Fast and Robust Sparsity-Aware Block Diagonal Representation

Aylin Taştan, Michael Muma, *Senior Member, IEEE*, and Abdelhak M. Zoubir, *Life Fellow, IEEE*

***Abstract*—The block diagonal structure of an affinity matrix is a commonly desired property in cluster analysis because it represents clusters of feature vectors by non-zero coefficients that are concentrated in blocks. However, recovering a block diagonal affinity matrix is challenging in real-world applications, in which the data may be subject to outliers and heavy-tailed noise that obscure the hidden cluster structure. To address this issue, we first analyze the effect of different fundamental outlier types in graph-based cluster analysis. A key idea that simplifies the analysis is to introduce a vector that represents a block diagonal matrix as a piece-wise linear function of the similarity coefficients that form the affinity matrix. We reformulate the problem as a robust piece-wise linear fitting problem and propose a Fast and Robust Sparsity-Aware Block Diagonal Representation (FRS-BDR) method, which jointly estimates cluster memberships and the number of blocks. Comprehensive experiments on a variety of real-world applications demonstrate the effectiveness of FRS-BDR in terms of clustering accuracy, robustness against corrupted features, computation time and cluster enumeration performance.**

***Index Terms*—Block diagonal representation, affinity matrix, similarity matrix, eigenvalues, subspace clustering.**

I. Introduction

A BLOCK diagonally structured affinity matrix represents clusters of feature vectors by non-zero coefficients that are concentrated in blocks. Such a structure is an informative model to describe hidden relationships. It has numerous applications, e.g., denoising [1], [2], recognition [3], semi-supervised learning [4], [5], [6], subspace learning and clustering/classification [6], [7], [8], [9], [10], [11], [12], [13], [14], [15], [16].

Manuscript received 23 March 2023; revised 5 September 2023 and 17 November 2023; accepted 17 November 2023. Date of publication 19 December 2023; date of current version 29 December 2023. The work of A. Taştan was supported by the Republic of Turkey Ministry of National Education. The work of M. Muma was supported in part by the LOEWE initiative (Hesse, Germany) within the emergenCITY centre and in part by the ERC Starting Grant ScReeningData under Project 101042407. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. George Atia. *(Corresponding author: Aylin Taştan.)*

Aylin Taştan was with the Signal Processing Group, Technische Universität Darmstadt, 64283 Darmstadt, Germany. She is now with the Pattern Recognition Group, University of Bern, CH-3012 Bern, Switzerland (e-mail: a.tastan@spg.tu-darmstadt.de; aylin.tastan@unibe.ch).

Michael Muma is with the Robust Data Science Group, Technische Universität Darmstadt, 64283 Darmstadt, Germany (e-mail: michael.muma@tu-darmstadt.de).

Abdelhak M. Zoubir is with the Signal Processing Group, Technische Universität Darmstadt, 64283 Darmstadt, Germany (e-mail: zoubir@ieee.org).

This article has supplementary downloadable material available at https://doi.org/10.1109/TSP.2023.3343565, provided by the authors.

Digital Object Identifier 10.1109/TSP.2023.3343565

Commonly used existing block diagonal representation (BDR) methods impose a structure on the affinity matrix using regularization with block diagonal (BD) priors, e.g. based on a low-rank property [17], [18], [19], [20], sparsity [21], [22], [23] or a known number of blocks K [6], [7], [8], [9]. For example the method in [6], which is one of the current benchmarks BDR methods, controls the number of connected components in the affinity matrix by imposing a rank constraint on the Laplacian matrix. An alternative popular approach [7], proposes a K-block regularizer that is defined by the sum of the K smallest eigenvalues of the Laplacian matrix to compute a BD affinity matrix. A major challenge of these methods is the need to determine appropriate BD priors which play a crucial role in achieving accurate BDR results. Due to its key role in BDR methods, the determination of sparsity/low-rank level has been intensively investigated from different viewpoints, e.g. similarity coefficients' distribution [24], connectedness [25], geometric analysis [26] and supervised learning [27], [28]. Recently, in [9], an alternative unsupervised approach based on eigenvalues has been proposed to deduce the sparsity level in a BD matrix. The eigenvalue analysis is, however, restricted to the setting of independent blocks.

A further significant challenge when working with real-world data is the presence of heavy-tailed noise and outliers [29], [30], [31], that might obscure the eigenvalue structure in corrupted data sets. This results in a performance degradation for BDR approaches that rely on estimating eigenvalues to determine connectedness. To illustrate the necessity for robustness, a graph partitioning application is shown in Fig. 1 for a defined level of sparsity using the well-known handwritten digit samples from the MNIST data base [32]. In the exemplary graph model, the red edges represent connections to outliers while the remaining edges are the informative edges, where green, blue and yellow lines represent the within-cluster edges of digits 9, 4 and 3, respectively. The red ellipses indicate cluster assignments that are computed based on the general graph partitioning principle, in which the number of edges that cross the cut is minimized [33]. As can be seen, unconnected outlying digit samples ('Type I outliers') are assigned into a small cluster while a different type of outliers ('Type II outliers') that create false positive connections between multiple clusters cause a merging of characters four and nine into one large cluster.

In this work, we propose a method for robustly estimating an underlying BD structure, given an outlier-corrupted affinity matrix. We call this method: *Fast and Robust Sparsity-Aware Block Diagonal Representation*. We build upon the definition

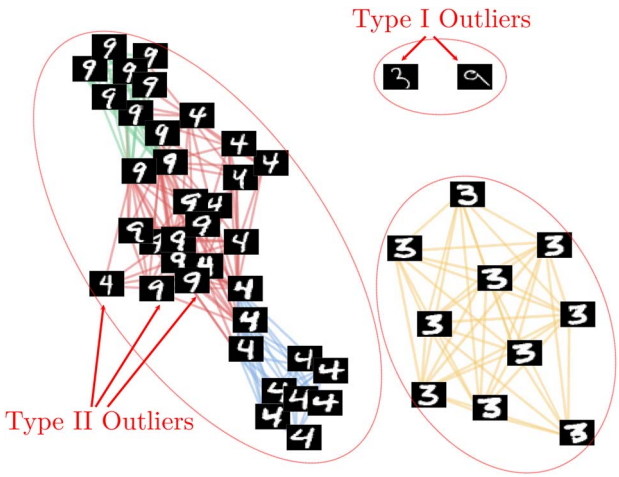


Fig. 1. Exemplary graph partitioning digit samples from MNIST data base [32].

of a vector $\mathbf{v}$ that we recently introduced in [9] to represent the BD affinity matrix as a piece-wise linear function. Compared to existing popular BDR approaches, such as, [6], [7], [8], the optimization is efficiently performed in a vector space instead of matrix space. Additionally and in contrast to [9], the method is robust against outliers. Our main contributions are summarized as follows:

1) We perform comprehensive robustness analysis that quantifies the effects of outliers. In particular, our theoretical analysis shows how the vector $\mathbf{v}$ and the eigenvalues, which carry substantial information about the BD structure, are influenced by outliers.
2) Our analysis enables the development of a BDR algorithm that is (i) robust against outliers that obscure the target BD structure and (ii) computationally efficient by re-formulating the problem as a piece-wise linear function optimization instead of a matrix-optimization. We show that our proposed method provides mathematically interpretable results in challenging settings where deriving eigenvalue information is no longer possible (i.e., in the extreme case when all blocks are connected because of corruption by outliers).

The paper is organized as follows. Section II contains a summary of notations and a brief discussion on eigen-decomposition. The detailed eigenvalue analysis and outlier effects are presented in Section III. The simplification of the graph Laplacian matrix analysis by means of vector $\mathbf{v}$ and the associated outlier effect analysis are the subject of Section IV. The proposed FRS-BDR method is detailed in Section V and experimental evaluations demonstrating the performance of FRS-BDR in comparison to popular BDR approaches are shown in Section VI. Finally, conclusions are drawn in Section VII. The codes that implement the FRS-BDR method are available at: https://github.com/A-Tastan/FRS-BDR.

II. Preliminaries

A. Summary of Notation

Lower and upper-case bold letters denote vectors and matrices, respectively; $|x|$ denotes the absolute value of x; $\|\mathbf{x}\|$ denotes the norm of vector $\mathbf{x}$ while $\mathrm{med}(\mathbf{x})$ denotes its median; $\mathrm{sign}(x) = x/|x|$; $\mathrm{diag}(x_1, ..., x_N)$ denotes a diagonal matrix of size $N \times N$ with $x_1, ..., x_N$ on its diagonal; $\mathbf{I}$ denotes the identity matrix; $\mathbf{1}$ denotes the column vector of ones; $\hat{\mathbf{x}}$ denotes the estimate of vector $\mathbf{x}$; $\tilde{\mathbf{W}}$ refers to a corrupted affinity matrix; i, j and k are index operators for the blocks, e.g. $\tilde{\mathbf{W}}_i$ denotes ith block in $\tilde{\mathbf{W}}$; m, n and r are index operators for the samples; finally I and II denote, respectively, index operators for the Type I and Type II outliers.

B. Eigen-Decomposition of Laplacian Matrix

Let data set $\mathbf{X} = [\mathbf{x}_1 \dots \mathbf{x}_N] \in \mathbb{R}^{M \times N}$ with M denoting the feature dimension and N being the number of feature vectors, be represented as a graph $G = \{V, E, \mathbf{W}\}$, where V denotes the vertices, E represents the edges, and $\mathbf{W} \in \mathbb{R}^{N \times N}$ is the symmetric affinity matrix. The affinity matrix is computed from $\mathbf{X}$ by choosing an appropriate similarity measure, such as, the cosine similarity for which $w_{m,n} = \mathbf{x}_m^\top \mathbf{x}_n$, $m \neq n$ s.t. $\|\mathbf{x}_m\|_2 = 1$, $\|\mathbf{x}_n\|_2 = 1$. Let $\mathbf{L} \in \mathbb{R}^{N \times N}$ denote the nonnegative definite Laplacian matrix that is defined by the eigen-problem

$$\mathbf{L}\mathbf{y}_m = \lambda_m \mathbf{y}_m, \tag{1}$$

or in a generalized eigenvalue problem form

$$\mathbf{L}\mathbf{y}_m = \lambda_m \mathbf{D}\mathbf{y}_m, \tag{2}$$

with associated eigenvalues $0 \leq \lambda_0 \leq \lambda_1 \leq ... \leq \lambda_{N-1}$ sorted in ascending order. Here, $\mathbf{L} = \mathbf{D} - \mathbf{W}$, where $\mathbf{D} \in \mathbb{R}^{N \times N}$ is a diagonal weight matrix with edge weights $d_{m,m} = \sum_n w_{m,n}$ on the diagonal, λ_m denotes the mth eigenvalue and $\mathbf{y}_m \in \mathbb{R}^N$ is the eigenvector associated with λ_m.

III. Eigenvalue Analysis and Outlier Effects

The eigen-decomposition of a Laplacian matrix has numerous applications [33], [34], [35], [36], [37] and, in particular, it plays a crucial role in graph-based cluster analysis [14], [38], [39], [40], [41], [42], [43], [44]. However, isolated outliers and outliers that induce undesired correlations between different clusters may negatively impact the eigen-decomposition, leading to a breakdown of clustering algorithms [38], [39]. Section III-A summarizes briefly our previous findings in [9]. A new series of solutions based on the standard eigen-decomposition in Eq. (1) is provided in Appendix B of the accompanying material [45]. Then, the effect of outliers and group similarity on eigenvalues is analyzed in Section III-B for both eigen-decompositions, i.e. for Eqs. (1) and (2).

A. Target Eigenvalues for Graph-Based Clustering

As graph partitioning approaches seek to partition the set of vertices in G into disjoint sets and minimizing the number of the edges that cross the cut [26], [46], [47], an ideal, i.e., target BD affinity matrix is defined in [9] as follows.

Definition III.1: (Target BD Affinity Matrix, [9]). Let $\mathbf{W} \in \mathbb{R}^{N \times N}$ be a K block zero-diagonal symmetric affinity matrix with blocks $\mathbf{W}_1, \mathbf{W}_2, ..., \mathbf{W}_K$ on its diagonal. Each block $\mathbf{W}_i$, $i = 1, ..., K$ is associated with a number $N_i \in \mathbb{Z}_+ > 1$ of feature vectors and concentrated around a similarity constant $w_i \in \mathbb{R}^+, i = 1, ..., K$ with negligibly small variations. $\mathbf{W}$ is called the target affinity matrix if and only if the similarity coefficients between different blocks are all zero-valued.

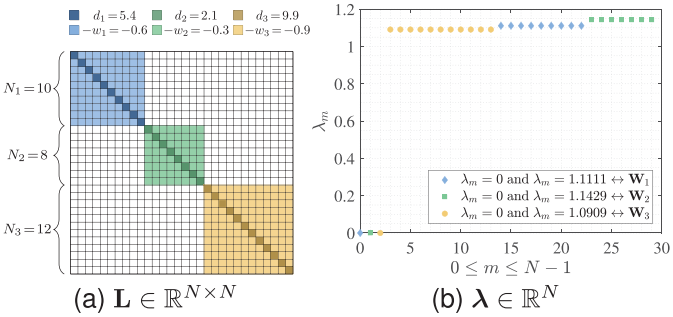


Fig. 2. Exemplary target Laplacian matrix and its eigenvalues ($\mathbf{n} = [10, 8, 12]^\top \in \mathbb{R}^K$, $N = 30$, $K = 3$).

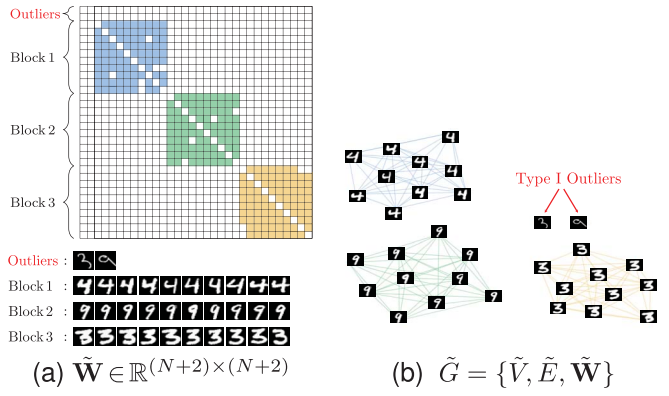


Fig. 3. Illustration of Type I outliers. The colored cells in the corrupted BD affinity matrix $\tilde{\mathbf{W}}$ represent non-zero edge weights in graph $\tilde{G}$.

Based on this definition, the corresponding ideal graph G includes only edges between vertices associated with the same block. In [9], using spectral analysis, we showed that if there exists a $\mathbf{W}$ as in Definition III.1, the eigenvalues of the associated Laplacian matrix $\mathbf{L} \in \mathbb{R}^{N \times N}$ will be of the following form

$$\boldsymbol{\lambda} = \text{sort}\left(\underbrace{0, ..., 0}_{K}, \underbrace{\frac{N_1}{N_1 - 1}, ..., \frac{N_1}{N_1 - 1}}_{N_1 - 1}, ..., \underbrace{\frac{N_K}{N_K - 1}, ..., \frac{N_K}{N_K - 1}}_{N_K - 1}\right), \quad (3)$$

where $\boldsymbol{\lambda} \in \mathbb{R}^N$ denotes the vector of target eigenvalues and $\text{sort}(.)$ is the sorting operation in ascending order.

Fig. 2 illustrates the vector of target eigenvalues $\boldsymbol{\lambda}$ associated with a Laplacian matrix of $K = 3$ blocks where each block is assumed to be concentrated around a constant $w_i \in \mathbb{R}^+$, e.g. $\mathbf{w} = [0.6, 0.3, 0.9]^\top \in \mathbb{R}^K$. Fig. 2(b) confirms the findings of [9], i.e., that the smallest eigenvalue is zero-valued and the remaining $N_i - 1$ number of eigenvalues are $\frac{N_i}{N_i - 1}$ for each block $i = 1, ..., K$.

For clustered data, the target block diagonal model in Definition III.1 represents the optimal level of sparsity with internally dense and externally disjoint groups of vertices. If the observed data would ideally follow this model, it would not contain outliers and the sparsity level could directly be deduced from the percentage of zero-valued entries in the affinity matrix. It is evident that setting additional entries in the affinity matrix in Definition III.1 to zero (resulting in an overly sparse graph) will reduce the internally dense structure of a cluster and lead to the occurrence of Type I outliers (see Definition III.2). In contrast, an overly dense graph, is obtained by adding undesired edges between blocks, which is consistent with the occurrence of Type II outliers (see Definition III.3) and its extreme case of group similarity (see Definition III.4). The introduced theoretical analysis in the following section describes the effect of these fundamental outlier types on the optimal level of sparsity and shows how optimizing the level of sparsity based on the determined target block diagonal model prevents these fundamental outlier effects and addresses robustness and sparsity jointly.

B. Outlier Effects on Target Eigenvalues

From Eq. (3), it follows that the non-zero components of the target eigenvalues contain the block size information. However, in practice, such a target vector is not available. Especially for outlier-corrupted affinity matrices, the blocks might be obscured (see also Fig. 8 for an example), which results, e.g., in a performance degradation of an eigenvalue-based block size estimate. To quantify this more precisely, and subsequently derive robust BDR methods, we next define some fundamental outlier types and analyze their effects on the target eigenvalues.

Definition III.2: (Type I Outliers, [31]). The feature vectors corresponding to the vertices that do not share edges with any of the samples are called Type I outliers.

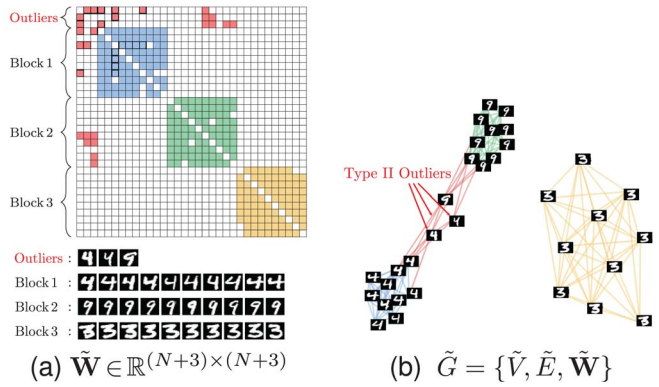
Definition III.2 is illustrated in Fig. 3 in which the unconnected vertices in $\tilde{G}$ are Type I outliers. Since the multiplicity of the zero-valued eigenvalues of $\mathbf{L}$ equals the number of connected components [48], this means that N_I Type I outliers lead to N_I additional zero-valued eigenvalues [31].

In real-world scenarios the number of Type I outliers varies and their occurrence, generally speaking, is affected by multiple factors: One significant delimiter for the number of Type I outliers is the data structure. For example, a simple similarity measure, i.e. $\mathbf{W} = \mathbf{X}^\top \mathbf{X}$ will produce a sparse affinity matrix only when the feature vectors are sparse. In practice, using images or medical observations as feature vectors usually generates non-sparse affinity matrices for a simple similarity measure (e.g. $\mathbf{W} = \mathbf{X}^\top \mathbf{X}$) while using, e.g. a term-document matrix as data matrix may result in a sparse matrix and consequently to the occurrence of Type I outliers. The second important delimiter is the affinity matrix construction. In more details, for a sparse affinity matrix construction method increasing sparsity produces Type I outliers. An example illustrating the link between Type I outliers and sparse affinity matrix construction is shown in Fig. 10 and in Appendix E.1 of the accompanying material [45] for the MNIST data base.

Next, we study the effect of Type II outliers, defined as follows:

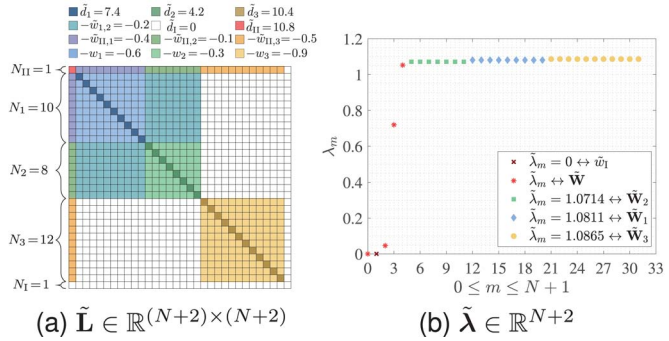
Definition III.3: (Type II Outliers, [31]). The feature vectors corresponding to the vertices that share edges with more than one group of feature vectors are called Type II outliers.

Definition III.3 is illustrated in Fig. 4, which shows that the connectedness of Type II outliers to multiple groups of feature vectors obscures the target group structure and poses a challenge to BDR methods.



(a) $\tilde{\mathbf{W}} \in \mathbb{R}^{(N+3)\times(N+3)}$ (b) $\tilde{G} = \{\tilde{V}, \tilde{E}, \tilde{\mathbf{W}}\}$

Fig. 4. Illustration of Type II outliers. The red colored cells in $\tilde{\mathbf{W}}$ correspond to edges of Type II outliers.



(a) $\tilde{\mathbf{L}} \in \mathbb{R}^{(N+2)\times(N+2)}$ (b) $\tilde{\boldsymbol{\lambda}} \in \mathbb{R}^{N+2}$

Fig. 5. Exemplary corrupted Laplacian matrix and its eigenvalues ($\mathbf{n} = [10, 8, 12]^\top \in \mathbb{R}^K$, $N = 30$, $K = 3$).

In contrast to Type I outliers studied in [31], the effect of Type II outliers on eigenvalues is still an open problem. Therefore, an analysis of the Type II outliers' effect on the eigenvalues of the Laplacian matrix is provided for the generalized eigen-decomposition in Eq. (2) as follows.[1]

Theorem 1: Let $\tilde{\mathbf{W}} \in \mathbb{R}^{(N+1)\times(N+1)}$ define a symmetric affinity matrix, that is equal to $\mathbf{W}$, except for an additional Type II outlier that shares similarity coefficients with K blocks where $\tilde{w}_{\mathrm{II},K} > 0$ denotes the similarity coefficient between the Type II outlier and the Kth block. Then, for the associated corrupted Laplacian matrix $\tilde{\mathbf{L}} \in \mathbb{R}^{(N+1)\times(N+1)}$ with eigenvalues $\tilde{\boldsymbol{\lambda}} \in \mathbb{R}^{N+1}$, it holds that

$$\begin{cases} N_1 - 1 \text{ elements of } \tilde{\boldsymbol{\lambda}} \text{ are equal to } \dfrac{N_1 w_1 + \tilde{w}_{\mathrm{II},1}}{\tilde{d}_1}, \\ N_2 - 1 \text{ elements of } \tilde{\boldsymbol{\lambda}} \text{ are equal to } \dfrac{N_2 w_2 + \tilde{w}_{\mathrm{II},2}}{\tilde{d}_2}, \\ \quad\vdots \\ N_K - 1 \text{ elements of } \tilde{\boldsymbol{\lambda}} \text{ are equal to } \dfrac{N_K w_K + \tilde{w}_{\mathrm{II},K}}{\tilde{d}_K}, \\ \text{the smallest element of } \tilde{\boldsymbol{\lambda}} \text{ is equal to zero,} \end{cases}$$

and the remaining K eigenvalues are the roots of

$$\prod_{j=1}^{K} (\tilde{w}_{\mathrm{II},j} - \tilde{\lambda}\tilde{d}_j) \left(- \sum_{j=1}^{K} \frac{N_j \tilde{w}_{\mathrm{II},j} \tilde{d}_j}{\tilde{w}_{\mathrm{II},j} - \tilde{\lambda}\tilde{d}_j} - \tilde{d}_{\mathrm{II}} \right) = 0,$$

where $\tilde{d}_{\mathrm{II}} = \sum_{j=1}^{K} N_j \tilde{w}_{\mathrm{II},j}$ and $\tilde{d}_j = (N_j - 1)w_j + \tilde{w}_{\mathrm{II},j}$.

Proof: See Appendix A.1 in [45]. □

We next introduce an extreme case of Type II outliers based on the following definition.

Definition III.4: (Group Similarity). If an entire group of vertices shares edges with another group of vertices we call this, group similarity.

The Laplacian matrix of Definition III.4 can be considered as a single connected component which means that the number of zero-valued eigenvalues equals to one [48]. In contrast to this simple interpretation, the remaining eigenvalues can be formulated as a function of intra-blocks and inter-blocks similarity coefficients where inter-blocks similarity coefficients are generally smaller-valued than those of intra-blocks in real-world scenarios. To provide a mathematical understanding of this, the following theorem quantifies the effect of group similarity on the target eigenvalues.

Theorem 2: Let $\tilde{\mathbf{W}} \in \mathbb{R}^{N\times N}$ define an affinity matrix that is equal to $\mathbf{W}$, except that block i has similarity with the remaining $K-1$ blocks with $\tilde{w}_{i,j} = \tilde{w}_{j,i} > 0$ denoting the value around which the similarity coefficients between blocks i and j are concentrated for $j = 1, ..., K$ and $i \neq j$. Then, the eigenvalues $\tilde{\boldsymbol{\lambda}} \in \mathbb{R}^N$ of $\tilde{\mathbf{L}} \in \mathbb{R}^{N\times N}$ are as follows:

$$\begin{cases} N_i - 1 \text{ elements of } \tilde{\boldsymbol{\lambda}} \text{ are equal to } \dfrac{N_i w_i + \sum\limits_{\substack{j=1, \\ j\neq i}}^{K} N_j \tilde{w}_{i,j}}{\tilde{d}_i}, \\ N_j - 1 \text{ elements of } \tilde{\boldsymbol{\lambda}} \text{ are equal to } \dfrac{N_j w_j + N_i \tilde{w}_{i,j}}{\tilde{d}_j}, \\ \quad\vdots \\ N_K - 1 \text{ elements of } \tilde{\boldsymbol{\lambda}} \text{ are equal to } \dfrac{N_K w_K + N_i \tilde{w}_{i,K}}{\tilde{d}_K}, \\ \text{the smallest element of } \tilde{\boldsymbol{\lambda}} \text{ is equal to zero,} \end{cases}$$

and the remaining $K-1$ eigenvalues in $\tilde{\boldsymbol{\lambda}}$ are the roots of

$$\prod_{\substack{j=1 \\ j\neq i}}^{K} (N_i \tilde{w}_{i,j} - \tilde{\lambda}\tilde{d}_j) \left(- \sum_{\substack{j=1 \\ j\neq i}}^{K} \frac{\tilde{d}_j N_j \tilde{w}_{i,j}}{N_i \tilde{w}_{i,j} - \tilde{\lambda}\tilde{d}_j} - \tilde{d}_i \right) = 0,$$

where $\tilde{d}_j = (N_j - 1)w_j + N_i \tilde{w}_{i,j}$, $\tilde{d}_i = (N_i - 1)w_i + \sum_{\substack{j=1 \\ j\neq i}}^{K} N_j \tilde{w}_{i,j}$.

Proof: See Appendix A.2 in [45]. □

A Laplacian matrix $\tilde{\mathbf{L}}$ that is corrupted with all discussed outlier types is displayed in Fig. 5(a), while the above derived outlier effects on the eigenvalues are visually summarized in Fig. 5(b).

Remark 1: To derive the theoretical results, simplifying assumptions[2], such as, concentration of the similarity coefficients within a block around a mean value are required. In practice,

[1] For an analysis based on the standard eigen-decomposition in Eq. (1), see Appendix B in [45].

[2] For our further analysis about loosening assumptions based on eigenvectors, see Theorem 1 in [49].

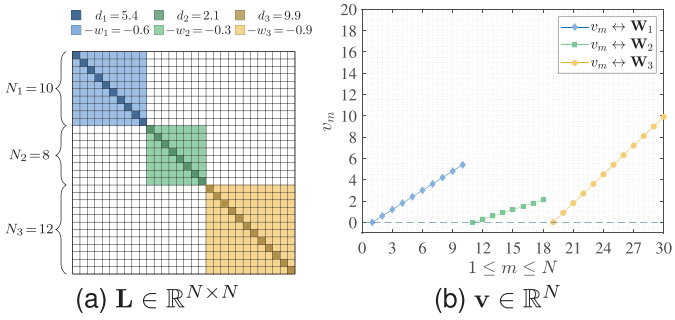


Fig. 6. Exemplary target Laplacian matrix and corresponding vector $\mathbf{v}$ ($\mathbf{n} = [10, 8, 12]^\top \in \mathbb{R}^K$, $N = 30$, $K = 3$).

these assumptions may only be approximately fulfilled. However, will see later in Section VI that the numerical performance gain obtained by suppressing outliers' effects outweigh the model mismatch in the considered benchmark data sets. Further, existing theoretical works on the spectrum of BDR methods, e.g. [50], [51], [52], [53], make even stricter assumptions, because the spectrum of an adjacency matrix can be found in closed form for simple models [51]. According to the most recent generic analyses [50] and [51], the spectrum of a BDR has been computed for planted partition model (PPM) with equal community sizes which is a special case of stochastic block model (SBM) assuming that the probability of having an edge within the cluster is constant and equal for all clusters while probability of having an edge to a different cluster is also constant and the same for all clusters. There are also some other researches on eigenvalues to determine the limiting distribution of the edge eigenvalues [52] and that of the outlier eigenvalues [53]. Even though these previously introduced spectral properties of random matrices are interesting to understand the complex graph structures, the available information about the spectrum of a BD matrix is limited to the considered simple models, i.e. PPM, and the available information about the eigenvalues is limited to eigenvalues of non-weighted graphs.

IV. Simplified Laplacian Matrix Analysis and Outlier Effects

In the preceding sections, outlier effects have been analyzed for $N \times N$ Laplacian matrices, which may lead to computationally heavy methods for large graphs. In this section, we therefore re-formulate the problem in $N \times 1$ vector space. In particular, assuming that $\mathbf{W}$ is symmetric and BD[3], the analysis is simplified by defining the vector $\mathbf{v} \in \mathbb{R}^N$ as follows [9]

$$v_m = \sum_{n=m}^{N} l_{m,n}, \tag{4}$$

where v_m and $l_{m,n}$, respectively, denote the mth and (m,n)th components of $\mathbf{v}$ and $\mathbf{L}$.

A. Target Vector $\mathbf{v}$ for Graph-Based Clustering

In [9], we have shown that the target vector $\mathbf{v}$ is a piece-wise linear function of the following form.

[3]A sparse matrix can be transformed into a BD form using the Reverse Cuthill-McKee (RCM) algorithm [54].

Definition IV.1: (Target Vector $\mathbf{v}$, [9]). The target vector $\mathbf{v}$ is a piece-wise linear function of the following form

$$v_m = f(m) = \begin{cases} (m-\ell_1)w_1 & \text{if} \quad \ell_1 \leq m \leq u_1 \\ \vdots & \\ (m-\ell_K)w_K & \text{if} \quad \ell_K \leq m \leq u_K, \end{cases}$$

where $\ell_1 = 1$, $u_1 = N_1$, $\ell_i = \sum_{k=1}^{i-1} N_k + 1$ and $u_i = \sum_{k=1}^{i} N_k$ for $i = 2, ..., K$.

An illustration is provided in Fig. 6 for a $K = 3$ block Laplacian matrix. As can be seen, the changepoints of the piecewise linear function provide information about the block size. To arrive at robust methods, we next determine the outlier effects on $\mathbf{v}$.

B. Outlier Effects on Target Vector $\mathbf{v}$

For a Type I outlier-corrupted affinity matrix $\tilde{\mathbf{W}} \in \mathbb{R}^{(N+1)\times(N+1)}$ that is identical to $\mathbf{W}$, except for a single Type I outlier $\mathrm{o_I}$, the overall edge weight associated with $\mathrm{o_I}$ is zero-valued, i.e. $\tilde{d}_{\mathrm{I}} = 0$. Based on Def. IV.1, it is straightforward to show that the component in the associated corrupted vector $\tilde{\mathbf{v}} \in \mathbb{R}^{N+1}$ that is associated with Type I outliers is zero valued, i.e., $\tilde{v}_{\mathrm{I}} = 0$. The Type II outliers' effect on $\mathbf{v}$ is as follows.

Theorem 3: Let $\tilde{\mathbf{W}} \in \mathbb{R}^{(N+1)\times(N+1)}$ define a Type II outlier-corrupted BD affinity matrix that is identical to $\mathbf{W} \in \mathbb{R}^{N\times N}$ except for a single Type II outlier that has non-zero similarity coefficients with all blocks. Assuming that the similarity coefficients associated with the outlier $\mathrm{o_{II}}$ and the blocks $j \in \{1, ..., K\}$ are concentrated around $\tilde{w}_{\mathrm{II},j}$, the components, whose indexes are valued between the outlier index and the largest index of the jth block, such that $m_{\mathrm{II}} < m \leq u_j$, increase by $\tilde{w}_{\mathrm{II},j}$ in the corrupted vector $\tilde{\mathbf{v}} \in \mathbb{R}^{N+1}$. Further, the component associated with the Type II outlier is given by

$$\tilde{v}_{\mathrm{II}} = \begin{cases} 0 & \text{if} \quad 0 < m_{\mathrm{II}} < \ell_1 \\ (m_{\mathrm{II}} - \ell_1)\tilde{w}_{\mathrm{II},1} & \text{if} \quad \ell_1 < m_{\mathrm{II}} < \ell_2 \\ \vdots & \\ \sum_{j=1}^{K-1} N_j \tilde{w}_{\mathrm{II},j} + (m_{\mathrm{II}} - \ell_K)\tilde{w}_{\mathrm{II},K} & \text{if} \quad \ell_K < m_{\mathrm{II}} \leq N+1 \end{cases},$$

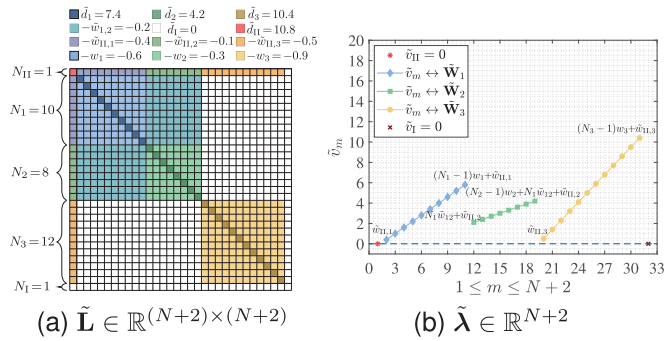
where ℓ_j denotes the lowest index of the jth block.

Proof: See Appendix C.1 in [45]. □

The effect of group similarity on $\mathbf{v}$ is as follows.

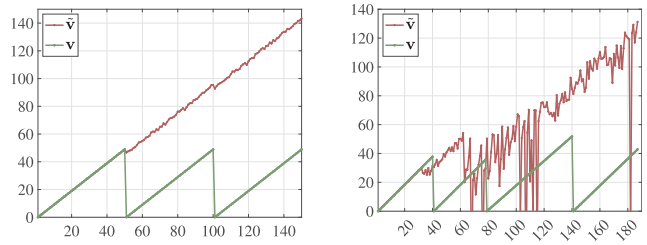
Theorem 4: Let $\tilde{\mathbf{W}} \in \mathbb{R}^{N\times N}$ define a corrupted affinity matrix that is identical to $\mathbf{W} \in \mathbb{R}^{N\times N}$, except that block i has non-zero similarity coefficients with the remaining $K-1$ blocks with $\tilde{w}_{i,j} = \tilde{w}_{j,i} > 0$ denoting the similarity coefficients around which, blocks i and j are concentrated. These similarities result in an increase by $N_i \tilde{w}_{i,j}$ in the components associated with the blocks $j = i+1, ..., K$ of $\tilde{\mathbf{v}} \in \mathbb{R}^N$ while the components of $j < i$ remain the same. Further, the components associated with block i remain the same for $i = 1$ and increase by $\sum_{j=1}^{i-1} N_j \tilde{w}_{i,j}$ for $2 \leq i \leq K$.

Proof: See Appendix C.2 in [45]. □



(a) $\tilde{\mathbf{L}} \in \mathbb{R}^{(N+2)\times(N+2)}$ (b) $\tilde{\boldsymbol{\lambda}} \in \mathbb{R}^{N+2}$

Fig. 7. Exemplary corrupted Laplacian matrix and corresponding $\tilde{\mathbf{v}}$ ($\mathbf{n} = [10, 8, 12]^\top \in \mathbb{R}^K$, $N = 30$, $K = 3$).



(a) Iris (b) Person Identification

Fig. 8. Exemplary deviations from the target vector $\mathbf{v}$. The affinity matrix is defined by $\mathbf{W} = \mathbf{X}^\top \mathbf{X}$.

In the sequel, we analyze the worst case of group similarity, i.e., similarity of all blocks. Note that, in this case, eigenvalues can not be formulated as a function of similarity coefficients due to the impossibility of simplifying determinants of full matrices via Gaussian elimination. However, recovering the structure of $\mathbf{W}$ based on $\mathbf{v}$ is possible based on the following result.

Corollary 4.1: Let $\tilde{\mathbf{W}} \in \mathbb{R}^{N\times N}$ define a corrupted affinity matrix that is identical to $\mathbf{W} \in \mathbb{R}^{N\times N}$, except that each block $i = 1, ..., K$ has non-zero similarity coefficients with the remaining $K-1$ blocks with $\tilde{w}_{i,j} = \tilde{w}_{j,i} > 0$ denoting the similarity coefficients around which, blocks i and j are concentrated for $j = 1, ..., K$ and $i \neq j$. This leads to a piece-wise linear function given by

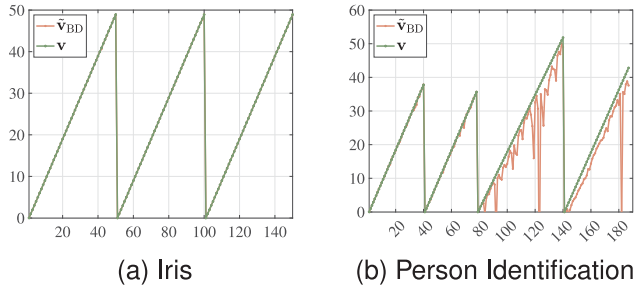
$$\tilde{v}_m = \begin{cases} (m-\ell_1)w_1 & \text{if } \ell_1 \leq m \leq u_1 \\ (u_1-\ell_1+1)\tilde{w}_{1,2} + (m-\ell_2)w_2 & \text{if } \ell_2 \leq m \leq u_2 \\ \vdots & \\ \sum_{i=1}^{K-1}(u_i-\ell_i+1)\tilde{w}_{i,K} + (m-\ell_K)w_K & \text{if } \ell_K \leq m \leq u_K \end{cases}$$

where $\ell_1 = 1, u_1 = N_1, \ell_i = \sum_{k=1}^{i-1} N_k + 1$ and $u_i = \sum_{k=1}^{i} N_k$ for $i = 2, ..., K$.

Proof: See Appendix C.2 in [45]. □

An exemplary corrupted Laplacian matrix $\tilde{\mathbf{L}}$ and corresponding $\tilde{\mathbf{v}}$ illustrating our theoretical findings are shown in Fig. 7(a) and 7(b), respectively. Consistent with Section III-B, outliers of Type I result in zeros in $\tilde{\mathbf{v}}$. Additionally, Type II outliers and group similarity lead to an increase in the target vector $\mathbf{v}$ as quantified in Theorems 3 and 4, respectively.

To demonstrate the degree of model mismatch in the considered real-world data sets due to the simplifying assumptions



(a) Iris (b) Person Identification

Fig. 9. Exemplary deviations from the target vector $\mathbf{v}$. The BD affinity matrix is defined by removing the undesired similarity coefficients between different blocks of $\mathbf{W} = \mathbf{X}^\top \mathbf{X}$.

that were necessary to derive our theory, exemplary vectors associated with the corrupted affinity matrices that are subject to group similarity as in Corollary 4.1 and the corresponding target vector $\mathbf{v}$'s are illustrated in Figs. 8(a) and 8(b), respectively, for the Iris [55] and Person Identification [56] data sets. As can be seen, undesired similarity coefficients between different blocks result in shifts from the target piece-wise linear functions starting from the second linear pieces, consistent with our theory in Corollary 4.1. In particular, assumptions and findings of Corollary 4.1 hold well in real-world data sets, especially, when the data sets include densely connected clusters of points, e.g. Ceramic [57] and Iris [55].[4] Additionally, corrupted data sets, e.g. Person Identification [56] whose corresponding affinity matrix is subject to Type I outliers and group similarity results in large deviations from the target piece-wise linear function with group similarity shifts and small-valued $\tilde{\mathbf{v}}$ components corresponding to Type I outliers as it has been theoretically shown in previous. A further analysis illustrating the degree of model mismatch between the target BD model and a BD model with varying similarity coefficients within the blocks is shown in Fig. 9(a) and 9(b), respectively, for the Iris and Person Identification data sets. Even though highly corrupted data sets generate large deviations from the assumed models in real-world scenarios, an appropriate BDR suppresses these outlier effects by providing an optimal level of sparsity which is a major motivation of our proposed algorithm that will be detailed in the sequel.

V. The Proposed Method

In Section V-B, we briefly discuss the key ideas of the proposed method. Following this, a step-by-step detailed mathematical explanation is provided in Section V-C. We then analyze the computational complexity in Section V-D. Additionally, a comprehensive visual summary is provided in Appendix F.1 of the accompanying material [45] and a pseudo-code algorithm of FRS-BDR is given in Algorithm 2.

A. Problem Statement: Jointly Addressing Robustness and Sparsity

With the results of Sections III and IV in place, we are ready to understand the relationship between the level of sparsity and

[4]For further real-world data examples, see Appendix E.2 in [45].

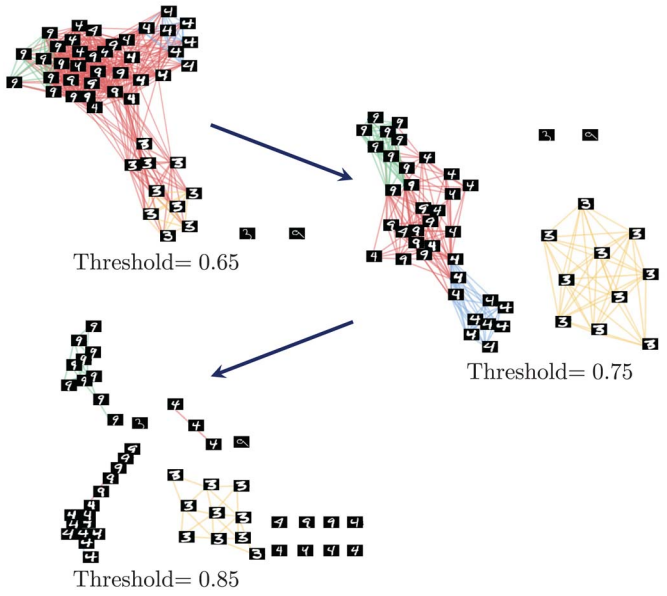


Fig. 10. Example graphs for increasing sparsity. An initial affinity matrix is defined by $\mathbf{W} = \mathbf{X}^\top \mathbf{X}$ and the example graphs are obtained by removing the edges whose corresponding edge weight is smaller than the defined threshold value.

the previously defined outlier types to highlight the importance of jointly addressing robustness and sparsity. In a generic example, Fig. 10 shows that a dense graph (top) contains high amounts of group similarity while increasing sparsity reduces the number of Type II outliers (middle). Finally, further increasing sparsity generates Type I outliers until at some point the underlying true cluster structure is completely lost. This means that an inaccurate determination of the sparsity level leads to the above discussed outlier effects for existing approaches, such as, e.g. [21], [22], [23]. In this section, we therefore propose a new method that addresses robustness and sparsity *jointly*.

More precisely, let a given data set of feature vectors $\mathbf{X} \in \mathbb{R}^{M\times N}$ be represented as a weighted graph $G = \{V, E, \mathbf{W}\}$, i.e., $\mathbf{W} = \mathbf{X}^\top \mathbf{X}$ and $\|\mathbf{x}_m\| = 1$, $m = 1, ..., N$. Further, let $\mathbf{D}$ and $\mathbf{L} \in \mathbb{R}^{N\times N}$ denote, respectively, the overall edge weight and the Laplacian matrices associated with $\mathbf{W}$. Then, the goal of this work is to robustly estimate a K block zero-diagonal symmetric affinity matrix $\mathbf{W} \in \mathbb{R}^{N\times N}$ using the available information about the vector $\mathbf{v}$ and an eigen-decomposition. The number of blocks K is assumed to be unknown and $\mathbf{X}$ may be subject to heavy-tailed noise and outliers which results in undesired effects, such as group similarity. The number of outliers is assumed to be unknown. Computational efficiency is also of fundamental interest. *Thus, in brief, the overall aim is to develop a fast and sparsity aware BDR method that is robust against outliers and group similarity.*

B. Main Ideas and Outline of Proposed Method

This section summarizes the main ideas of our proposed *Fast and Robust Sparsity-Aware Block Diagonal Representation (FRS-BDR)* method. The full details of each step are given in Section V-C.

To provide a general understanding, a high-level flow diagram illustrating the key steps of FRS-BDR is provided in Fig. 11. As shown in the figure, the method consists of two general steps, i.e., enhancing BD structure (Step 1) and estimating vector $\mathbf{v}$ (Step 2). The computation step starts with a given Type I outlier-corrupted and non-sparse Laplacian matrix $\mathbf{L}$ (Step 1.0: Initialization in Fig. 11). According to the explicit Definition III.2 of Type I outliers, the method first removes the similarity coefficients associated with Type I outliers, which are represented in red color, from $\mathbf{L}$ (Step 1.1: Type I Outlier Removal in Fig. 11). Then, the next step is to structure the resulting matrix $\dot{\mathbf{L}}$ in a BD form $\ddot{\mathbf{L}}$ with a similarity-based BD ordering that we present in the sequel (Step 1.2: Similarity-based Block Diagonal Ordering in Fig. 11). The last part of Step 1 is, to obtain vector $\mathbf{v}$ in form of K discrete linear segments by computing an ordered sparse Laplacian matrix $\ddot{\mathbf{L}}$ (Step 1.3: Sparsity for Excessive Group Similarity in Fig. 11). Then, the estimation step starts with a changepoint detection that we propose, to compute the possible block sizes (Step 2.1: Compute Candidate Block Sizes in Fig. 11). For each possible block size vector, i.e., $\mathbf{n}_r = [8, 10, 12]^\top \in \mathbb{Z}_+^K$ in this illustrating example, the method computes a target vector $\mathbf{v}^{(r)}$ and a corresponding estimate $\hat{\mathbf{v}}^{(r)}$ as a function of the estimated target similarity coefficients (Step 2.2.1: Estimate Target Similarity Coefficients in Fig. 11). Further, for every undesired similarity coefficient around which the blocks are concentrated, the shifted vectors (see Corollary 4.1) are computed separately and the undesired similarity coefficients are estimated (Step 2.2.2: Estimate Undesired Similarity Coefficients in Fig. 11). Finally, the estimate $\hat{\mathbf{v}} \in \mathbb{R}^{N-N_\mathrm{I}}$ is computed for the block size vector which provides the best fit to the computed vector $\ddot{\mathbf{v}}$.

C. FRS-BDR Algorithm

1) Step 1: Enhancing BD Structure: The key requirement for computing vector $\mathbf{v}$ based on Eq. (4) is recovering an approximately BD structured Laplacian matrix. Assuming that $\mathbf{W}$ (and the associated $\mathbf{L}$) are symmetric and sparse matrices, they can be ordered in a BD form [54] based on which vector $\mathbf{v}$ can be directly computed. However, in general, similarity measures may not produce sparse affinity matrices. We therefore discuss the most challenging scenario, i.e., that $\mathbf{W}$ is subject to Type I outliers and all blocks exhibit similarity. Considering the Type I outliers' effect on the target vector $\mathbf{v}$ (see Section IV-B), the proposed vector $\mathbf{v}$ computation starts with Type I outlier detection (Step 1.1). Then, a new BD ordering based on the similarity coefficients is proposed to generate a BD ordered Laplacian matrix (Step 1.2). Lastly, a sparse Laplacian matrix design is detailed for the case of excessive group similarity (Step 1.3).

(a) Step 1.1: Type I outlier removal: Type I outliers are detected according to

$$\mathbf{x}_m \in \mathbf{O}_\mathrm{I} \text{ if } w_{m,n} = 0 \text{ for } \forall n = 1, ..., N \text{ and } m \neq n, \quad (5)$$

where $\mathbf{O}_\mathrm{I} \in \mathbb{R}^{M\times N_\mathrm{I}}$ denotes the matrix of Type I outliers, $\mathbf{x}_m \in \mathbb{R}^M$ is the mth feature vector for $m = 1, ..., N$, $w_{m,n}$ is the m, nth similarity coefficient corresponding to $\mathbf{x}_m$ (due to the symmetry of $\mathbf{W}$, $w_{m,n} = w_{n,m}$).

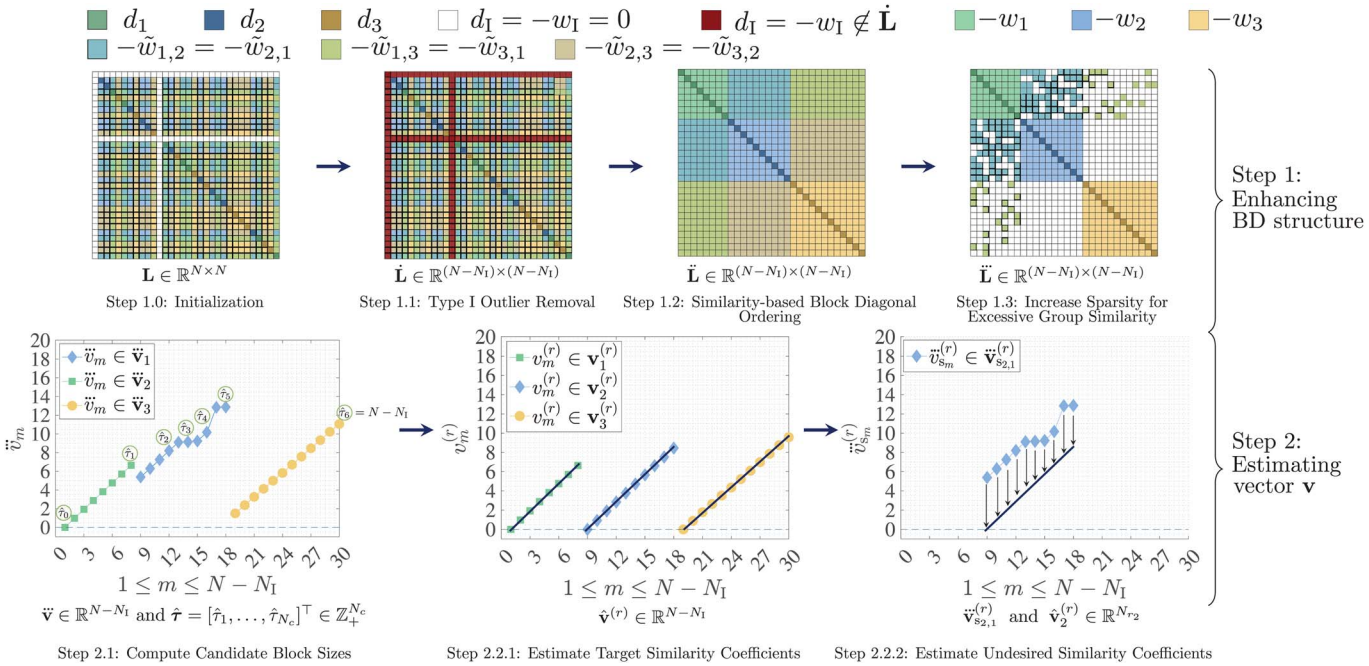


Fig. 11. High-level flow diagram illustrating the key steps of FRS-BDR using a generic example with $K = 3$ clusters.

Type I outlier removal based on Eq. (5) directly follows Definition III.2 which means that the operation does not require a determination of the number of outliers. It is an important preliminary step since the presence of Type I outliers may lead to an inaccurate sparsity increase in Step 1.3 due to their effects on the eigenvalues or an incorrect candidate block size estimation in Step 2.1 based on their effects on the vector $\mathbf{v}$.

(b) Step 1.2: Similarity-based BD ordering (sBDO): Let $\dot{\mathbf{X}} \in \mathbb{R}^{M\times(N-N_\mathrm{I})}$, $\dot{\mathbf{W}}, \dot{\mathbf{D}}$ and $\dot{\mathbf{L}} \in \mathbb{R}^{(N-N_\mathrm{I})\times(N-N_\mathrm{I})}$ be the resulting matrices after Step 1.1. The vector of the BD order, i.e., $\hat{\mathbf{b}} \in \mathbb{Z}_+^{N-N_\mathrm{I}}$ is determined based on the following steps.

Step 1.2.1: Initialization: The BD order vector $\hat{\mathbf{b}}^{(1)}$ is comprised of the node index of maximum overall edge weight (i.e., $\dot{d}_{\max}$).

Step 1.2.2: Adding the most similar neighbor to $\hat{\mathbf{b}}^{(s)}$*:* Let $\hat{\mathbf{b}}^{(s)} = [\hat{b}_1, ..., \hat{b}_{s-1}]^\top \in \mathbb{Z}_+^{s-1}$, with $s = 2, ..., N - N_\mathrm{I}$, denote the BD order vector at the sth stage. Assuming that the neighbors set is non-empty[5], the most similar neighbor to $\hat{\mathbf{b}}^{(s)}$ at the sth stage is determined by

$$\hat{b}_s = \underset{m\in\{1,2,...,N-N_\mathrm{I}\}}{\arg\max} \left\{ \sum_{n=1}^{s-1} \dot{w}_{m,\hat{b}_n} \right\}, \tag{6}$$

where $m \in \mathbb{Z}_+$ such that $1 \le m \le N - N_\mathrm{I}$ denotes a neighbor node.

An example of the sBDO algorithm is illustrated in Fig. 12 and technically summarized in Algorithm 1. As can be seen from Fig. 12, starting from node five, whose overall edge weight is largest valued, the method selects the neighbors based on their edge weights that represent the similarity to previously selected nodes. After selecting all neighbors, the method jumps to the node that has the maximum overall edge weight among the remaining nodes and determines the ordering of the associated neighbors.

[5]If it is empty the method simply stacks the node index of maximum overall edge weight into $\hat{\mathbf{b}}^{(s)}$.

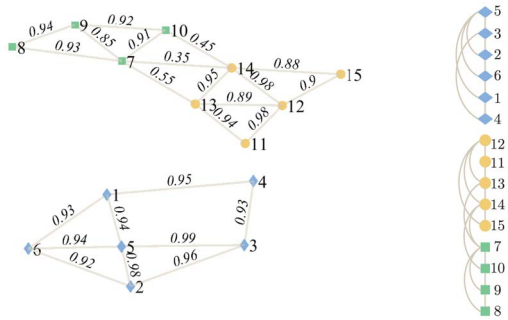


Fig. 12. Exemplary plot of the sBDO algorithm.

Algorithm 1: sBDO

Input: $\dot{\mathbf{W}}, \dot{\mathbf{D}} \in \mathbb{R}^{(N-N_\mathrm{I})\times(N-N_\mathrm{I})}$
Initialization:
Find the node of maximum overall edge weight $\dot{d}_{\max}$
for $s = 2, ..., (N - N_\mathrm{I})$ **do**
  Adding the most similar neighbor to $\hat{\mathbf{b}}^{(s)}$*:*
  if *at least one neighbor exists* **then**
    Estimate $\hat{b}_s$ using Eq. (6) and stack into $\hat{\mathbf{b}}^{(s)}$
  else
    Find the node with maximum overall edge weight among unselected nodes and stack $\hat{b}_s$ into $\hat{\mathbf{b}}^{(m)}$
  end
end
Output: Estimated order vector $\hat{\mathbf{b}}^{(s)} \in \mathbb{Z}_+^{(N-N_\mathrm{I})}$

Different from the Reverse Cuthill-McKee (RCM) [54], which is a well-known block diagonal ordering method, the proposed sBDO algorithm incorporates useful information from the similarity coefficients. By doing this, the sBDO ordering

method does not require making specific assumptions[6]. on the similarity coefficients or a sparse matrix structure that is necessary in RCM algorithm. In challenging scenarios, for example, starting the ordering with a Type II outlier the sBDO algorithm continues selecting vertices from the most similar cluster therewith quickly suppressing the effect of Type II outlier's similarity coefficients.[7]

(c) Step 1.3: Increase sparsity for excessive group similarity: Let $\ddot{\mathbf{W}}$, $\ddot{\mathbf{D}}$ and $\ddot{\mathbf{L}} \in \mathbb{R}^{(N-N_{\mathrm{I}})\times(N-N_{\mathrm{I}})}$ be the matrices resulting from Step 1.2. A sparsity improved Laplacian matrix $\ddot{\mathbf{L}} \in \mathbb{R}^{(N-N_{\mathrm{I}})\times(N-N_{\mathrm{I}})}$ is designed[8] by increasing sparsity as long as, at least, the two smallest eigenvalues are close to zero[9]. After computing $\ddot{\mathbf{L}}$, the vector $\ddot{\mathbf{v}} \in \mathbb{R}^{N-N_{\mathrm{I}}}$ is obtained using Eq. (4)[10].

Increasing sparsity for excessive group similarity is an optional step for the FRS-BDR algorithm. In particular, it is designed to obtain a sparsity improved Laplacian matrix $\ddot{\mathbf{L}}$ whose associated vector $\ddot{\mathbf{v}}$ provides distinct changepoints that can be easily computed in Step 2.1. In this way, the negative impact of excessive group similarity, which obscures the piece-wise linear functions (for details, see Corollary 4.1), is suppressed and changepoints become more visible. However, this operation does not enforce a block diagonal affinity matrix since it eliminates only a small portion of the undesired similarity coefficients. Therefore, the following steps estimate the vector $\mathbf{v}$ as a function of desired similarity coefficients and that of undesired that will be removed to obtain a BDR.

2) Step 2: Estimating Vector $\mathbf{v}$*:* This step models $\ddot{\mathbf{v}}$ as a K-piece linear function of similarity coefficients around which the blocks are assumed to be concentrated (for details, see Corollary 4.1.), i.e.,

$$\ddot{\mathbf{v}}_i = \mathbf{v}_i + \mathbf{1}\sum_{\substack{j=1\\ j\neq i}}^{i-1} N_j \tilde{w}_{i,j}, i = 1, ..., K, \tag{7}$$

where

$$\mathbf{v}_i = [0, w_i, ..., (N_i - 1)w_i]^\top \in \mathbb{R}^{N_i} \tag{8}$$

denotes the ith linear segment of the target vector $\mathbf{v}_i$, w_i is the similarity coefficient around which the block i is concentrated and $\tilde{w}_{i,j}$ is the undesired similarity coefficient between blocks i and j around which they are concentrated, $\mathbf{1} \in \mathbb{R}^{N_i}$ is the column vector of ones, N_i and N_j are, respectively, the size of block i and j.

(a) Step 2.1: Computing candidate block sizes: Let $N_c \in \mathbb{Z}_+$ denote the number of changepoints, let $\boldsymbol{\tau} = [\tau_1, \tau_2, ..., \tau_{N_c}]^\top \in \mathbb{Z}_+^{N_c}$ be the vector containing the corresponding locations in $\ddot{\mathbf{v}}$, and let $\tau_0 = 0$ and $\tau_{N_c+1} = N$.

[6]For examples of similarity coefficients' empirical distributions, see Appendix E.3 in [45].

[7]For the analysis of sBDO performance, see Appendix F.4.5 in [45].

[8]For the exemplary sparse Laplacian matrix design algorithms, see Appendix F.3 in [45].

[9]For the definition of *close to zero*, see Appendix F.2 in [45]

[10]The vector $\mathbf{v}$ can alternatively be computed using $\ddot{\mathbf{L}} \in \mathbb{R}^{(N-N_{\mathrm{I}})\times(N-N_{\mathrm{I}})}$ after executing Steps 1.1 and 1.2 if, at least, the two smallest eigenvalues of $\ddot{\mathbf{L}}$ are close to zero.

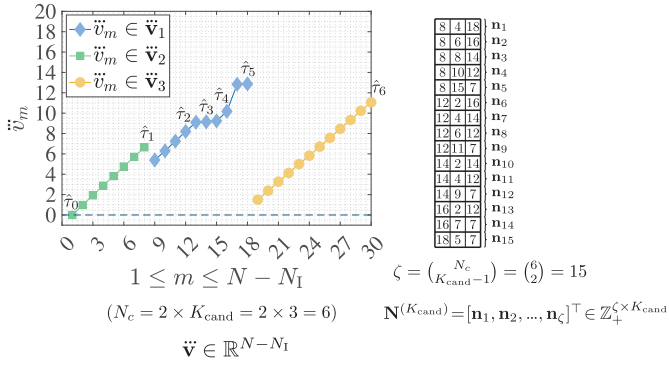


Fig. 13. Exemplary plot of computing candidate block sizes.

Then, to estimate the model for vector $\ddot{\mathbf{v}}$ based on Eq. (7), our first step is to detect the changepoints in $\ddot{\mathbf{v}}$ by minimizing the following penalized least-squares function as in [58]

$$\sum_{i=1}^{N_c+1} \sum_{m=\tau_{i-1}+1}^{\tau_i} (\ddot{v}_m - \hat{v}_m)^2 + \gamma N_c, \tag{9}$$

where $\ddot{v}_m$ and $\hat{v}_m$ denote, respectively, the mth point in the ith linear segment of $\ddot{\mathbf{v}}$ and the corresponding least-squares linear fit. γ is the penalty parameter that controls the number of changepoints N_c. In particular, Eq. (9) considers all possible changepoints for $\gamma = 0$ and it rejects including additional changepoints if the residual error is smaller than the determined penalty parameter γ. Different from determining γ directly, this step increases the value of γ gradually as long as the function finds a lower number of changepoints than a predefined maximum number of changepoints $N_{c_{\max}} \in \mathbb{Z}_+$ which is a reasonably small number satisfying $K - 1 \le N_{c_{\max}}$. Then, for a candidate number of blocks from a given vector, i.e., $K_{\mathrm{cand}} \in [K_{\min}, ..., K_{\max}]^\top \in \mathbb{Z}_+^{N_K}$, the resulting number of changepoints N_c and corresponding locations $\boldsymbol{\tau}$ in Eq. (9) are used to compute the candidate size vectors $\mathbf{n}_r = [N_{r_1}, N_{r_2}, ..., N_{r_{K_{\mathrm{cand}}}}]^\top \in \mathbb{Z}_+^{K_{\mathrm{cand}}}$, $r = 1, ..., \zeta$ that are designed by combination of all possible size vectors with $\zeta = \binom{N_c}{K_{\mathrm{cand}}-1}$. Lastly, the block-size matrix associated with a candidate number of blocks, i.e.,

$$\mathbf{N}^{(K_{\mathrm{cand}})} = [\mathbf{n}_1, \mathbf{n}_2, ..., \mathbf{n}_\zeta]^\top \in \mathbb{Z}_+^{\zeta \times K_{\mathrm{cand}}}, \tag{10}$$

is formed.[11]

The computation of candidate block sizes illustrated in Fig. 13 for a candidate block number $K_{\mathrm{cand}} = 3$. After estimating the changepoints using Eq. (9), a possible block size matrix, i.e. $\mathbf{N}^{(K_{\mathrm{cand}})} \in \mathbb{Z}_+^{\zeta \times K_{\mathrm{cand}}}$, with $\zeta = 15$ is computed for all possible block size combinations.

In this step, the changepoint locations are determined based on a piece-wise linear fit of the vector $\ddot{\mathbf{v}}$ using Eq. (9). This is a fundamental step to compute the candidate block sizes. However, the obtained information from Eq. (9) does not provide the target and undesired similarity coefficients which are needed to structure the affinity matrix in a block diagonal form. In other

[11]In practice, the candidate size vectors including the block sizes that are smaller than a predefined minimum number of nodes in the blocks $N_{\min}$ can be removed from $\mathbf{N}^{(K_{\mathrm{cand}})}$.

words, the estimated piece-wise linear fit is a combination of these similarity coefficients as it has been illustrated in Fig. 7(b). Therefore, Step 2.2, i.e. estimating the target and undesired similarity coefficients individually is a necessary step to obtain information about all similarity coefficients. A more detailed explanation of similarity coefficients' estimation is provided in the following.

(b) Step 2.2: Estimating matrix of similarity coefficients:

b.1) Step 2.2.1: Estimate Target Similarity Coefficients Suppose that N_{r_i} denotes the size of the ith linear segment from a candidate size vector $\mathbf{n}_r$, as defined in Eq. (10). Further, let $\mathbf{v}^{(r)} \in \mathbb{R}^{(N-N_1)}$ denote the target vector $\mathbf{v}$ associated with $\mathbf{n}_r$ defined by

$$v_m^{(r)} = \sum_{n=m}^{u_{r_i}} \ddot{l}_{m,n} \quad \text{s.t.} \quad \begin{matrix} \ell_{r_i} \le m \le u_{r_i} \\ i = 1, ..., K_{\text{cand}} \end{matrix}, \tag{11}$$

where the mth and (m,n)th components of $\mathbf{v}^{(r)}$ and $\ddot{\mathbf{L}}$ are denoted, respectively, by $v_m^{(r)}$ and $\ddot{l}_{m,n}$, $\ell_{r_1} = 1$, $u_{r_1} = N_{r_1}$, $\ell_{r_i} = \sum_{k=1}^{i-1} N_{r_k} + 1$ and $u_{r_i} = \sum_{k=1}^{i} N_{r_k}$ for $i = 2, ..., K_{\text{cand}}$.

After computing $\mathbf{v}^{(r)}$ using Eq. (11), with Definition IV.1, we model it as a K-piece linear function of the target similarity coefficients. The model parameters are estimated in the FRS-BDR algorithm by applying the algorithm from [59] that determines a plane-based piece-wise linear fit. In more details, for every linear segment $i = 1, ..., K_{\text{cand}}$ associated with K_{cand}, the method first estimates the parameters of the linear fit. Then, it estimates the target similarity coefficients $w_i, ..., w_{K_{\text{cand}}}$ based on the slope of piece-wise linear fit estimates. A step-by-step detailed description of the plane-based piece-wise linear fit algorithm to determine $\mathbf{v}^{(r)}$ is given in Section IX.A of the supplementary material.

b.2) Step 2.2.2: Estimate Undesired Similarity Coefficients In this step, the shifted vectors of $\mathbf{v}^{(r)}$ are computed as follows

$$\ddot{\mathbf{v}}_{s_{i,j}}^{(r)} = \mathbf{v}_i^{(r)} + \ddot{\mathbf{v}}_{i,j}^{(r)}, \quad \begin{matrix} i = 2, ..., K_{\text{cand}}, \\ j = 1, ..., i-1 \end{matrix} \tag{12}$$

where $\ddot{\mathbf{v}}_{i,j}^{(r)} \in \mathbb{R}^{N_{r_i}}$ denotes the vector of increase, associated with the undesired group similarity between block i and j, and $\ddot{\mathbf{v}}_{s_{i,j}}^{(r)}$ is the associated shifted target vector.[12] Then, combining the results from Eq. (7), Eq. (11) and Eq. (12), the undesired similarity coefficients between different blocks can be estimated as

$$\hat{w}_{i,j}^{(r)} = \frac{\text{med}(\ddot{\mathbf{v}}_{s_{i,j}}^{(r)} - \hat{\mathbf{v}}_i^{(r)})}{N_{r_j}} \quad \begin{matrix} i = 2, ..., K_{\text{cand}} \\ j = 1, ..., i-1 \end{matrix}, \tag{13}$$

where $\text{med}(.)$ denotes the median operator, N_{r_j} is defined in Eq. (10), and $\hat{w}_{i,j}^{(r)}$ is the undesired similarity coefficient estimate between i and j.

Remark 2: Alternative to using the median operator as an estimator in Eq. (13), one could consider using the sample mean estimator based on the theory in Sections III and IV. However, for the sample mean, a single outlying component has an unbounded effect on estimating undesired similarity coefficient, while the median operator provide robustness with the highest possible breakdown value of 50% (for a detailed discussion about robustness comparisons, see Section 1.3 in [29]). This property of the median provides robustness even in real-world cases where our theoretical assumptions are not fully fulfilled.

[12]For details, see Section IX.B of the supplementary material.

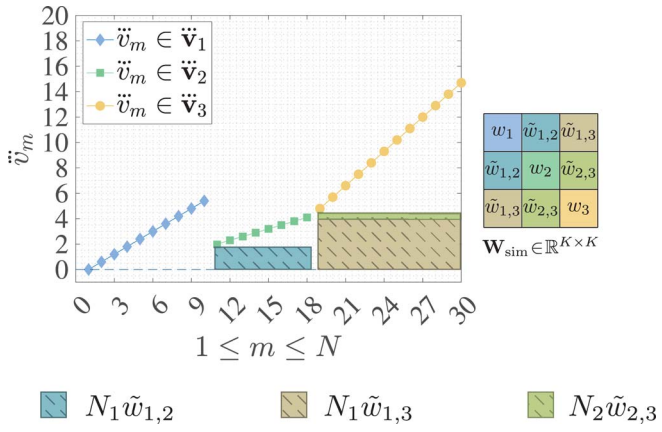


Fig. 14. Exemplary plot of $\ddot{\mathbf{v}}$ and $\mathbf{W}_{\text{sim}}$ with $K_{\text{cand}} = K$, $\mathbf{n} = [10, 8, 12]^\top \in \mathbb{R}^K$, $\text{diag}(\mathbf{W}_{\text{sim}}) = [0.6, 0.3, 0.9]^\top \in \mathbb{R}^K$, $\tilde{w}_{1,2} = 0.2$, $\tilde{w}_{1,3} = 0.4$, and $\tilde{w}_{2,3} = 0.1$.

To clarify Steps 2.2.1 and 2.2.2, an example with $K_{\text{cand}} = K$ illustrating the computation of vector $\ddot{\mathbf{v}}$ and a matrix $\mathbf{W}_{\text{sim}} \in \mathbb{R}^{K \times K}$ is shown in Fig. 14. As can be seen, the target similarity coefficients, which are the diagonal elements of $\mathbf{W}_{\text{sim}}$, i.e., $\text{diag}(\mathbf{W}_{\text{sim}}) = [w_1, w_2, ..., w_K]^\top \in \mathbb{R}^K$, represent an estimate of the slopes of the $K_{\text{cand}} = K$ linear segments in $\ddot{\mathbf{v}}$. Further, off-diagonal elements of $\mathbf{W}_{\text{sim}}$ represent undesired similarity coefficients between different blocks and are calculated by computing the undesired shifts that have been highlighted as shaded areas in Fig. 14.

b.3) Step 2.3: Estimating vector $\ddot{\mathbf{v}}$ and $\mathbf{W}_{\text{sim}}$ From the computed estimates $\hat{\mathbf{W}}_{\text{sim}}^{(r)} \in \mathbb{R}^{K_{\text{cand}} \times K_{\text{cand}}}$ and $\hat{\mathbf{v}}^{(r)} \in \mathbb{R}^{(N-N_1)}$, the vector $\hat{\ddot{\mathbf{v}}}_i^{(r)}$ is computed by plugging in the associated intermediate estimates for all $r = 1, ..., \zeta$ and $K_{\text{cand}} = K_{\min}, ..., K_{\max}$ into Eq. (7) and determining the final estimate as

$$\hat{\ddot{\mathbf{v}}} = \underset{\mathbf{n}_r \in \mathbf{N}^{(K_{\text{cand}})}}{\text{argmin}} \|\ddot{\mathbf{v}} - \hat{\ddot{\mathbf{v}}}^{(r)}\|_2 \tag{14}$$

where $\forall \hat{w}_i^{(r)} \in \text{diag}(\hat{\mathbf{W}}_{\text{sim}}^{(r)})$, $\hat{w}_i^{(r)} > \hat{w}_{i,j}^{(r)}$ holds for $i = 1, ..., K_{\text{cand}}$, $j = 1, ..., K_{\text{cand}}$ and $i \neq j$.

Since the target block diagonal model with internally dense externally disjoint clusters represents the optimum level of sparsity, the closeness of the estimate of $\ddot{\mathbf{v}}$ to the target piece-wise linear function directly provides information of how well the algorithm was able to remove the undesired edges and therewith determine the sparsity level. In particular, the estimate of vector $\ddot{\mathbf{v}}$ provides fundamental information about the number of blocks, the number of elements for every block, desired and undesired similarity coefficients associated with each block that have been collected in the matrix $\mathbf{W}_{\text{sim}}$. To design a BDR that provides a good balance with internally dense externally sparse clusters, the desired similarity coefficients, the proposed strategy preserves the similarity coefficients corresponding to

the diagonal entries of $\mathbf{W}_{\text{sim}}$ while removing that of undesired similarity coefficients corresponding to the off-diagonal entries of $\mathbf{W}_{\text{sim}}$.[13]

The proposed FRS-BDR is summarized in Algorithm 2. The codes are provided at: https://github.com/A-Tastan/FRS-BDR.

D. Computational Analysis of FRS-BDR

A comprehensive computational analysis is computed in Section X of the supplementary material by determining the number of fladd, flmlt, fldiv and flam. The Landau's big O symbol is used for the cases when the complexity is not specified as above operations. For a detailed information, see [60], [61]. Our analysis showed that the complexity of FRS-BDR strongly depends on the initial structure of the affinity matrix and the number of blocks K. In addition to the numeric analysis, the complexity is analyzed experimentally in the following sections.

VI. Experimental Results

This section benchmarks the proposed FRS-BDR method in a broad range of real data experiments, including cluster enumeration and handwritten digit, object and face clustering.

Data sets: The performance is analyzed using the well-known data sets for handwritten digit clustering [32], [62], for object clustering [63], for face clustering [64], [65], [66] and for cluster enumeration [67], [68], [69], [70], [71]. The detailed information about the data sets is given in the following sections.

Baselines: For the task of subspace clustering, FRS-BDR is benchmarked against seven state-of-the-art BDR approaches [6], [7], [8], [9], [10], two low-rank representation methods [17], [18], a sparse representation method (SSC) [21], a robust principal component analysis method (FRPCAG) [72], a robust spectral clustering method (RSC) [73] and the initial affinity matrix that is defined by $\mathbf{W} = \mathbf{X}^\top \mathbf{X}$. For cluster enumeration[14], the method is benchmarked against seven popular community detection methods, i.e. [74], [75], [76], [77], [78], [79] and our previously proposed method that is called SPARCODE in [24].

Parameter setting: In all experiments, the parameters are optimally tuned for the competitor approaches, while FRS-BDR is computed with the default parameters that are detailed in Section XI of the supplementary material.

Evaluation metrics: The computation time (t) and average clustering accuracy $(\bar{c}_{\text{acc}})$ are used for the subspace clustering performance analysis. In cluster enumeration, the empirical probability of detection (p_{det}), modularity (mod) and conductance (cond) are used in addition to t. The evaluation metrics are comprehensively explained in Section XI of the supplementary material.

A. Handwritten Digit Clustering

The effectiveness of FRS-BDR in handwritten digit clustering is shown based on the following popular real-world data sets:

[13]For examples that analyze the mismatch between the target and estimated BD structure, see Appendix E.2 in [45].

[14]For the numerical cluster enumeration results, see Appendix F.4.4.2 of the accompanying material [45].

Algorithm 2: FRS-BDR

Input: $\mathbf{X}\in\mathbb{R}^{M\times N}$, $K_{\min}$, $K_{\max}$, $N_{c_{\max}}$, $N_{\min}$(opt.)
Compute $\mathbf{W}\in\mathbb{R}^{N\times N}$ i.e. $\mathbf{W}=\mathbf{X}^\top\mathbf{X}$ for $\forall\mathbf{x}_m\in\mathbf{X}$, $\|\mathbf{x}_m\|$=1
Step 1: Enhancing BD Structure
Step 1.1: Type I Outlier Removal
Compute $\dot{\mathbf{W}}$, $\dot{\mathbf{D}}$ and $\dot{\mathbf{L}}\in\mathbb{R}^{(N-N_{\text{I}})\times(N-N_{\text{I}})}$ via Eq. (5)
Step 1.2: Similarity-based Block Diagonal Ordering
Perform Algorithm 1 to achieve $\hat{\mathbf{b}}^{(s)}\in\mathbb{Z}_+^{(N-N_{\text{I}})}$
Obtain $\ddot{\mathbf{W}}$, $\ddot{\mathbf{D}}$ and $\ddot{\mathbf{L}}\in\mathbb{R}^{(N-N_{\text{I}})\times(N-N_{\text{I}})}$ using $\hat{\mathbf{b}}^{(s)}$
Step 1.3 (opt.): Sparsity for Excessive Group Similarity
Design $\ddot{\mathbf{L}}\in\mathbb{R}^{(N-N_{\text{I}})\times(N-N_{\text{I}})}$ for the desired method,
i.e. Algorithm 3 or 4 in [45]
Compute $\ddot{\mathbf{v}}\in\mathbb{R}^{(N-N_{\text{I}})\times 1}$ corresponding to $\ddot{\mathbf{L}}$ using Eq. (4)
(or alternatively $\ddot{\mathbf{v}}\in\mathbb{R}^{(N-N_{\text{I}})\times 1}$ corresponding to $\ddot{\mathbf{L}}$)
Step 2: Estimating Vector v
for $K_{\text{cand}}=K_{\min},\ldots,K_{\max}$ **do**
  Step 2.1: Computing Candidate Block Sizes
  Compute $\mathbf{N}^{(K_{\text{cand}})}\in\mathbb{Z}_+^{\zeta\times K_{\text{cand}}}$ using Eqs. (9)-(10)
  Step 2.2: Estimating $\mathbf{W}_{\text{sim}}$
  for $\mathbf{n}_r=\mathbf{n}_1,\ldots,\mathbf{n}_\zeta$ **do**
    Step 2.2.1: Estimating Target Similarity Coefficients
    Compute $\mathbf{v}^{(r)}\in\mathbb{R}^{(N-N_{\text{I}})}$ using Eq. (11)
    for $i=1,\ldots,K_{\text{cand}}$ **do**
      Calculate $\boldsymbol{\Sigma}_i^{(r)}\in\mathbb{R}^{2\times 2}$ and $\boldsymbol{\mu}_i^{(r)}\in\mathbb{R}^2$ for $\boldsymbol{\Upsilon}_i^{(r)}$
      Find $\hat{\boldsymbol{\vartheta}}_i^{(r)}\in\mathbb{R}^2$ and $\hat{\text{b}}_i^{(r)}\in\mathbb{R}$
      Find $\hat{\mathbf{v}}_i^{(r)}\in\mathbb{R}^{N_{ri}}$ and compute $\hat{w}_i$
    end
    Form $\text{diag}(\hat{\mathbf{W}}_{\text{sim}}^{(r)})=[\hat{w}_1^{(r)},\hat{w}_2^{(r)},\ldots,\hat{w}_{K_{\text{cand}}}^{(r)}]^\top\in\mathbb{R}^{K_{\text{cand}}}$
    and $\hat{\mathbf{v}}^{(r)}=[(\hat{\mathbf{v}}_1^{(r)})^\top,(\hat{\mathbf{v}}_2^{(r)})^\top,\ldots,(\hat{\mathbf{v}}_{K_{\text{cand}}}^{(r)})^\top]^\top\in\mathbb{R}^{(N-N_{\text{I}})}$
    Step 2.2.2: Estimating Undesired Similarity Coefficients
    for $i=2,\ldots,K_{\text{cand}}$ **do**
      for $j=1,\ldots,i-1$ **do**
        Compute $\ddot{\mathbf{v}}_{\text{s}_{i,j}}^{(r)}\in\mathbb{R}^{(N-N_{\text{I}})}$ using Eqs. (12)
        Compute $\hat{w}_{i,j}^{(r)}$ using Eq. (13) and stack $\hat{\mathbf{W}}_{\text{sim}}^{(r)}$
      end
    end
    Estimate $\ddot{\mathbf{v}}^{(r)}$ using Eq. (7)
    Update $\hat{\ddot{\mathbf{v}}}$ based on Eq. (14)
  end
end
Output: $\hat{\ddot{\mathbf{v}}}$, $\hat{\mathbf{W}}_{\text{sim}}$, $\hat{\mathbf{n}}$

MNIST data set: The data base includes 60,000 training and 10,000 test images corresponding to 10 digits. For a varying number of subjects $K=\{2,3,5,8,10\}$, the data matrix $\mathbf{X}$ is generated using 100 randomly selected images from the test set for every subject where the images are used as feature vectors and normalized. As in [7], $\mathbf{X}$ of size $784\times 100K$ is produced for the images of size 28×28.

USPS data set: 7291 training and 2007 test images of size 16×16 are contained in the data set. The data matrix $\mathbf{X}$ is computed by following the same procedure, except for using 50 randomly selected images from the test set for every subject. As a result, for the images of size 16×16, the data matrix $\mathbf{X}$ of size $256\times 50K$ corresponding to a number of subjects $K=\{2,3,5,8,10\}$, is obtained.

In contrast to object and face applications that we will detail in the following sections, the data matrix $\mathbf{X}$ of high dimensional feature vectors is directly used in initial affinity matrix design. The initial affinity matrix, i.e. $\mathbf{W}=\mathbf{X}^\top\mathbf{X}$ is used as

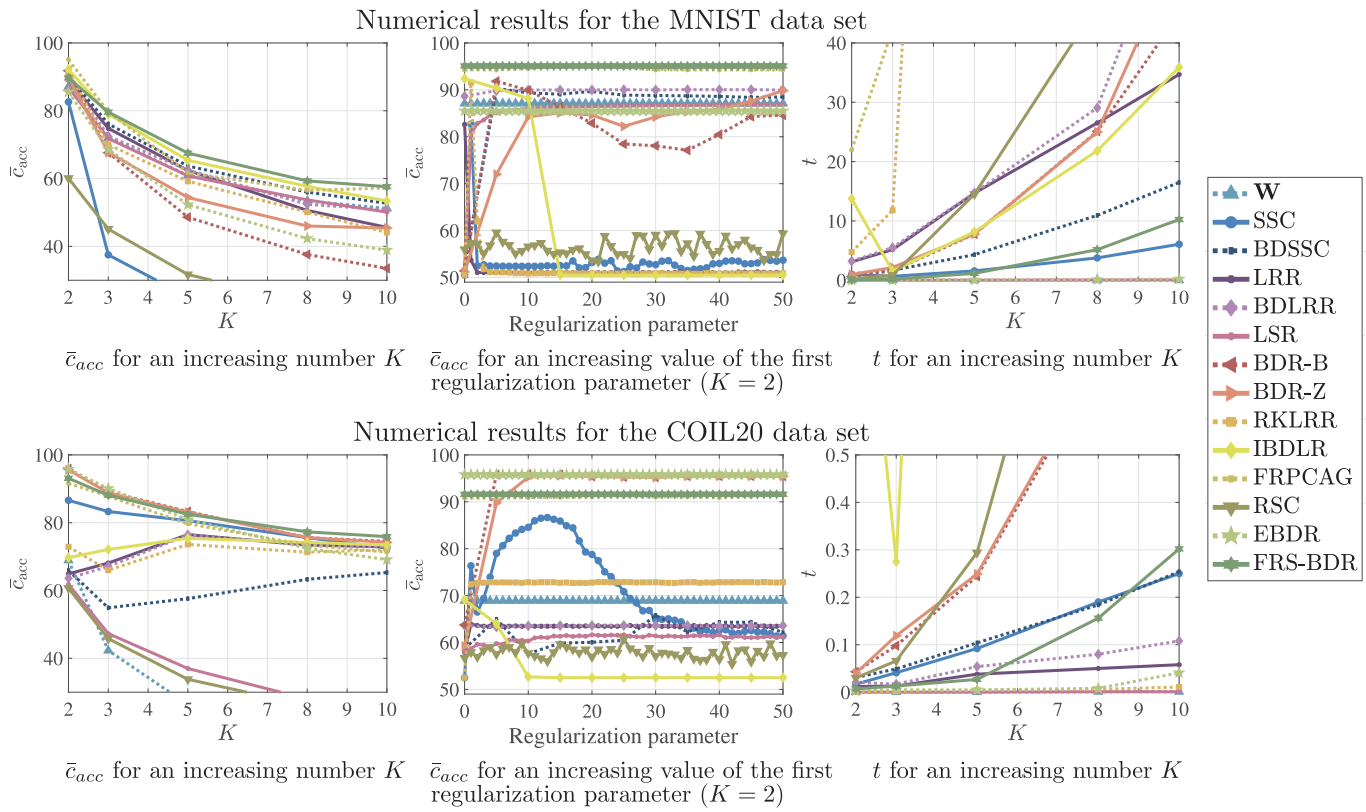


Fig. 15. Numerical results for the COIL20 and MNIST data sets. The regularization parameters of the competing methods are tuned for optimal performance in all settings while the proposed method determines the parameters using Algorithms 1 and 2. In the regularization parameter performance analysis, for all competing methods that use two parameters, the second one is tuned optimally while varying the first parameter.

an input to BDR approaches [6], [7], [8], [9], [10], low-rank representation methods [17], [18] and the sparse representation method in [21] to design affinity matrices in a desired form. Then, spectral clustering[15] is applied to the resulting affinity matrices of different methods. Different from affinity matrix construction methods, FRPCAG [72] and RSC [73] algorithms use the data matrix $\mathbf{X}$ as an input. These methods determine the affinity matrices based on their default construction where the number of neighbors is defined by gradually decreasing the number of all neighbors until the methods do not fail.

An example of digit clustering results is shown in Fig. 15 for the MNIST data base. A broad set of analyses including MNIST and USPS data bases is provided in Appendix F.4.1 of the accompanying material [45]. Even though the performance of SSC [21], BDSSC [6], BDLRR [6], BDR-B [7], BDR-Z [7], IBDLR [8], LSR [10], LRR [17], RKLRR [18], FRPCAG [72] and RSC [73] is reported for an optimal tuning of the parameters, which is not feasible in practice, the FRS-BDR achieves the highest clustering accuracy results in almost all cases. Further, the regularization parameter effect analysis in Fig. 15 shows that BDR-B and BDR-Z performances are sensitive to the choice of the first regularization parameter, even when tuning the second one optimally. Based on the computation time analysis, the main drawback of competitor approaches is that they are sensitive to the dimension of the feature space whereas FRS-BDR is an efficient algorithm for the data sets including high dimensional feature vectors.

To quantify the performance of different BDR approaches in terms of the sparsity, an additional set of experiments analyzing modularity (mod) and conductance (cond) scores, which are the commonly used quality metrics for this analysis, are introduced in Appendix F.4 of the accompanying material [45]. The numerical analysis demonstrates that the proposed FRS-BDR algorithm provides a "good balance" in sparsity with large-valued modularity scores and small-valued conductance scores in most of the cases.[16] The analysis confirms the results of the clustering accuracy performance analysis that FRS-BDR algorithm shows a good performance compared to the optimally tuned BDR approaches while providing considerably better performance than optimally tuned low-rank representation methods. Different from structuring all clusters based on a single determined sparsity parameter (which may be difficult to tune in practice), our approach allows for treating every block differently, depending on the occurrence of the outliers' effect within each block and this makes the proposed FRS-BDR method advantageous in terms of balancing the sparsity.

[15]For the details about spectral clustering, see Section XI of the supplementary material.

[16]The modularity and conductance performance of the proposed FRS-BDR algorithm could be further improved by enforcing the estimated blocks to be distinct but such a step may result in a performance degradation in clustering accuracy which is more important in these clustering applications.

B. Object Clustering

This section introduces a set of experiments that are performed on the COIL20 [63] data base of 20 objects. In COIL20, each object has 72 images where the images are taken by rotating the object on a turntable in five degree intervals. In our experiments, the processed COIL20 data set in [80] containing images of size 32×32 pixels is used. Then, the data set $\mathbf{X}$ of size 1024×400 is generated by selecting 20 images randomly for every object. The feature space is reduced to 10 based on the PCA performance, which is provided in Appendix F.4.2.1 of the accompanying material [45].

As in [7], a performance analysis of every application is conducted for an increasing value of K, i.e., $K = \{2, 3, 5, 8, 10\}$ using 100 randomly selected subject combinations. To obtain the affinity matrices for the competing methods, the regularization parameters are manually tuned on a grid of 50 values. Finally, spectral clustering [48] is applied and the results in Fig. 15, for an increasing value of K, are obtained analogously to [7] (see Appendix F.4.2 in [45] for further details). The average clustering accuracy $\bar{c}_{\text{acc}}$ results show that FRS-BDR performs best while EBDR is an efficient method for small values of K. In terms of t, the main competitors BDR-B and BDR-Z show poor performance whereas FRS-BDR performs relatively good even for large values of K. This computational advantage of the proposed method can be explained with its simple nature, i.e. finding a piece-wise linear function robustly, which is easy to solve in comparison to analyzing the graph structure in a matrix space as in the existing BDR methods.

The BDR-B and BDR-Z methods show poor performance for small-valued regularization parameters even though the second regularization parameter is optimally tuned. An important point is that these approaches reach their best results lately in comparison to experiments on face clustering data sets that are explained in the following section.

C. Face Clustering

In this section, the subspace clustering performances of different methods are benchmarked in terms of their $\bar{c}_{\text{acc}}$ and t by using the following application details:

ORL data set: The data set includes 10 images of 40 different subjects that are taken at different times by varying the lighting, facial expressions and details. As in [8], we resize all images to 32×32 to obtain a data matrix $\mathbf{X}$ of size 1024×400 using normalized features. The feature space dimension is reduced to nine using Principal Component Analysis (PCA) in order to reduce the computation time[17].

JAFFE data set: The JAFFE data set comprises 213 images of seven facial expressions from 10 Japanese female models. As in [8], the images are resized to 64×64 pixels and the data set $\mathbf{X}$ of size 4096×213 is computed using resized images as normalized feature vectors before applying PCA to reduce the dimensionality to 14 features[18].

Yale data set: 165 grayscale images of 15 different individuals. For every subject, the data set contains 11 images that capture different facial expressions. The data matrix $\mathbf{X}$ of size 1024×165 is constructed as in the ORL Data Set[19].

After determining the number of PCA features, the same procedure as in object clustering is performed and the performance is reported for a different number of subjects $K = \{2, 3, 5, 8, 10\}$ in Fig. 16. For a detailed performance analysis, see Appendix F.4.3 of the accompanying material [45].

The average clustering accuracy $\bar{c}_{\text{acc}}$ and computation time t for the ORL and the JAFFE data sets are provided in Fig. 16. Consistent with the previous experiments, FRS-BDR shows the best clustering accuracy performance among all approaches in almost all cases. In terms of t, FRS-BDR shows a reasonably good performance until the number of subjects reaches $K = 8$. A reduction for a large value of K can be obtained by adjusting $N_{c_{\max}}$. Extensive further numerical experiments are reported in Appendices E.5.3.1, E.5.3.2, and E.5.3.3. of [45].

D. Subspace Clustering on Well-Known Clustering Data Sets

This section investigates the subspace clustering performance of different approaches in terms of their average clustering accuracy using the following popular clustering data sets: Breast Cancer Wisconsin (Breast Cancer) [67], Chemical Composition of Ceramic (Ceramic) [57], Vertebral Column [68], Fisher's iris (Iris) [55], Radar-based Human Gait (Human Gait) [69], Ovarian Cancer [70], Person Identification [56] and Parkinson [71]. To analyze subspace clustering performances on popular clustering data sets, subspace clustering is first performed on the initial affinity matrix that is defined by $\mathbf{W} = \mathbf{X}^\top \mathbf{X}$. Analogous to the handwritten digit clustering application in Section VI-A, the data matrix is used as an input to the FRPCAG [72] and RSC [73] methods while state-of-the-art BDR methods use the initial affinity matrix that is defined by $\mathbf{W} = \mathbf{X}^\top \mathbf{X}$ as an input to design BD structured affinity matrices. Then, spectral clustering as detailed in Section XI of the supplementary material is performed on the BD affinity matrix estimates. For the FRPCAG [72] and RSC [73] methods, spectral clustering is performed based their eigenvector estimates. As in previous experiments, the competitor approaches' results are shown for optimally tuned parameters while the proposed FRS-BDR is performed with the default parameters.

The clustering accuracy performances of different block-diagonal representation approaches are detailed in terms of their average clustering accuracy in Table I. As can be seen from Table I, FRS-BDR provides a similar performance as the maximum clustering accuracy of its strongest competitors (BDR-B, BDR-Z, BD-LRR) while it outperforms all other block diagonal representation approaches. The method is also computationally efficient in comparison to most of the competitors based on the additional experiments that are given in Appendix F.4.4 of the accompanying material in [45].

[17]For the PCA analysis of the ORL data set, see Appendix F.4.3.1 of the accompanying material [45].

[18]For the PCA analysis of the JAFFE data set, see Appendix F.4.3.2 of the accompanying material [45].

[19]For the PCA analysis of the Yale data set, see Appendix F.4.3.3 of the accompanying material [45].

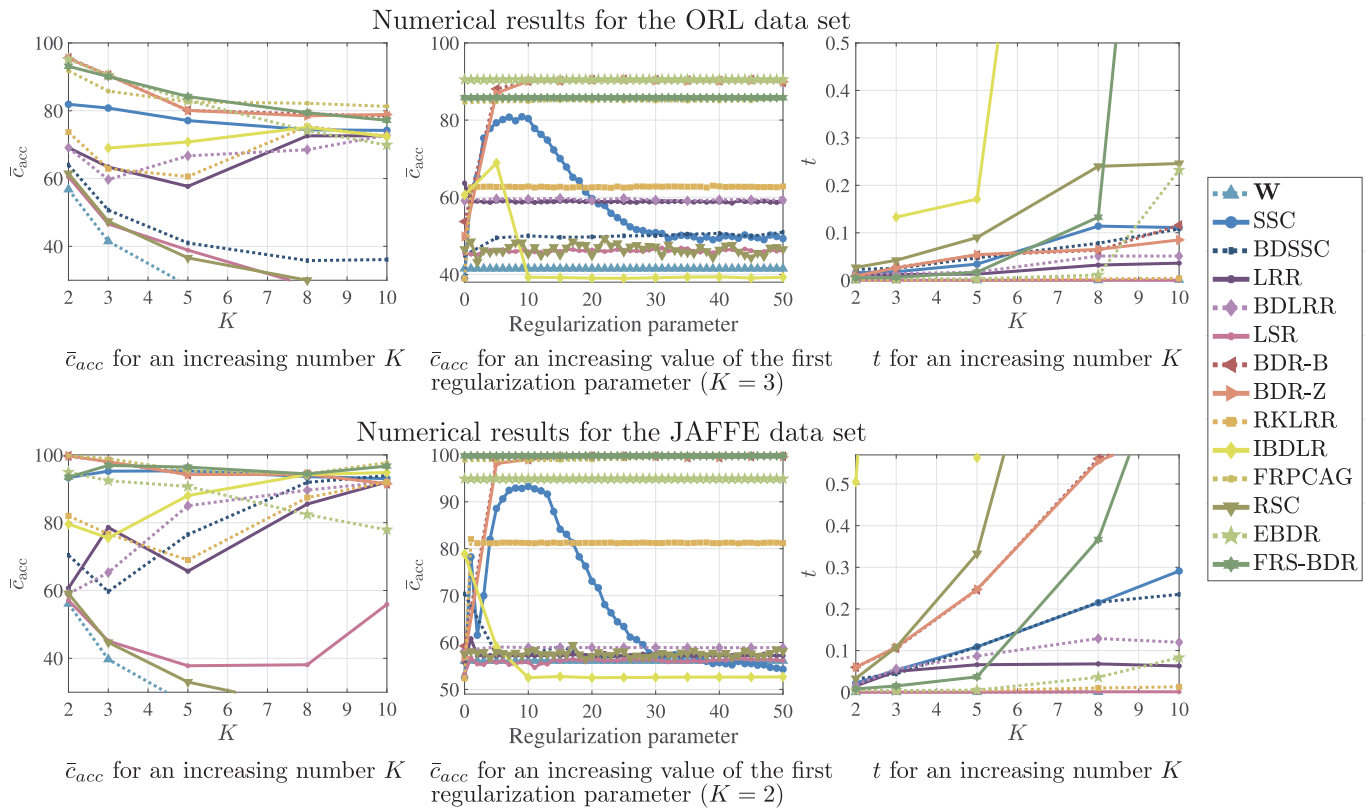


Fig. 16. Numerical results for the ORL and JAFFE data sets. The regularization parameters of the competing methods are tuned for optimal performance in all settings while the proposed method determines the parameters using Algorithms 1 and 2. In the regularization parameter performance analysis, for all competing methods that use two parameters, the second one is tuned optimally while varying the first parameter.

TABLE I

SUBSPACE CLUSTERING PERFORMANCE OF DIFFERENT BLOCK DIAGONAL REPRESENTATION APPROACHES ON WELL-KNOWN CLUSTERING DATA SETS. $\mathbf{W}$ REPRESENTS THE SUBSPACE CLUSTERING RESULTS THAT ARE OBTAINED BY USING THE INITIAL AFFINITY MATRIX $\mathbf{W} = \mathbf{X}^\top\mathbf{X}$ AS AN INPUT TO SPECTRAL CLUSTERING ALGORITHM. THE REMAINING COLUMNS SHOW AFFINITY MATRIX CONSTRUCTION METHODS THAT ARE USING $\mathbf{W}$ AS INPUT AND PERFORMING SPECTRAL CLUSTERING ON THE SPARSE AFFINITY MATRIX ESTIMATES. THE PERFORMANCES ARE SUMMARIZED IN TERMS OF $\bar{c}_{\text{acc}}$ FOR PARAMETER-FREE APPROACHES INCLUDING $\mathbf{W}$, EBDR AND FRS-BDR WHILE THE REMAINING METHODS ARE SHOWN FOR $c_{\text{accmin}} - c_{\text{accmax}}$. 'x' DENOTES THE FAILED RESULTS DUE TO THE COMPLEX-VALUED EIGENVECTORS

	Subspace Clustering Performances for Different Block Diagonal Representation Methods												
		Minimum-Maximum Clustering Accuracy ($c_{\text{accmin}} - c_{\text{accmax}}$) for Different Regularization Parameters											
Data Set	**W**	SSC	BD-SSC	LRR	BD-LRR	LSR	BDR-B	BDR-Z	RKLRR	IBDLR	FRPCAG	RSC	EBDR	**FRS-BDR**
Breast Cancer [67],	88.2	51.0-74.7	50.3-88.2	54.3-90.3	88.0-90.0	73.5-88.2	62.4-90.0	52.9-90.2	62.6-91.7	60.3-90.0	60.5-88.2	50.1-58.5	85.2	90.1
Ceramic [57],	98.9	51.1-98.9	51.1-100	95.5-98.9	95.5-98.9	54.5-98.9	51.1-100	51.1-98.9	51.1-95.5	51.1-98.9	50.0-100	50.0-69.3	98.9	98.9
Vertebral Column [68],	73.2	50.0-77.7	50.3-74.8	53.9-72.6	72.6-72.6	62.6-75.8	67.4-76.8	71.9-76.8	67.4-71.3	67.4-76.1	51.0-75.8	50.0-69.7	74.8	75.8
Iris [55],	78.0	34.7-82.7	34.0-83.3	38.7-80.7	80.0-98.0	78.0-82.7	34.0-96.7	65.3-96.7	34.0-80.0	34.7-84.0	34.0-89.3	35.3-50.0	98.0	96.7
Human Gait [69],	77.3	20.3-77.4	20.1-77.5	26.1-83.9	78.9-83.5	55.4-75.9	20.3-84.8	26.4-84.5	20.5-85.5	20.4-81.6	50.8-77.8	22.1-26.0	81.1	77.1
Ovarian Cancer [70],	61.7	51.4-73.6	50.9-71.3	52.3-76.4	54.2-76.4	51.9-66.2	53.7-75.9	51.9-74.1	55.6-88.4	55.6-75.5	77.8-89.3	50.0-69.4	77.8	77.3
Person Identification [56],	x	33.7-96.8	31.6-95.7	49.7-94.7	71.1-94.7	33.2-64.2	31.6-96.3	59.4-95.7	34.2-94.1	33.7-95.7	29.4-92.5	28.9-41.2	97.3	96.8
Parkinson [71],	61.3	50.4-58.8	50.0-61.3	50.4-54.2	50.4-61.3	57.9-61.3	50.4-61.3	50.0-61.3	50.4-61.7	50.4-61.3	50.0-67.5	50.4-72.1	56.7	58.2
Average	76.9	42.8-80.1	42.3-81.5	52.6-81.4	73.8-84.4	58.4-76.6	46.4-85.2	53.6-84.8	47.0-83.5	46.7-82.9	50.4-85.1	42.1-57.0	83.7	83.9

E. Robustness Analysis

A further analysis evaluating robustness of the proposed FRS-BDR method in noisy scenarios with corruptions in data/feature space, is reported in this section. To analyze robustness, against outliers, digit samples from MNIST [32] and USPS [62] data sets are corrupted with salt and pepper noise and Poisson noise. Object and face recognition data sets are not included to robustness analysis due to the performance degradation of the (non-robust) PCA that is part of the feature generation.

The robustness analysis results of different methods are shown in Fig. 17 for the MNIST and USPS data sets that are corrupted with salt and pepper noise for an increasing percentage of outlier contamination. As in previously analyzed scenarios, the proposed FRS-BDR shows relatively good performance compared to the optimally tuned approaches for both data sets. This is because the proposed method leverages the derived theory on how an ideal block diagonal structure is disturbed by outliers and this allows to precisely remove the effects of Type I and Type II outliers as well as the group similarity. Many block diagonal affinity matrix construction methods that we compare against are not robust against outliers and it is well-known that performance of non-robust methods can severely be degraded in presence of outliers [29].

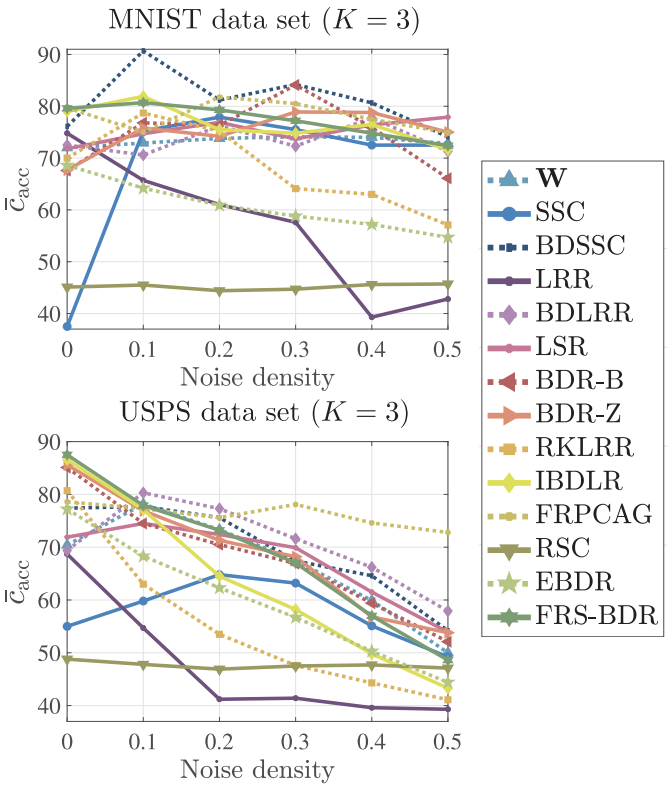


Fig. 17. Robustness analysis results for the MNIST and USPS data sets. The regularization parameters of the competing methods are tuned for optimal performance in all settings while the proposed method determines the parameters using Algorithms 1 and 2. $\bar{c}_{\text{acc}}$ performances are shown for increasing density value of the salt and pepper noise.

VII. Conclusion

A robust method to recover a block diagonal affinity matrix in challenging scenarios has been presented. The proposed Fast and Robust Sparsity-Aware Block Diagonal Representation (FRS-BDR) method jointly estimates cluster memberships and the number of blocks. It builds upon our presented theoretical results that describe the effect of different fundamental outlier types in cluster analysis, allowing a reformulation of the problem as a robust piece-wise linear fitting problem. Comprehensive experiments, including a variety of real-world applications demonstrate the effectiveness of FRS-BDR compared to optimally tuned benchmark methods in terms of clustering accuracy, computation time and cluster enumeration performance. Since all codes are made available, the FRS-BDR method can also easily be benchmarked on other larger-scale data sets, e.g. [81], [82], [83].

References

[1] Z. Kong and X. Yang, "Color image and multispectral image denoising using block diagonal representation," *IEEE Trans. Image Process.*, vol. 28, no. 9, pp. 4247–4259, Sep. 2019.

[2] Y. Dar, A. M. Bruckstein, M. Elad, and R. Giryes, "Postprocessing of compressed images via sequential denoising," *IEEE Trans. Image Process.*, vol. 25, no. 7, pp. 3044–3058, Jul. 2016.

[3] Z. Zhang, Y. Xu, L. Shao, and J. Yang, "Discriminative block diagonal representation learning for image recognition," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 7, pp. 3111–3125, Jul. 2018.

[4] C.-G. Li, Z. Lin, H. Zhang, and J. Guo, "Learning semi-supervised representation towards a unified optimization framework for semi-supervised learning," in *Proc. IEEE Conf. Comp. Vision*, 2015, pp. 2767–2775.

[5] Y. Qin, H. Wu, X. Zhang, and G. Feng, "Semi-supervised structured subspace learning for multi-view clustering," *IEEE Trans. Image Process.*, vol. 31, pp. 1–14, 2021.

[6] J. Feng, Z. Lin, H. Xu, and S. Yan, "Robust subspace segmentation with block diagonal prior," in *Proc. IEEE Conf. Comp. Vision Pattern Recognit.*, 2014, pp. 3818–3825.

[7] 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.

[8] X. Xie, X. Guo, G. Liu, and J. Wang, "Implicit block diagonal low-rank representation," *IEEE Trans. Image Process.*, vol. 27, no. 1, pp. 477–489, Jan. 2018.

[9] A. Taştan, M. Muma, and A. M. Zoubir, "Eigenvalue-based block diagonal representation and application to p-nearest neighbor graphs," in *Proc. 30th Eur. Signal Process. Conf.*, 2022, pp. 1761–1765.

[10] C.-Y Lu, H. Min, Z.-Q. Zhao, L. Zhu, D.-S. Huang, and S. Yan, "Robust and efficient subspace segmentation via least squares regression," in *Proc. Eur. Conf. Comp. Vision*, 2012, pp. 347–360.

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

[12] F. Wu, Y. Hu, J. Gao, Y. Sun, and Yin. B, "Ordered subspace clustering with block diagonal priors," *IEEE Trans. Cybern.*, vol. 46, no. 12, pp. 3209–3219, Dec. 2016.

[13] X. Zhang, F. Sun, G. Liu, and Y. Ma, "Fast low-rank subspace segmentation," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 5, pp. 1293–1297, May 2014.

[14] X. Zhu, S. Zhang, Y. Li, J. Zhang, L. Yang, and Y. Fang, "Low-rank sparse subspace for spectral clustering," *IEEE Trans. Knowl. Data Eng.*, vol. 31, no. 8, pp. 1532–1543, Aug. 2019.

[15] Y. Ding, S. Pan, and Y. Chong, "Robust spatial–spectral block diagonal structure representation with fuzzy class probability for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 3, pp. 1747–1762, Mar. 2020.

[16] C. Xing, M. Wang, Z. Wang, C. Duan, and Y. Liu, "Diagonalized low-rank learning for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–12, 2022.

[17] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma, "Robust recovery of subspace structures by low-rank representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 1, pp. 171–184, Jan. 2013.

[18] S. Xiao, M. Tan, D. Xu, and Z. Y. Dong, "Robust kernel low-rank representation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 11, pp. 2268–2281, Nov. 2016.

[19] G. Liu and S. Yan, "Latent low-rank representation for subspace segmentation and feature extraction," in *Proc. Int. Conf. Comp. Vision*, 2011, pp. 1615–1622.

[20] G. Liu, Z. Lin, and Y. Yu, "Robust subspace segmentation by low-rank representation," in *Proc. ICML*, vol. 1, 2010, p. 8.

[21] E. Elhamifar and R. Vidal, "Sparse subspace clustering: Algorithm, theory, and applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 11, pp. 2765–2781, Nov. 2013.

[22] L. Fan, G. Lu, Y. Wang, and T. Liu, "Block diagonal sparse subspace clustering," in *Proc. 13th Int. Conf. Wireless Commun. Signal Process. (WCSP)*, 2021, pp. 1–6.

[23] J. Wang, K. Zhang, P. Wang, K. Madani, and C. Sabourin, "Unsupervised band selection using block diagonal sparsity for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens. Lett.*, vol. 14, no. 11, pp. 2062–2066, Nov. 2017.

[24] A. Taştan, M. Muma, and A. M. Zoubir, "Sparsity-aware robust community detection," *Signal Process.*, vol. 187, 2021, Art. no. 108147.

[25] B. Nasihatkon and R. Hartley, "Graph connectivity in sparse subspace clustering," in *Proc. CVPR*, 2011, pp. 2137–2144.

[26] S. Arora, S. Rao, and U. Varizani, "Expander flows, geometric embeddings and graph partitioning," *J. ACM*, vol. 56, pp. 1–37, 2009.

[27] N. García-Pedrajas, J. A. R. Del Castillo, and G. Cerruela-García, "A proposal for local k values for k-nearest neighbor rule," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 2, pp. 470–475, Feb. 2017.

[28] S. S. Mullick, S. Datta, and S. Das, "Adaptive learning-based k-nearest neighbor classifiers with resilience to class imbalance," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 11, pp. 5713–5725, Nov. 2018.

[29] A. M. Zoubir, V. Koivunen, E. Ollila, and M. Muma, *Robust Statistics for Signal Processing*. Cambridge, U.K.: Cambridge Univ. Press, 2018.

[30] P. J. Rousseeuw and A. M. Leroy, *Robust Regression and Outlier Detection*. Hoboken, NJ, USA: Wiley, 2005.

[31] A. Taştan, M. Muma, and A. M. Zoubir, "Robust regularized locality preserving indexing for Fiedler vector estimation," submitted for publication.

[32] T. Hastie and P. Y. Simard, "Metrics and models for handwritten character recognition," *Stat. Sci.*, vol. 13, no. 1, pp. 54–65, 1998.

[33] S. Arora, S. Rao, and U. Vazirani, "Geometry, flows, and graph-partitioning algorithms," *Commun. ACM*, vol. 51, pp. 96–105, 2008.

[34] D. Cai, X. He, and J. Han, "Document clustering using locality preserving indexing," *IEEE Trans. Knowl. Data Eng.*, vol. 17, no. 12, pp. 1624–1637, Dec. 2005.

[35] D. Cai, X. He, W. V. Zhang, and J. Han, "Regularized locality preserving indexing via spectral regression," in *Proc. 16th ACM Conf. Inf. Knowl. Manage.*, 2007, pp. 741–750.

[36] J. Lu and Y.-P. Tan, "Regularized locality preserving projections and its extensions for face recognition," *IEEE Trans. Syst. Man Cybern. B Cybern.*, vol. 40, no. 3, pp. 958–963, Jun. 2010.

[37] M. Artac, M. Jogan, and A. Leonardis, "Incremental PCA for on-line visual learning and recognition," in *Proc. Int. Conf. Pattern Recognit.*, vol. 3, 2002, pp. 781–784.

[38] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," in *Proc. Adv. Process. Neural Inf. Syst.,* vol. 14, 2001, pp. 849–856.

[39] 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.

[40] A. Taştan, M. Muma, and A. M. Zoubir, "Robust spectral clustering: A locality preserving feature mapping based on M-estimation," in *Proc. 29th Eur. Signal Process. Conf.*, 2021, pp. 851–855.

[41] M. Belkin and P. Niyogi, "Laplacian eigenmaps and spectral techniques for embedding and clustering," in *Proc. Conf. Adv. Neural Inf. Process. Syst.*, vol. 14, 2001, pp. 585–591.

[42] M. Belkin and P. Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," *Neural Comput.*, vol. 15, pp. 1373–1396, 2003.

[43] Z. Li, F. Nie, X. Chang, L. Nie, H. Zhang, and Y. Yang, "Rank-constrained spectral clustering with flexible embedding," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 12, pp. 6073–6082, Dec. 2018.

[44] X. Li, W. Hu, C. Shen, A. Dick, and Z. Zhang, "Context-aware hypergraph construction for robust spectral clustering," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 10, pp. 2588–2597, Oct. 2014.

[45] A. Taştan, M. Muma, and A. M. Zoubir, "Supplementary information II: Fast and robust sparsity-aware block diagonal representation," 2023. Accessed: Dec. 2, 2023. [Online]. Available: https://arxiv.org/abs/2312.01137

[46] D. Matula and F. Shahrokhi, "Sparsest cuts and bottlenecks in graphs," *Discrete Appl. Math.*, vol. 27, pp. 113–123, 1990.

[47] R. Andersen and Y. Peres, "Finding sparse cuts locally using evolving sets," in *Proc. 41st Annu. Symp. Theory Comput.*, 2009, pp. 235–244.

[48] U. Von Luxburg, "A tutorial on spectral clustering," *Stat. Comput.*, vol. 17, pp. 395–416, 2007.

[49] A. Taştan, M. Muma, E. Ollila, and A. M. Zoubir, "Sparsity-aware block diagonal representation for subspace clustering," in *Proc. 31th Eur. Signal Process. Conf.*, to be published.

[50] K. Avrachenkov, L. Cottatellucci, and A. Kadavankandy, "Spectral properties of random matrices for stochastic block model," in *Proc. 13th Int. Symp. Model. Optim. Mobile, Ad Hoc, Wireless Netw. (WiOpt)*, 2015, pp. 537–544.

[51] J. Miettinen, S. Vorobyov, and E. Ollila, "Modelling and studying the effect of graph errors in graph signal processing," *Signal Process.*, vol. 189, 2021, Art. no. 108256.

[52] M. Tang, "The eigenvalues of stochastic blockmodel graphs," 2018. Accessed: Mar. 30, 2018. [Online]. Available: https://arxiv.org/abs/1803.11551

[53] A. Athreya, J. Cape, and M. Tang, "Eigenvalues of stochastic blockmodel graphs and random graphs with low-rank edge probability matrices," *Sankhya A*, vol. 46, pp. 1–28, 2021.

[54] E. Cuthill and J. McKee, "Reducing the bandwidth of sparse symmetric matrices," in *Proc. 24th Nat. Conf.*, 1969, pp. 157–172.

[55] R. A. Fisher, "The use of multiple measurements in taxonomic problems," *Ann. Eugen.*, vol. 7, pp. 179–188, 1936.

[56] F. K. Teklehaymanot, A.-K. Seifert, M. Muma, M. G. Amin, and A. M. Zoubir, "Bayesian target enumeration and labeling using radar data of human gait," in *Proc. 26th Eur. Signal Process. Conf. (EUSIPCO)*, 2018, pp. 1342–1346.

[57] Z. He, M. Zhang, and H. Zhang, "Data-driven research on chemical features of Jingdezhen and Longquan celadon by energy dispersive X-ray fluorescence," *Ceram. Int.*, vol. 42, pp. 5123–5129, 2016.

[58] R. Killick, P. Fearnhead, and I. A. Eckley, "Optimal detection of changepoints with a linear computational cost," *J. Am. Stat. Assoc.*, vol. 107, pp. 1590–1598, 2012.

[59] X. Yang et al., "Piecewise linear regression based on plane clustering," *IEEE Access*, vol. 7, pp. 29845–29855, 2019.

[60] G. W. Stewart, *Matrix Algorithms: Volume I Basic Decompositions*. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 1998.

[61] G. W. Stewart, *Matrix Algorithms: Volume II Eigensystems*. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 2001.

[62] J. J Hull, "A database for handwritten text recognition research," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 16, no. 5, pp. 550–554, May 1994.

[63] S. A. Nene, S. K. Nayar, and H. Murase, "Columbia object image library (coil-20)," Tech. Rep., Columbia Univ., 1995.

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

[65] M. Lyons, S. Akamatsu, M. Kamachi, and J. Gyoba, "Coding facial expressions with gabor wavelets," in *Proc. 3rd IEEE Int. Conf. Autom. Face Gesture Recognit.*, 1998, pp. 200–205.

[66] P. N Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces vs. fisherfaces: Recognition using class specific linear projection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 711–720, Jul. 1997.

[67] W. H. Wolberg and O. L. Mangasarian, "Multisurface method of pattern separation applied to breast cytology diagnosis," *Proc. Natl. Acad. Sci.*, vol. 87, pp. 9193–9196, 1989.

[68] A. R. Rocha Neto, R. Sousa, G. A. Barreto, and J. S. Cardoso, "Diagnostic of pathology on the vertebral column with embedded reject option," in *Proc. Iberian Conf. Pattern Recognit. Image Anal.*, 2011, pp. 588–595.

[69] A.-K. Seifert, M. Amin, and A. M. Zoubir, "Toward unobtrusive in-home gait analysis based on radar micro-Doppler signatures," *IEEE Trans. Biomed. Eng.*, vol. 66, no. 9, pp. 2629–2640, Sep. 2019.

[70] T. P. Conrads et al., "High-resolution serum proteomic features for ovarian cancer detection," *Endocr.-Relat. Cancer*, vol. 11, pp. 163–178, 2004.

[71] L. Naranjo, C. J. Perez, Y. Campos-Roca, and J. Martin, "Addressing voice recording replications for Parkinson's disease detection," *Expert Syst. Appl.*, vol. 46, pp. 286–292, 2016.

[72] N. Shahid, N. Perraudin, V. Kalofolias, G. Puy, and P. Vandergheynst, "Fast robust PCA on graphs," *IEEE J. Sel. Top. Signal Process*, vol. 10, pp. 740–756, 2016.

[73] A. Bojchevski, Y. Matkovic, and S. Günnemann, "Robust spectral clustering for noisy data: Modeling sparse corruptions improves latent embeddings," in *Proc. 23rd ACM SIGKDD Int. Conf. Knowl. Discovery Data Min.*, 2017, pp. 737–746.

[74] V. D. Blondel, J. L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *J. Stat. Mech: Theory Exp.*, vol. 10, 2008, Art. no. P10008.

[75] E. L. Martelot and C. Hankin, "Multi-scale community detection using stability as optimization criterion in a greedy algorithm," in *Proc. Int. Conf. Knowl. Discovery Inf. Retrieval*, 2011, pp. 208–217.

[76] I. Psorakis, S. Roberts, M. Ebden, and B. Sheldon, "Overlapping community detection using Bayesian non-negative matrix factorization," *Phys. Rev. E*, vol. 83, 2011, Art. no. 066114.

[77] X. Bai, P. Yang, and X. Shi, "An overlapping community detection algorithm based on density peaks," *Neurocomputing*, vol. 226, pp. 7–15, 2017.

[78] S. Sobolevsky, R. Campari, A. Belyi, and C. Ratti, "General optimization technique for high-quality community detection in complex networks," *Phys. Rev. E*, vol. 90, 2014, Art. no. 012811.

[79] L. Bohlin, D. Edler, A. Lancichinetti, and M. Rosvall, "Community detection and visualization of networks with the map equation framework," in *Measuring Scholarly Impact*. Cham, Switzerland: Springer International Publishing, 2014, pp. 3–34.

[80] D. Cai, X. He, J. Han, and T. S. Huang, "Graph regularized nonnegative matrix factorization for data representation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 8, pp. 1548–1560, Aug. 2011.

[81] L. Fei-Fei, R. Fergus, and P. Perona, "One-shot learning of object categories," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 4, pp. 594–611, Apr. 2006.

[82] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms," 2017. Accessed: Sep. 15, 2017. [Online]. Available: https://arxiv.org/abs/1708.07747

[83] Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Y. Ng, "Reading digits in natural images with unsupervised feature learning," in *Proc. NIPS Workshop Deep Learn. Unsupervised Feature Learn.*, 2011.