			25	0.7075	0.7200	0.8225	0.7975	0.8300	0.8175	**0.8925**
			30	0.7425	0.7125	0.8075	0.7925	0.8350	0.8250	**0.8675**
COIL5	0.7800	0.7828	2	0.6236	0.7939	0.8153	0.8261	0.8458	0.7947	**0.8519**
			3	0.6300	0.7950	0.8125	0.8486	0.7614	0.8386	**0.8528**
			4	0.6431	0.7997	0.7889	0.8372	0.7733	0.8556	**0.8869**
			5	0.6289	0.8033	0.7614	0.8622	0.7978	0.8683	**0.8922**
			6	0.6453	0.8033	0.7806	**0.8928**	0.7631	0.8769	**0.8928**
COIL10	0.7300	0.7803	2	0.6403	0.6543	0.7610	**0.7817**	0.7592	0.7693	0.7574
			3	0.6233	0.7508	0.7572	**0.7932**	0.7619	0.7797	0.7779
			4	0.6358	0.7478	0.7483	**0.8082**	0.7636	0.7844	0.8076
			5	0.6508	0.7460	0.7644	0.8057	0.7749	0.7697	**0.8131**
			6	0.6438	0.7475	0.7606	0.7994	0.7589	0.7946	**0.8121**
COIL15	0.7207	0.7658	2	0.6919	0.6965	0.7479	0.7562	0.7768	**0.7870**	0.7520
			3	0.7152	0.7281	0.7592	0.7745	**0.7854**	.7838	0.7602
			4	0.7156	0.7505	0.7545	0.7872	0.7886	**0.8174**	0.8063
			5	0.7023	0.7634	0.7330	0.8025	0.7600	**0.8153**	0.7956
			6	0.6858	0.7547	0.7459	0.7907	0.7643	**0.8200**	0.7982
ISOLET6	0.8342	0.8389	5	0.8378	0.8372	0.8331	0.8417	0.8414	0.8394	**0.8556**
			10	**0.8831**	0.8397	**0.8831**	0.8453	0.8519	0.8592	0.8667
			15	**0.9075**	0.8467	0.8967	0.8347	0.8978	0.8539	0.8808
			20	0.9303	0.8556	**0.9311**	0.8706	0.9139	0.9086	0.9100
			25	**0.9478**	0.8581	0.9469	0.8853	0.9319	0.9048	0.9372
ISOLET12	0.7781	0.7706	5	0.7349	0.7122	**0.8218**	0.7931	0.8053	0.8078	0.8039
			10	0.7015	0.7399	0.8404	0.8076	0.8313	0.8157	**0.8458**
			15	0.7243	0.7644	0.8375	0.8429	0.8413	0.8339	**0.8550**
			20	0.7181	0.8125	0.8424	0.8601	0.8425	**0.8732**	0.8690
			25	0.7303	0.8482	0.8453	0.8901	0.8733	**0.9099**	0.8928
ISOLET18	0.7133	0.7081	5	0.6619	0.6646	0.6619	0.7133	0.7336	0.7475	**0.7491**
			10	0.6497	0.6855	0.7607	0.7556	0.7528	0.7556	**0.7613**
			15	0.6568	0.7150	0.7707	0.7737	0.7782	0.7847	**0.7862**
			20	0.6631	0.7467	0.7737	0.7845	0.7888	**0.8180**	0.8016
			25	0.6518	0.7610	0.7907	0.8359	0.8194	**0.8457**	0.8230

TABLE 6. NMI of different algorithms and strategies.

Datasets	BDR-Z	SBDR	p(%)	URASC	URASC+AL	ABDR+UR	Rand	LNSM	DQIC	Ours
Dig6	0.5843	0.6254	2	0.5528	0.5810	0.5833	0.5876	0.5986	0.5866	**0.6186**
			3	0.5522	0.5780	0.5845	0.6094	0.6198	0.6204	**0.6341**
			4	0.5630	0.5887	0.5771	0.6204	0.6344	0.6081	**0.6606**
			5	0.5607	0.5831	0.5607	0.6183	0.6338	0.6169	**0.6734**
			6	0.5695	0.5982	0.5615	0.6463	0.6486	0.6280	**0.7093**
Dig8	0.5565	0.6074	2	0.5098	0.4753	0.5630	0.5521	0.5568	0.5743	**0.5750**
			3	0.5161	0.4818	0.5624	0.5594	0.5701	**0.5848**	0.5806
			4	0.5377	0.4851	0.5545	0.5752	0.5866	0.5993	**0.6098**
			5	0.5462	0.4988	0.5387	0.5891	0.5969	0.6101	**0.6222**
			6	0.5499	0.5084	0.5365	0.6120	0.6022	0.6114	**0.6409**
Dig10	0.5354	0.6077	2	0.4472	0.4280	**0.5587**	0.5335	0.5408	0.5427	0.5573
			3	0.4734	0.4373	0.5604	0.5409	0.5519	0.5549	**0.5695**
			4	0.4850	0.4357	0.5518	0.5462	0.5584	0.5573	**0.5793**
			5	0.5030	0.4435	0.5377	0.5786	0.5494	0.5639	**0.6000**
			6	0.5110	0.4316	0.5228	0.5862	0.5914	0.5720	**0.6189**
ORL20	0.9041	0.9108	10	0.8939	0.8898	0.9112	0.9074	**0.9242**	0.9114	0.9130
			15	0.8973	0.8931	0.9135	0.9118	**0.9327**	0.9140	0.9244
			20	0.9121	0.8972	0.9234	0.9105	**0.9392**	0.9136	0.9302
			25	0.9126	0.9118	0.9259	0.9306	0.9420	0.9301	**0.9590**
			30	0.9199	0.9158	0.9356	0.9226	0.9539	0.9333	**0.9583**
ORL30	0.8976	0.9014	10	0.8781	0.8766	0.9005	0.8951	**0.9062**	0.8985	0.9018
			15	0.8752	0.8792	0.9030	0.9040	**0.9217**	0.9056	0.9201
			20	0.8853	0.8844	0.9051	0.9013	0.9219	0.9109	**0.9228**
			25	0.8862	0.9080	0.9120	0.9155	0.9330	0.9164	**0.9519**
			30	0.8978	0.9045	0.9191	0.9095	0.9383	0.9234	**0.9504**
ORL40	0.8999	0.8941	10	0.8663	0.8760	0.8788	0.8971	0.8972	0.8843	**0.9018**
			15	0.8737	0.8927	0.9085	0.9046	0.9142	0.8963	**0.9210**
			20	0.8629	0.8836	0.9066	0.9178	**0.9227**	0.9051	0.9132
			25	0.8728	0.8980	0.9041	0.9089	0.9258	0.9051	**0.9563**
			30	0.8756	0.9102	0.9089	0.9053	0.9318	0.9126	**0.9440**
COIL5	0.7488	0.7895	2	0.6884	0.8079	0.8070	0.8085	0.8017	0.7745	**0.8277**
			3	0.6791	0.8092	0.8026	0.8163	0.7720	0.8232	**0.8281**
			4	0.6902	0.8232	0.7845	0.8189	0.7778	0.8191	**0.8529**
			5	0.6826	0.8308	0.7655	0.8348	0.8094	0.8434	**0.8633**
			6	0.6932	0.8308	0.7804	0.8453	0.8050	0.8393	**0.8647**
COIL10	0.7823	0.8521	2	0.7340	0.7310	0.8115	**0.8231**	0.8057	0.8063	0.8013
			3	0.7142	0.8170	0.8094	**0.8253**	0.8233	0.8169	0.8186
			4	0.7181	0.8149	0.8077	0.8288	0.8242	0.8198	**0.8327**
			5	0.7628	0.8162	0.8055	0.8345	0.8169	0.8117	**0.8350**
			6	0.7231	0.8188	0.8006	0.8317	0.8156	0.8179	**0.8342**
COIL15	0.8118	0.8732	2	0.8117	0.7920	0.8283	0.8375	0.8468	**0.8539**	0.8224
			3	0.8147	0.8196	0.8312	0.8482	**0.8489**	0.8479	0.8438
			4	0.8169	0.8335	0.8179	0.8523	0.8599	**0.8627**	0.8589
			5	0.8086	0.8377	0.8195	**0.8603**	0.8475	0.8511	0.8543
			6	0.8034	0.8319	0.8129	0.8536	0.8558	**0.8596**	0.8539
ISOLET6	0.8560	0.8656	5	0.8540	0.8529	0.8540	0.8604	0.8636	0.8624	**0.8638**
			10	0.8747	0.8571	0.8540	0.8663	**0.8763**	0.8734	0.8691
			15	0.8941	0.8651	0.8767	0.8697	**0.8954**	0.8725	0.8740
			20	0.9030	0.8682	0.8908	0.8999	**0.9182**	0.9049	0.8657
			25	0.9209	0.8688	0.9032	0.9201	**0.9264**	0.9049	0.8680
ISOLET12	0.8102	0.8075	5	0.8099	0.8066	0.8099	0.8293	0.8371	0.8367	**0.8375**
			10	0.8141	0.8284	0.8474	0.8360	0.8602	0.8372	**0.8651**
			15	0.8300	0.8410	0.8531	0.8527	0.8635	0.8568	**0.8742**
			20	0.8395	0.8715	0.8566	0.8695	0.8761	0.8746	**0.8895**
			25	0.8368	0.8872	0.8622	**0.9031**	0.8893	0.8938	0.8994
ISOLET18	0.7824	0.7804	5	0.7775	0.7873	0.7775	**0.8233**	0.8063	0.8065	0.8154
			10	0.7851	0.8127	0.8179	**0.8317**	0.8230	0.8106	0.8297
			15	0.7848	0.8245	0.8222	0.8382	0.8383	0.8346	**0.8437**
			20	0.7970	0.8388	0.8198	**0.8722**	0.8508	0.8492	0.8604
			25	0.7961	0.8489	0.8328	**0.8920**	0.8616	0.8616	0.8689

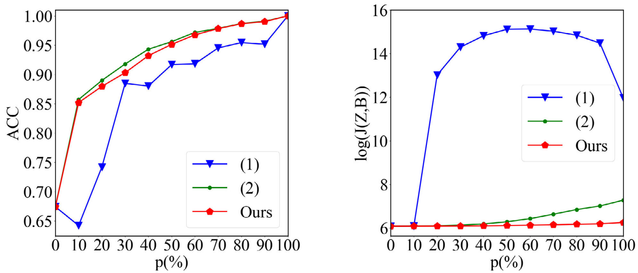
2) $Z_{M \cup C} = 0$ [16], [29]

Set 0 at all constrained entries in the representation matrix, and then reconstruct the affinity matrix W for the spectral clustering, i.e., W_{ij} is set to v_0, if x_i and x_j are must-linked, and 0 if they are cannot-linked. This method reduces the noise in the representation matrix, however, setting v_0 among all the same class data will inevitably cause a large reconstruction error.

3) $Z_C = 0$ (THE PROPOSED ONE)

Similar to but different from the second method, ABDR sets 0 at the cannot-linked entries of representation matrix Z, and leaves the representative coefficients between must-linked points to be calculated by the algorithm, which is more consistent with the natural subspace structure. The must-link constraints are used to enhance the affinity between the corresponding pairwise points if the natural affinity values are too small, as shown in (4).

The test dataset Dig6 is used to compare three incorporating methods. The labeled data are acquired by the proposed active learning strategy, and the sampling proportion varied from 0% to 100%. Specifically, v_0 is set at 0.2, and the results are shown in Figure 12. Figure 12(a) shows that the second method is approximately equal to ours in terms of ACC, and the first one is worst performed. We calculate the logarithm of the objective function values of ABDR armed with the three methods, as shown in Figure 12(b). As expected, the first method produces the largest objective function value, and ours is the lowest and the best one.



(a) ACC. (b) $log(J(Z, B))$.

FIGURE 12. Comparison of incorporating methods on Dig6.

VI. CONCLUSION

Combination of active learning and constrained clustering is a trivial way of incorporating knowledge into the clustering process. In this paper, we constrain the subspace clustering algorithm BDR-Z with pairwise constraints, which are converted from the labeled data revealed by the oracle. We optimize the objective function of the proposed ABDR, and discuss its convergence. Furthermore, the three incorporating methods are compared and evaluated on the test datasets. Experimental results validate the effectiveness of ABDR on complex clustering tasks. In terms of W/T/L and statistical tests, our proposed constrained framework ABDR and active selection strategy outperform the compared state-of-art algorithms.

In future research, we plan to improve and explore ABDR in the following aspects.

- Add regularization terms into the objective function of ABDR for better recovery of subspace structures from noisy data, and explore novel incorporating methods to increase the utilization of limited human resources.
- Alleviate the time cost by approximating the affinity matrix with carefully selected representatives [51], [53]. Since spectral clustering is the main reason of high computational complexity, the efforts on scalability and robustness of spectral clustering are beneficial to our work, especially for large scale clustering problems.
- Investigate the ensemble clustering of ABDR to improve the robustness of clustering results further [58].

APPENDIX

See Tables 5 and 6.

REFERENCES

[1] L. Jing, M. K. Ng, and J. Z. Huang, "An entropy weighting k-means algorithm for subspace clustering of high-dimensional sparse data," *IEEE Trans. Knowl. Data Eng.*, vol. 19, no. 8, pp. 1026–1041, Aug. 2007.

[2] W. Hong, J. Wright, K. Huang, and Y. Ma, "Multiscale hybrid linear models for lossy image representation," *IEEE Trans. Image Process.*, vol. 15, no. 12, pp. 3655–3671, Dec. 2006.

[3] R. Basri and D. W. Jacobs, "Lambertian reflectance and linear subspaces," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 2, pp. 218–233, Feb. 2003.

[4] E. Elhamifar and R. Vidal, "Sparse subspace clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 2790–2797.

[5] H. Zhao, Z. Ding, and Y. Fu, "Ensemble subspace segmentation under blockwise constraints," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 7, pp. 1526–1539, Jul. 2018.

[6] C. Lu, J. Feng, Z. Lin, T. Mei, and S. Yan, "Subspace clustering by block diagonal representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 2, pp. 487–501, Feb. 2019.

[7] G. Liu, Z. Lin, and Y. Yu, "Robust subspace segmentation by low-rank representation," in *Proc. 27th Int. Conf. Mach. Learn. (ICML)*, 2010, pp. 663–670.

[8] J. Chen and J. Yang, "Robust subspace segmentation via low-rank representation," *IEEE Trans. Cybern.*, vol. 44, no. 8, pp. 1432–1445, Aug. 2014.

[9] Y. Yang and X. Zhang, "Block-diagonal subspace clustering with Laplacian rank constraint," in *Proc. IEEE 3rd Inf. Technol., Netw., Electron. Autom. Control Conf. (ITNEC)*, Mar. 2019, pp. 1556–1559.

[10] R. Vidal, Y. Ma, and S. Sastry, "Generalized principal component analysis (GPCA)," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 12, pp. 1945–1959, Dec. 2005.

[11] L. Wang, J. Huang, M. Yin, R. Cai, and Z. Hao, "Block diagonal representation learning for robust subspace clustering," *Inf. Sci.*, vol. 526, pp. 54–67, Jul. 2020.

[12] Z. Zhang, Y. Zhang, G. Liu, J. Tang, S. Yan, and M. Wang, "Joint label prediction based semi-supervised adaptive concept factorization for robust data representation," *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 5, pp. 952–970, May 2020.

[13] M. Yin, S. Xie, Z. Wu, Y. Zhang, and J. Gao, "Subspace clustering via learning an adaptive low-rank graph," *IEEE Trans. Image Process.*, vol. 27, no. 8, pp. 3716–3728, Aug. 2018.

[14] W. Wang, C. Yang, H. Chen, and X. Feng, "Unified discriminative and coherent semi-supervised subspace clustering," *IEEE Trans. Image Process.*, vol. 27, no. 5, pp. 2461–2470, May 2018.

[15] R. He, Y. Zhang, Z. Sun, and Q. Yin, "Robust subspace clustering with complex noise," *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 4001–4013, Nov. 2015.

[16] Y. Liu, K. Liu, C. Zhang, X. Wang, S. Wang, and Z. Xiao, "Entropy-based active sparse subspace clustering," *Multimedia Tools Appl.*, vol. 77, no. 17, pp. 22281–22297, Sep. 2018.

[17] J. Wang, X. Wang, F. Tian, C. H. Liu, and H. Yu, ''Constrained low-rank representation for robust subspace clustering,'' *IEEE Trans. Cybern.*, vol. 47, no. 12, pp. 4534–4546, Dec. 2017.

[18] J. E. van Engelen and H. H. Hoos, ''A survey on semi-supervised learning,'' *Mach. Learn.*, vol. 109, no. 2, pp. 373–440, Feb. 2020.

[19] K. Wagstaff, C. Cardie, S. Rogers, and S. Schrödl, ''Constrained k-means clustering with background knowledge,'' in *Proc. ICML*, vol. 1, 2001, pp. 577–584.

[20] A. A. Abin and V.-V. Vu, ''A density-based approach for querying informative constraints for clustering,'' *Expert Syst. Appl.*, vol. 161, Dec. 2020, Art. no. 113690, doi: 10.1016/j.eswa.2020.113690.

[21] S. Huang, H. Zhang, and A. Pizurica, ''Semisupervised sparse subspace clustering method with a joint sparsity constraint for hyperspectral remote sensing images,'' *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 12, no. 3, pp. 989–999, Mar. 2019.

[22] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, ''Gradient-based learning applied to document recognition,'' *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[23] F. S. Samaria and A. C. Harter, ''Parameterisation of a stochastic model for human face identification,'' in *Proc. IEEE Workshop Appl. Comput. Vis.*, Dec. 1994, pp. 138–142.

[24] S. A. Nene, S. K. Nayar, and H. Murase, ''Columbia object image library (COIL-20),'' Columbia Univ., New York, NY, USA, Tech. Rep. CUCS-005-96, 1996.

[25] K. Bache and M. Lichman. (2021). *UCI Machine Learning Repository*. [Online]. Available: http://archive.ics.uci.edu/ml

[26] U. von Luxburg, ''A tutorial on spectral clustering,'' *Statist. Comput.*, vol. 17, no. 4, pp. 395–416, Dec. 2007.

[27] C.-G. Li, C. You, and R. Vidal, ''Structured sparse subspace clustering: A joint affinity learning and subspace clustering framework,'' *IEEE Trans. Image Process.*, vol. 26, no. 6, pp. 2988–3001, Jun. 2017.

[28] M. Liu, Y. Wang, J. Sun, and Z. Ji, ''Structured block diagonal representation for subspace clustering,'' *Appl. Intell.*, vol. 50, pp. 2523–2536, Mar. 2020.

[29] Z. Xie and L. Wang, ''Active structure learning for block diagonal subspace clustering,'' in *Proc. 13th Int. Congr. Image Signal Process., Biomed. Eng. Informat. (CISP-BMEI)*, Oct. 2020, pp. 926–931.

[30] Y. Xu, S. Chen, J. Li, Z. Han, and J. Yang, ''Autoencoder-based latent block-diagonal representation for subspace clustering,'' *IEEE Trans. Cybern.*, early access, Nov. 18, 2020, doi: 10.1109/TCYB.2020.3031666.

[31] J. Guo, W. Yin, Y. Sun, and Y. Hu, ''Multi-view subspace clustering with block diagonal representation,'' *IEEE Access*, vol. 7, pp. 84829–84838, 2019.

[32] C. Chen, H. Qian, W. Chen, Z. Zheng, and H. Zhu, ''Auto-weighted multi-view constrained spectral clustering,'' *Neurocomputing*, vol. 366, pp. 1–11, Nov. 2019.

[33] X. Wang, S. Ding, and W. Jia, ''Active constraint spectral clustering based on Hessian matrix,'' *Soft Comput.*, vol. 24, no. 3, pp. 2381–2390, Feb. 2020.

[34] A. Biswas and D. Jacobs, ''Large scale image clustering with active pair-wise constraints,'' in *Proc. Int. Conf. Mach. Learn. Workshop Combining Learn. Strategies Reduce Label Cost*, 2011, vol. 2, no. 6, pp. 1–7.

[35] X. Wang and I. Davidson, ''Active spectral clustering,'' in *Proc. IEEE Int. Conf. Data Mining*, Dec. 2010, pp. 561–568.

[36] A. Biswas and D. Jacobs, ''Active image clustering with pairwise constraints from humans,'' *Int. J. Comput. Vis.*, vol. 108, nos. 1–2, pp. 133–147, May 2014.

[37] C. Xiong, D. Johnson, and J. J. Corso, ''Online active constraint selection for semi-supervised clustering,'' in *Proc. Eur. Conf. Artif. Intell., Act. Incremental Learn. Workshop*, 2012, pp. 1–6.

[38] C. Xiong, D. M. Johnson, and J. J. Corso, ''Active clustering with model-based uncertainty reduction,'' *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 1, pp. 5–17, Jan. 2017.

[39] P. Qian, Y. Jiang, S. Wang, K.-H. Su, J. Wang, L. Hu, and R. F. Muzic, ''Affinity and penalty jointly constrained spectral clustering with all-compatibility, flexibility, and robustness,'' *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 5, pp. 1123–1138, May 2017.

[40] K. L. Wagstaff, ''Value, cost, and sharing: Open issues in constrained clustering,'' in *Knowledge Discovery in Inductive Databases*. Berlin, Germany: Springer, 2006, pp. 1–10.

[41] M. A. Masud, J. Z. Huang, M. Zhong, and X. Fu, ''Generate pairwise constraints from unlabeled data for semi-supervised clustering,'' *Data Knowl. Eng.*, vol. 123, Sep. 2019, Art. no. 101715, doi: 10.1016/j.datak.2019.101715.

[42] X. Fang, Y. Xu, X. Li, Z. Lai, and W. K. Wong, ''Robust semi-supervised subspace clustering via non-negative low-rank representation,'' *IEEE Trans. Cybern.*, vol. 46, no. 8, pp. 1828–1838, Aug. 2016.

[43] X. Fang, Z. Tie, F. Song, and J. Yang, ''Robust subspace clustering via symmetry constrained latent low rank representation with converted nuclear norm,'' *Neurocomputing*, vol. 340, pp. 211–221, May 2019.

[44] A. A. Abin, ''Querying informative constraints for data clustering: An embedding approach,'' *Appl. Soft Comput.*, vol. 80, pp. 31–41, Jul. 2019.

[45] D. Klein, S. D. Kamvar, and C. D. Manning, ''From instance-level constraints to space-level constraints: Making the most of prior knowledge in data clustering,'' in *Proc. 19th Int. Conf.*, 2002, pp. 307–314.

[46] S. Basu, A. Banerjee, and R. J. Mooney, ''Active semi-supervision for pairwise constrained clustering,'' in *Proc. SIAM Int. Conf. Data Mining (SDM)*, Apr. 2004, pp. 333–344.

[47] Q. Xu, M. D. Jardins, and K. L. Wagstaff, ''Active constrained clustering by examining spectral eigenvectors,'' in *Proc. Int. Conf. Discovery Sci.* Singapore: Springer, 2005, pp. 294–307.

[48] H. Liu, Z. Tao, and Y. Fu, ''Partition level constrained clustering,'' *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 10, pp. 2469–2483, Oct. 2018.

[49] F. Nie, H. Wang, X. Cai, H. Huang, and C. Ding, ''Robust matrix completion via joint schatten p-norm and lp-norm minimization,'' in *Proc. IEEE 12th Int. Conf. Data Mining*, Dec. 2012, pp. 566–574.

[50] J. Bae, T. Helldin, M. Riveiro, S. Nowaczyk, M.-R. Bouguelia, and G. Falkman, ''Interactive clustering: A comprehensive review,'' *ACM Comput. Surv.*, vol. 53, no. 1, pp. 1–39, May 2020.

[51] D. Cai and X. Chen, ''Large scale spectral clustering via landmark-based sparse representation,'' *IEEE Trans. Cybern.*, vol. 45, no. 8, pp. 1669–1680, Aug. 2015.

[52] E. Andreotti, D. Edelmann, N. Guglielmi, and C. Lubich, ''Measuring the stability of spectral clustering,'' *Linear Algebra Appl.*, vol. 610, pp. 673–697, Feb. 2021.

[53] D. Huang, C.-D. Wang, J.-S. Wu, J.-H. Lai, and C.-K. Kwoh, ''Ultra-scalable spectral clustering and ensemble clustering,'' *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 6, pp. 1212–1226, Jun. 2020.

[54] M. Friedman, ''The use of ranks to avoid the assumption of normality implicit in the analysis of variance,'' *J. Amer. Stat. Assoc.*, vol. 32, no. 200, pp. 675–701, Dec. 1937.

[55] P. B. Nemenyi, ''Distribution-free multiple comparisons,'' Ph.D. dissertation, Dept. Math., Princeton Univ., Princeton, NJ, USA, 1963.

[56] B. Wu, Y. Zhang, B.-G. Hu, and Q. Ji, ''Constrained clustering and its application to face clustering in videos,'' in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2013, pp. 3507–3514.

[57] J. Lipor and L. Balzano, ''Leveraging union of subspace structure to improve constrained clustering,'' in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 2130–2139.

[58] D. Huang, C.-D. Wang, H. Peng, J. Lai, and C.-K. Kwoh, ''Enhanced ensemble clustering via fast propagation of cluster-wise similarities,'' *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 51, no. 1, pp. 508–520, Jan. 2021.

ZIQI XIE was born in Shandong, China, in 1996. He received the B.S. degree in computer science and technology from the Shandong University of Technology, China, in 2019. He is currently pursuing the M.S. degree in computer science with Yantai University, under the supervision of Prof. Lihong Wang. His main research interests include data mining and subspace clustering.

LIHONG WANG was born in Jilin, China, in 1970. She received the B.S. degree from Tsinghua University, China, in 1990, the M.S. degree from the University of Science and Technology of China (USTC), in 1993, and the Ph.D. degree from Shanghai University, China, in 2004. She is currently a Professor with the School of Computer and Control Engineering, Yantai University. Her research interests include data mining and machine learning.