Efficient Sparse Representation for Learning with High-Dimensional Data

Jie Chen, Shengxiang Yang, *Senior Member, IEEE,* Zhu Wang, Hua Mao

***Abstract*—Due to the capability of effectively learning intrinsic structures from high-dimensional data, techniques based on sparse representation have begun to display an impressive impact in several fields, such as image processing, computer vision and pattern recognition. Learning sparse representations is often computationally expensive due to the iterative computations needed to solve convex optimization problems in which the number of iterations is unknown before convergence. Moreover, most sparse representation algorithms focus only on determining the final sparse representation results and ignore the changes in the sparsity ratio during iterative computations. In this paper, two algorithms are proposed to learn sparse representations based on locality-constrained linear representation learning with probabilistic simplex constraints. Specifically, the first algorithm, called approximated local linear representation (ALLR), obtains a closed-form solution from individual locality-constrained sparse representations. The second algorithm, called approximated local linear representation with symmetric constraints (ALLR$_{\mathrm{SC}}$), further obtains all symmetric sparse representation results with a limited number of computations; notably, the sparsity and convergence of sparse representations can be guaranteed based on theoretical analysis. The steady decline in the sparsity ratio during iterative computations is a critical factor in practical applications. Experimental results based on public datasets demonstrate that the proposed algorithms perform better than several state-of-the-art algorithms for learning with high-dimensional data.**

***Index Terms*—Sparse representation, linear representation, low-dimensional structures, probabilistic simplex.**

I. Introduction

HIGH-DIMENSIONAL data are ubiquitous in many real problems involving machine learning. In practice, the high dimensionality of data inevitably increases the memory and computational time requirements of algorithms. In fact, high-dimensional data are often characterized by intrinsically low-dimensional structures [7], [14], [35]. Consequently, exploiting the low-dimensional structures in high-dimensional data has received considerable attention in recent years [5], [8], [18], [33], [39], [52].

Sparse representation is an extremely effective method for exploiting the intrinsic structure of high-dimensional data [21], [25], [37], [43], [45]. The goal of sparse representation is to find a compact representation of high-dimensional data by selecting a small subset of a given dictionary to identify low-dimensional structures in high-dimensional data. Inspired by recent advances in l_0-norm and l_1-norm techniques, many effective algorithms based on convex optimization or greedy pursuit have been proposed for seeking such representations [46], [48], [56]. For example, a typical representative of sparse representation algorithms is LASSO [41], which uses an l_1-regularizer to penalize the coefficients of the linear combination of several elements from an overcomplete dictionary. To improve the computational efficiency of sparse representation, a feature-sign search algorithm was presented to solve the l_1-regularized least squares problem [23]. This approach significantly accelerates the sparse representation process. In addition, Axiotis *et al.* presented an adaptively regularized hard thresholding (ARHT) algorithm that provides a strong tradeoff between the restricted isometry property condition and solution sparsity [2]. The bound of the ARHT algorithm is a strict constant for a general class of algorithms, e.g., LASSO. To address the issue of biased estimation for large coefficients in l_1-norm-based models, Boob *et al.* presented a level-constrained proximal point method that translates a nonconvex constrained problem into a sequence of convex subproblems with a gradually relaxed constraint level; each subproblem can be efficiently solved based on a fast routine for projection considering the surrogate constraint [3]. However, these methods require iterative computations and at least dozens of iterations before convergence. Considering specific data samples often leads to a high computational cost associated with finding the corresponding sparse representation by utilizing the above l_1-norm-based optimization techniques.

A variety of sparse representation techniques and their variants have been proposed to find the intrinsic structures of high-dimensional data [4], [40], [49]. For instance, an algorithm based on sparse representation, called sparse subspace clustering (SSC), takes advantage of the self-expressiveness property of data, and each data point can be efficiently represented as a sparse linear combination of other points from the same subspace [14]. A low-rank sparse subspace clustering algorithm that imposes low-rank and sparseness constraints on the data representation matrix is presented to capture the global and local structures of high-dimensional data [4], [6]. Consequently, the computational cost is low when finding sparse

Manuscript received August 2, 2020; revised June 8, 2021 and September 13, 2021; accepted October 4, 2021. This work was partially supported by National Natural Science Foundation of China (NSFC) under Grants 61303015 and 61673331, Sichuan Science and Technology Program under Grant 2021YJ0078, National Key R&D Program of China: Studies on Key Technologies and Equipment Supporting A High Quality and Highly Efficient Court Trial (2018YFC0830300) and AI in Law Advanced Deployed Discipline of Sichuan University. (Corresponding author: Shengxiang Yang).

J. Chen is with the College of Computer Science, Sichuan University, Chengdu 610065, China (E-mail: chenjie2010@scu.edu.cn).

S. Yang is with the School of Computer Science and Informatics, De Montfort University, Leicester LE1 9BH, U. K. (e-mail: syang@dmu.ac.uk).

Z. Wang is with the Law School, Sichuan University, Chengdu 610065, China (E-mail: wangzhu@scu.edu.cn).

H. Mao is with the Department of Computer and Information Sciences, Northumbria University, Newcastle, NE1 8ST, U.K. (E-mail: hua.mao@northumbria.ac.uk).

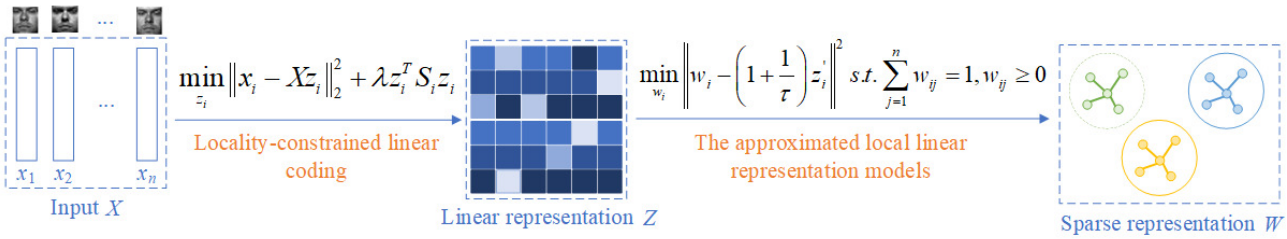


Fig. 1. Flowchart of an efficient sparse representation based on the approximated local linear representation models.

representations for all data samples. However, the majority of sparse representation algorithms aim to determine the final overall sparsity level of a sparse representation without considering individual sparsity. Therefore, each individual sparsity level for each data point cannot be theoretically guaranteed in determining the final overall sparsity of sparse representation results. Moreover, these methods often overlook the changes in the sparsity ratio during iterative computations, which is a critical factor in practical applications such as compression and image denoising [17], [31], [54].

Sparse representation coefficients can be adopted to characterize relationships among data samples [20], [22], [27]. There are three important characteristics of sparse representation coefficients: adaptive neighborhoods, sparsity and high discrimination power [47]. Specifically, sparse representation coefficients can be employed to obtain the weights for the pairwise relationship graph, which measures the similarity among high-dimensional samples [19], [42]. To capture the linear relationships among data samples, an adaptively sized neighborhood for all the data samples is essential to guarantee sparsity and effectively improve the discrimination power. Locality preservation is of considerable importance for characterizing the intrinsic structures of high-dimensional data through sparse representation and dictionary learning methods [32]. In particular, locality is more fundamental than sparsity from the perspective of data representation, as locality leads to sparsity [50]. A locality-constrained linear coding (LLC) is adopted to pursue the linear relationships among data samples instead of using the l_1-norm [44]. LLC is an efficient data representation algorithm with an analytical solution. A discriminative dictionary learning algorithm is proposed for image classification [28]; it employs the locality information and label information to ensure that similar profiles based on the corresponding components of the learned dictionary are similar. An adaptive locality-constrained latent representation method is proposed to recover multisubspace structures by encoding the adaptive locality and obtaining robust and strict block-diagonal representations [53]. These methods yield impressive results in practical applications such as face recognition and object recognition. However, the sparsity of data representations cannot be theoretically guaranteed if only on a locality constraint is considered. In addition, an efficient projection algorithm has been proposed to perform online learning in sparse feature spaces [12]. However, this algorithm cannot effectively capture the low-dimensional structures in high-dimensional data since its goal is to project a vector onto an l_1-ball.

In this paper, we propose two efficient sparse representation algorithms for learning with high-dimensional data, and they are based on local linear representation learning with a probabilistic simplex constraint. The flow chart of the proposed method is given in Fig. 1. The goal of the two algorithms extends beyond obtaining a compact high-fidelity representation of data samples. We present an iterative computational scheme for locality-constrained sparse linear representations. Based on the critical steps in this scheme, we consider the characteristics of locality-constrained linear representations and approximate projections and propose two efficient sparse representation algorithms for learning with high-dimensional data. First, we propose the approximated local linear representation (ALLR) method; the individual sparse representation results of this method can be obtained as a closed-form solution. Therefore, the ALLR approach can avoid iterative computations and achieve a low computational cost. Then, we propose the approximated local linear representation with simplex constraints (ALLR_{SC}) method; with this approach, all symmetric sparse representation results can be obtained in a limited number of computations, i.e., a strictly positive integer determined by users. The decline in the sparsity ratio during iterative computations and the convergence of a sparse representation are investigated in the experiments. As demonstrated by our experiments, the proposed methods perform better than several state-of-the-art algorithms for learning with high-dimensional data.

The main contributions of the paper can be summarized as follows:

1) The ALLR approach has a closed-form solution, and the sparsity of a final overall sparse representation of the ALLR approach can be theoretically guaranteed by considering each individual sparsity level.
2) The changes in the sparsity ratio during iterative computations indicate that the desired sparsity ratio for a final sparse representation can be obtained based on the parameter t_{max}. Additionally, t_{max} in the ALLR_{SC} method can limit the overall computational cost.
3) The convergence of a sparse representation obtained by the ALLR_{SC} method can be theoretically guaranteed.
4) Experimental results based on various real-world datasets show that the proposed algorithms significantly outperform competing algorithms in terms of performance and efficiency.

The remainder of this paper is organized as follows. Related

work on sparse representation techniques is briefly reviewed in Section II. The proposed model is presented in Section III. Section IV presents the proposed algorithms used to learn sparse representations based on locality-constrained linear representation learning with probabilistic simplex constraints. The experimental results for a variety of real datasets are presented in Section V. Finally, Section VI concludes the paper.

II. Related Work

In this section, we briefly review several sparse representation techniques that are closely related to our work.

A. Sparse Representation Methods

Let $\mathbf{D} = [\mathbf{d}_1, ..., \mathbf{d}_n] \in \mathbb{R}^{d \times n}$ be a matrix of d-dimensional data vectors. Obtaining the sparse representation of a vector $\mathbf{x} \in \mathbb{R}^d$ involves finding a sparse vector $\mathbf{z}$ in dictionary $\mathbf{D}$; most components of $\mathbf{z}$ are zero. Specifically, $\mathbf{z}$ can be obtained by solving the following l_0-norm minimization optimization problem:

$$\widetilde{\mathbf{z}} = \arg\min_{\mathbf{z}} \|\mathbf{x} - \mathbf{D}\mathbf{z}\|_2^2 + \lambda \|\mathbf{z}\|_0, \tag{1}$$

where $\lambda > 0$ is a regularization parameter and $\|\mathbf{z}\|_0$ is a count of the number of nonzero entries in the vector $\mathbf{z}$.

Problem (1) is usually relaxed to a sparsity-inducing l_1-norm minimization problem because it is a nonconvex and NP-hard problem [11]; i.e.,

$$\widetilde{\mathbf{z}} = \arg\min_{\mathbf{z}} \|\mathbf{x} - \mathbf{D}\mathbf{z}\|_2^2 + \lambda \|\mathbf{z}\|_1, \tag{2}$$

where $\|\mathbf{z}\|_1 = \sum_{i=1}^{n} |\mathbf{z}_i|$. The above optimization problem can be efficiently solved by convex optimization, such as through sparse coding algorithms [23], [41].

Let $\mathbf{Y} \in \mathbb{R}^{m \times n}$ be a matrix with n vectors. SSC is a classical sparse subspace learning method that considers the original data matrix $\mathbf{Y}$ as a dictionary in a sparse representation [14]. Specifically, the objective of SSC is to find the sparse representation for matrix $\mathbf{Y}$ by considering the following optimization problem:

$$\begin{aligned} &\min_{\mathbf{Z},\mathbf{E},\mathbf{R}} \|\mathbf{Z}\|_1 + \delta \|\mathbf{E}\|_1 + \eta \|\mathbf{R}\|_F^2 \\ &s.t.\ \mathbf{Y} = \mathbf{Y}\mathbf{Z} + \mathbf{E} + \mathbf{R},\ \mathbf{Z}^T\mathbf{1} = \mathbf{1},\ diag(\mathbf{Z}) = 0, \end{aligned} \tag{3}$$

where δ and η are regularization parameters.

To improve the discrimination ability of a sparse representation, LRRSC integrates S_0 pseudonorm regularization into the sparse representation result [4]. The following minimization problem is solved to obtain a sparse representation for the matrix $\mathbf{Y}$:

$$\min_{\mathbf{Z}} \frac{1}{2} \|\mathbf{Y} - \mathbf{Y}\mathbf{Z}\|_F^2 + \delta \|\mathbf{Z}\|_0 + \eta \|\mathbf{E}\|_{S_0} \ s.t.\ diag(\mathbf{Z}) = 0, \tag{4}$$

where the proximity operator of $\|\cdot\|_{S_0}$ is a hard thresholding function.

Problems (3) and (4) can be solved with a convex optimization approach, e.g., the alternating direction method of multipliers (ADMM). The critical step in SSC and LRRSC is to compute the sparse coefficient matrix $\mathbf{Z}$ by solving the respective optimization problems. Consequently, SSC and LRRSC explore the overall sparsity of a sparse representation for matrix $\mathbf{Y}$.

Algorithm 1 Algorithm for projection onto the simplex

1: **Input:** A vector $\mathbf{v} \in \mathbb{R}^n$ and a scalar $\alpha > 0$
2: Sort $\mathbf{v}$ into μ: $\mu_1 \geq \mu_2 \geq ... \geq \mu_n$,
3: Find $\rho = \max\left\{j : \mu_j - \frac{1}{j}\left(\sum_{r=1}^{j} \mu_r - \alpha\right) > 0, j \in [n]\right\}$,
4: Let $\delta = \frac{1}{\rho}\left(\sum_{i=1}^{\rho} \mu_i - \alpha\right)$,
5: **Output:** $\mathbf{z}$ where $z_i = max\left(v_i - \delta, 0\right), i \in [n]$.

B. Euclidean Projection to the Positive Simplex

Let $\mathbf{v} = [\mathbf{v}_1, \mathbf{v}_2 ..., \mathbf{v}_n] \in \mathbb{R}^n$ be an ordered vector; that is, $\mathbf{v}_1 \geq \mathbf{v}_2 \geq ... \geq \mathbf{v}_n$. The work of [12] proposed a novel scheme to find the minimum of $L(\mathbf{z})$ with an l_1-norm constraint on $\mathbf{z}$; i.e.,

$$\min_{\mathbf{z}} L(\mathbf{z}) \qquad s.t. \qquad \|\mathbf{z}\|_1 \leq \alpha, \tag{5}$$

where $\alpha > 0$ is a scalar. In particular, $L(\mathbf{z})$ is defined as a convex function; i.e.,

$$L(\mathbf{z}) = \frac{1}{2} \|\mathbf{z} - \mathbf{v}\|_2^2 . \tag{6}$$

This scheme first includes a Euclidean projection to the positive simplex to solve the following optimization problem:

$$min_{\mathbf{z}} \frac{1}{2} \|\mathbf{z} - \mathbf{v}\|_2^2 \quad s.t. \quad \sum_{i=1}^{n} \mathbf{z}_i = \alpha, \quad \mathbf{z}_i \geq 0. \tag{7}$$

The procedure for solving Problem (7) is given in Algorithm 1, and the temporal complexity of this process is $O(n \log(n))$. By obtaining the appropriate ρ, Algorithm 1 generally yields sparse results. By modifying Problem (6), a general l_1-norm constraint can be considered in the following optimization problem:

$$min_{\mathbf{z}} \frac{1}{2} \|\mathbf{z} - \mathbf{v}\|_2^2 \quad s.t. \quad \|\mathbf{z}\|_1 \leq \alpha. \tag{8}$$

Although finding the solution to Problem (8) requires iterative computations, the proposed approach provides faster convergence than previous methods, with a temporal complexity of $O(n)$ The computational procedure for solving Problem (8) is similar to Algorithm 1 but does not require an accurate estimate of ρ, thus promoting the sparsity of the final results.

III. A Locality-Constrained Linear Representation Model

A. Locality-Constrained Linear Representation

Let $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n] \in \mathbb{R}^{d \times n}$ be a set of data samples. To evaluate the membership among data samples, we use the self-representation property of data samples for a sparse representation. We formulate locality-constrained linear representation learning as a sparse representation problem in which all data samples are considered a dictionary of sparse

representation $\mathbf{D}$. Specifically, a general optimization problem for a sparse representation of a given data sample $\mathbf{x}_i$ can be defined as follows:

$$\min_{\mathbf{z}_i} L(\mathbf{z}_i) + \lambda \|\mathbf{z}_i\|_1. \tag{9}$$

Note that the values of the nonzero elements in a sparse vector, $\mathbf{z}_i \in \mathbb{R}^n$, can be adopted to evaluate the membership between each pair of data samples, i.e., $\mathbf{x}_i$ and the corresponding data sample. To precisely measure the similarities among data samples, we employ a probability distribution to characterize each entry of $\mathbf{z}_i$. Specifically, a probabilistic simplex constraint on $\mathbf{z}_i$ is introduced into Problem (9); i.e., $\sum_{j=1}^{n} \mathbf{z}_{ij} = 1, \mathbf{z}_{ij} \geq 0$, where $\mathbf{z}_{ij}$ represents the probability of transition from $\mathbf{x}_i$ to $\mathbf{x}_j$.

Locality suggests that certain relationships only exist between a point and its neighbors. In practice, most elements of $\mathbf{z}_i$ are zeroes since there is no relationship between a data point and data points far from it. Consequently, locality leads to the sparsity of $\mathbf{z}_i$. Based on this relation, we present a locality-constrained linear representation (LLR) model to approximately characterize such locality. Specifically, we consider locality-constrained linear coding $L(\mathbf{z})$ to obtain a sparse representation of high-dimensional data; i.e.,

$$L(\mathbf{z}) = \|\mathbf{x}_i - \mathbf{X}\mathbf{z}_i\|_2^2 + \lambda \mathbf{z}_i^T \mathbf{S}_i \mathbf{z}_i, \tag{10}$$

where $\mathbf{S}_i \in \mathbb{R}^{n \times n}$ is a locality regularization term defined as part of the Gaussian kernel to characterize an LLR. The matrix $\mathbf{S}_i$ is a locality adaptor that gives different degrees of freedom for each basis vector, such as $\mathbf{z}_i$, in proportional relation to the corresponding similarity to data point $\mathbf{x}_i$. All elements of the matrix $\mathbf{S}_i$ are set to zeros except the diagonal elements. The diagonal elements of $\mathbf{S}_i$ are defined as follows: $\mathbf{S}_i(j,j) = \exp(\frac{\|\mathbf{x}_i - \mathbf{x}_j\|_2}{\delta})$, where $\|\cdot\|_2$ denotes the l_2-norm and δ denotes the standard deviation. For example, we set δ as the average Euclidean distance for all pairs of data samples.

Through linear algebra, a new optimization problem for sparse representations can be formally described as follows:

$$\begin{aligned} &\min_{\mathbf{z}_i} \|\mathbf{x}_i - \mathbf{X}\mathbf{z}_i\|_2^2 + \lambda \mathbf{z}_i^T \mathbf{S}_i \mathbf{z}_i \\ &s.t. \quad \sum_{j=1}^{d} \mathbf{z}_{ij} = 1, \ \mathbf{z}_{ij} \geq 0. \end{aligned} \tag{11}$$

In Problem (11), the l_2-norm is adopted to characterize the error term.

B. Optimization

We present an optimization procedure to solve Problem (11) using an inexact augmented Lagrange multiplier (ALM) framework [29]. By introducing an auxiliary variable $\mathbf{k}_i$, Problem (11) can be converted into the following equivalent problem:

$$\begin{aligned} &\min_{\mathbf{z}_i, \mathbf{k}_i} \|\mathbf{x}_i - \mathbf{X}\mathbf{z}_i\|_2^2 + \lambda \mathbf{z}_i^T \mathbf{S}_i \mathbf{z}_i \\ &s.t. \quad \sum_{j=1}^{n} \mathbf{k}_{ij} = 1, \mathbf{k}_{ij} \geq 0, \mathbf{z}_i = \mathbf{k}_i. \end{aligned} \tag{12}$$

Algorithm 2 Solving Problem (11) by an inexact ALM framework

Input: $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n] \in \mathbb{R}^{d \times n}$, $\lambda > 0$
Initialize: $\mu = 10^{-3}$, $\mu_{\max} = 1, \rho = 1.1, \varepsilon = 10^{-4}$;
for $i = 1 : n$ **do**
  while not converged **do**
    Update the variables $\mathbf{z}_i$ and $\mathbf{k}_i$ as in Equation (14);
    Update the multiplier: $\mathbf{y}_i = \mathbf{y}_i + \mu(\mathbf{z}_i - \mathbf{k}_i)$;
    Update the parameter: $\mu = \min(\rho\mu, \mu_{\max})$;
    Check the convergence condition:
    $\|\mathbf{z}_i - \mathbf{k}_i\|_\infty < \varepsilon$;
  end while
end for
Output: $\mathbf{Z} = [\mathbf{z}_1, \mathbf{z}_2, ..., \mathbf{z}_n]$

The augmented Lagrangian function in Problem (12) is

$$\begin{aligned} &\min_{\mathbf{z}_i, \mathbf{k}_i, \mathbf{y}_i, \gamma_i} \|\mathbf{x}_i - \mathbf{X}\mathbf{z}_i\|_2^2 + \lambda \mathbf{z}_i^T \mathbf{S}_i \mathbf{z}_i + tr\left(\mathbf{y}_i^T (\mathbf{z}_i - \mathbf{k}_i)\right) \\ &+ \frac{\mu_i}{2} \|\mathbf{z}_i - \mathbf{k}_i\|_2^2 + \theta_i \left(\sum_{j=1}^{n} \mathbf{k}_{ij} - 1\right) - \gamma_i \cdot \mathbf{k}_i, \end{aligned} \tag{13}$$

where $\mathbf{y}_i \in \mathbb{R}^n$ and $\gamma_i \in \mathbb{R}^n$ are two vectors of the Lagrange multiplier and $\theta_i \in \mathbb{R}$ and $\mu_i \in \mathbb{R}$ are penalty parameters.

The above optimization problem can be effectively solved with an inexact ALM framework. The variables $\mathbf{z}_i$ and $\mathbf{k}_i$ can be alternately updated at each step, and the other variable is fixed. The updating schemes for the $(t+1)$-th iteration are:

$$\begin{aligned} \mathbf{z}_i^{t+1} &= \left(\mathbf{X}^T\mathbf{X} + \lambda \mathbf{S}_i + \mu_i \cdot \mathbf{E}\right)^{-1} \left(\mathbf{X}^T \mathbf{x}_i + \mu_i \mathbf{k}_i - \mathbf{y}_i\right), \\ \mathbf{k}_i^{t+1} &= \arg min \left\| \mathbf{k}_i - \left(\mathbf{z}_i + \frac{\mathbf{y}_i}{\mu_i}\right) \right\|_2^2 \\ &\quad + \theta_i \left(\sum_{j=1}^{n} \mathbf{k}_{ij} - 1\right) - \gamma_i \cdot \mathbf{k}_i, \end{aligned} \tag{14}$$

where $\mathbf{E} \in \mathbb{R}^{n \times n}$ is a matrix with elements equal to one. The second equation in Problem (14) is solved by Algorithm 1. The complete procedure for solving Problem (11) is given in Algorithm 2.

IV. Approximated Local Linear Representation Models

The goal of this work is to efficiently find sparse representations of high-dimensional data to evaluate the membership among data samples. Most sparse representation algorithms focus on the final overall sparsity of a sparse representation without considering individual sparsity. The final results are compact and high-fidelity representations but may not be truly sparse. Consequently, we focus on the content of sparse representations for high-dimensional data rather than compact and high-fidelity representations based on undetermined linear systems. The steps in obtaining a sparse representation include determining individual sparse representations, which naturally promote overall sparsity, and evaluating the changes in the sparsity ratio during iterative computations. Based on an

analysis of the advantages of locality-constrained linear representation and approximate projection methods, we propose two efficient sparse representation algorithms for learning with high-dimensional data.

A. Approximated Local Linear Representation

A potential disadvantage of Algorithm 2 is that it requires iterative computations for locality-constrained linear representation learning, which may lead to a high computational cost in practice. However, it is clear that solving two equations is a critical step during the iterative computations of Algorithm 2. In the first equation, a locality-constrained linear representation for high-dimensional data is obtained as a closed-form solution. Then, with the second equation, the final sparse representation results can be obtained by projecting the locality-constrained linear representation to the probabilistic simplex. Consequently, this approach yields two-stage approximated locality-constrained linear representations that can be used as sparse representations instead of those obtained by solving Problem (11).

We first consider locality-constrained linear representation learning for high-dimension data without a probabilistic simplex constraint. The first optimization problem is defined as follows:

$$\min_{\mathbf{z}_i} \|\mathbf{x}_i - \mathbf{X}\mathbf{z}_i\|_2^2 + \lambda \mathbf{z}_i^T \mathbf{S}_i \mathbf{z}_i. \tag{15}$$

The solution to Problem (15) can be analytically obtained by

$$\mathbf{z}_i^* = \left(\mathbf{X}^T\mathbf{X} + \lambda \mathbf{S}_i\right)^{-1}\left(\mathbf{X}^T\mathbf{x}_i\right). \tag{16}$$

In theory, locality must lead to sparsity, but not necessarily vice versa. Additionally, the results of locality-constrained linear coding are not necessarily sparse in practice because defining an appropriate $\mathbf{S}_i$ in Problem (11) is difficult. Consequently, we further focus on obtaining a sparse representation from locality-constrained linear coding with a probabilistic simplex constraint. The results of the locality-constrained linear coding method should be normalized before obtaining sparse results because of the probabilistic simplex constraint; i.e., $\mathbf{z}_i^{'} = \frac{\mathbf{z}_i^*}{sum\left(\mathbf{z}_i^*\right)}$.

The projection to the simplex algorithm involves Euclidean projection onto an l_1-ball [12]. There are two critical advantages of the simplex model that are conducive to the LLR model for characterizing locality. First, the simplex model theoretically guarantees that locality leads to the sparsity of individual linear representations in the LLR model. Second, the computational complexity of the projection to the simplex algorithm is $O(n\log(n))$; thus, this approach can be considered an efficient simplex model with a high degree of sparsity. It is beneficial to efficiently pursue sparsity for individual data samples. Consequently, we introduce the probabilistic simplex constraint in the LLR model.

The new representation should be as close to $\mathbf{z}_i^{'}$ as possible when the sparsity constraint is satisfied. In particular, each pair of representation vectors should be similar in terms of the intrinsic geometry of the data distribution, and the difference in angular information should be small. Consequently, the following optimization problem can be formed:

$$\min_{\mathbf{w}_i} \left\|\mathbf{w}_i - \mathbf{z}_i^{'}\right\|_2^2 - \frac{1}{\tau}\left\langle \mathbf{w}_i, \mathbf{z}_i^{'}\right\rangle \ \ s.t.\ \sum_{j=1}^{n} \mathbf{w}_{ij} = 1, \mathbf{w}_{ij} \geq 0, \tag{17}$$

where $\tau > 0$ is a parameter and $\langle\cdot\rangle$ represents the inner product of two vectors.

Problem (17) is equivalent to the following problem

$$\min_{\mathbf{w}_i} \left\|\mathbf{w}_i - \left(1 + \frac{1}{\tau}\right)\mathbf{z}_i^{'}\right\|_2^2 \ \ s.t.\ \sum_{j=1}^{n} \mathbf{w}_{ij} = 1, \mathbf{w}_{ij} \geq 0. \tag{18}$$

Problem (18) can be solved by Algorithm 1. Finally, the complete procedure for the two-stage approximate sparse representation is outlined in Algorithm 3.

Algorithm 3 The ALLR Algorithm

1: **Input:** $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n] \in \mathbb{R}^{d\times n}$, $\lambda > 0, \tau > 0$
2: **Initialize:** $\alpha = 1$
3: **for** $i = 1 : n$ **do**
4: Obtain solution (16) by solving Problem (15).
5: $\mathbf{z}_i^*$ is normalized as follows: $\mathbf{z}_i^{'} = \frac{\mathbf{z}_i^*}{sum\left(\mathbf{z}_i^*\right)}$.
6: Obtain the optimal solution $\mathbf{w}_i$ by solving Problem (18) using Algorithm 1.
7: **end for**
8: **Output:** $\mathbf{W} = [\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_n]$.

For given data samples, we first employ a locality-constrained linear representation to characterize locality for each individual data sample in Problem (15). Then, we use the probabilistic simplex constraint strategy to promote the sparsity of individual linear representations. Specifically, individual sparsity for each data sample is exploited in Problem (18) by combining Euclidean projections in the simplex algorithm with locality-constrained conditions. Consequently, the overall sparsity of a linear representation of samples can be guaranteed by obtaining individual sparse representations.

B. Approximated Local Linear Representation with a Symmetric Constraint

After obtaining the sparse representation $\mathbf{Z}$ in Algorithm 3, we usually perform a postprocessing step to define an affinity matrix $\mathbf{Z}^{'}$; i.e., $\mathbf{Z}^{'} = \frac{\left(\mathbf{Z}+\mathbf{Z}^T\right)}{2}$. However, each element $\mathbf{Z}_{ij}^{'}$ cannot accurately characterize the relationship between $\mathbf{x}_i$ and $\mathbf{x}_j$ in the coefficient matrix $\mathbf{Z}^{'}$.

To improve the symmetry of postprocessing, we consider a new sparse representation with a symmetric constraint for all data samples. Problem (17) can be reformulated as the following optimization problem:

$$\begin{aligned} &\min_{\mathbf{W}} \|\mathbf{W} - \mathbf{Z}\|_F^2 - \frac{1}{\tau}\langle \mathbf{W}, \mathbf{Z}\rangle \\ &s.t. \quad \sum_{j=1}^{n} \mathbf{w}_{ij} = 1, \mathbf{w}_{ij} \geq 0, \mathbf{W} = \mathbf{W}^T. \end{aligned} \tag{19}$$

Algorithm 4 The ALLR_{SC} Algorithm

Input: $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_n] \in \mathbb{R}^{d\times n}$, $\lambda > 0, t_{max} \in [n]$
Initialize: $\alpha = 1; t = 0; \varepsilon = 10^{-2}$
for $i = 1 : n$ **do**
  Obtain the optimal solution $\mathbf{z}_i^*$ using Algorithm 3.
end for
$\mathbf{W}_0 = \mathbf{Z}^{'}$;
while $t \leq t_{max}$ **do**
  $t = t + 1$;
  $\mathbf{Z}_t = \mathbf{W}_{t-1}$;
  for $i = 1 : n$ **do**
    Obtain the optimal solution $\mathbf{w}_i^t$ by solving Problem (18) using Algorithm 1.
  end for
  $\mathbf{W}_t = \frac{(\mathbf{W}_t + \mathbf{W}_t^T)}{2}$;
end while
Output: $\mathbf{W}_t$.

The above problem can be rewritten as

$$\begin{aligned} &\min_{\mathbf{W}} \left\| \mathbf{W} - \left(1 + \frac{1}{\tau}\right)\mathbf{Z} \right\|_F^2 \\ &s.t. \quad \sum_{j=1}^{n} \mathbf{w}_{ij} = 1, \mathbf{w}_{ij} \geq 0, \mathbf{W} = \mathbf{W}^T. \end{aligned} \tag{20}$$

Problem (20) can be decomposed into n independent problems for each data sample $\mathbf{x}_i$ in each iteration. Each problem can be individually solved by Algorithm 1. The complete details are shown in Algorithm 4.

To analyze the sparsity and convergence of the sparse representation results in Algorithm 4, we further consider the following problem:

$$\min_{\mathbf{w}^t} \left\| \mathbf{w}^t - \left(1 + \frac{1}{t}\right)\mathbf{z}^t \right\|_2^2 \; s.t. \; \sum_{i=1}^{n} \mathbf{w}_i^t = 1, \mathbf{w}_i^t \geq 0, \tag{21}$$

where $\mathbf{z}^0$ is given in descending order, $\mathbf{z}^t = \mathbf{w}^{t-1}$ and $t \geq 1$. Theorems 1 and 2 detail the sparsity and convergence conditions in Algorithm 4, respectively.

Theorem 1 *Let $\mathbf{w}^t$ and $\mathbf{w}^{t+1}$ be two optimal solutions to Problem* (21) *at t and $t+1$, respectively, when $t \geq 1$. The sparsity ratio (SR) of a vector $\mathbf{w}$ is defined as $SR(\mathbf{w}^t) = \frac{|\mathbf{w}^t|_0}{len(\mathbf{w}^t)}$, where len($\mathbf{w}^t$) is the number of elements in $\mathbf{w}^t$. The SR of $\mathbf{w}$ will decrease as t increases, i.e., $SR(\mathbf{w}^{t+1}) \leq SR(\mathbf{w}^t)$. Suppose $\forall i \in [1, n]$, and the above inequality holds if*

$$\mathbf{z}_i^t < \frac{1}{\rho_t (t+1)},$$

where $t > 1$ and ρ_t is the number of strictly positive elements in $\mathbf{w}^t$.

Proof *According to Algorithm 1, we have*

$$\mathbf{w}_i^t = \max\left\{ \left(1 + \frac{1}{t}\right)\mathbf{z}_i^t - \delta_t, 0 \right\}$$

and

$$\delta_t = \frac{1}{\rho_t}\left(\sum_{i=1}^{\rho_t} \left(1 + \frac{1}{t}\right)\mathbf{z}_i^t - 1 \right).$$

Then,

$$\sum_{i=1}^{n} \mathbf{z}_i^t = \sum_{i=1}^{n} \mathbf{w}_i^{t-1} = 1,$$

where $t > 1$.

Assume that the claim $\delta_t \leq 0$ holds. Thus,

$$\sum_{i=1}^{n} \mathbf{w}_i^t = \sum_{i=1}^{n} \max\left\{ \left(1 + \frac{1}{t}\right)\mathbf{z}_i^t - \delta_t, 0 \right\} > 1.$$

However, this does not hold, contradicting $\sum_{i=1}^{n} \mathbf{w}_i^t = 1$. Hence, $\delta_t > 0$.

Let $\rho_t = |\mathbf{w}^t|_0$ and $\rho_{t+1} = |\mathbf{w}^{t+1}|_0$, and we have

$$\mathbf{w}_i^{t+1} = \max\left\{ \left(1 + \frac{1}{t+1}\right)\mathbf{w}_i^t - \delta_{t+1}, 0 \right\}.$$

This implies $\rho_t \geq \rho_{t+1}$ such that $SR(\mathbf{w}^{t+1}) \leq SR(\mathbf{w}^t)$.

Suppose the following inequality holds,

$$\left(1 + \frac{1}{t}\right)\mathbf{z}_i^t - \delta_t < 0,$$

which means that the number of strictly positive elements in $\mathbf{w}^t$ is less than that in $\mathbf{z}^t$. Since $\mathbf{z}^t = \mathbf{w}^{t-1}$, this implies $\|\mathbf{w}^t\|_0 < \|\mathbf{w}^{t+1}\|_0$.

Furthermore, suppose $\forall i \in [1, n]$ and $t > 1$, and we get

$$\begin{aligned} &\left(1 + \frac{1}{t}\right)\mathbf{z}_i^t - \delta_t \\ &= \left(1 + \frac{1}{t}\right)\mathbf{z}_i^t - \frac{1}{\rho_t}\left(\sum_{i=1}^{\rho_t} \left(1 + \frac{1}{t}\right)\mathbf{z}_i^t - 1 \right) \\ &= \left(1 + \frac{1}{t}\right)\mathbf{z}_i^t - \frac{1}{\rho_t} \cdot \frac{1}{t} < 0, \end{aligned}$$

where ρ_t is the number of strictly positive elements in z^t. This means a condition satisfies $\mathbf{z}_i^t < \frac{1}{\rho_t(t+1)}$ such that the above inequality holds.

Theorem 2 *In Algorithm 4, the objective value of Problem* (21) *will decrease until convergence as t increases if $\mathbf{w}_i^t$ satisfies the following condition, i.e., $\forall i \in [1, n]$,*

$$\mathbf{w}_i^t \leq \frac{(t+1)^2}{t\,(t+2)}\mathbf{z}_i^t,$$

where $t > 1$ and ρ_t is the number of strictly positive elements in $\mathbf{w}^t$.

Proof *Suppose $\mathbf{w}^t$ and $\mathbf{w}^{t+1}$ are two optimal solutions of Problem* (21) *at t and $t+1$ respectively when $t \geq 1$. Let ρ_t and $\rho_{(t+1)}$ be the number of strictly positive elements in $\mathbf{w}^t$ and $\mathbf{w}^{t+1}$, respectively. Sort $\mathbf{z}^t$ into $\mathbf{v}^t$: $\mathbf{v}_1 \geq \mathbf{v}_2 \geq ... \geq \mathbf{v}_{\rho_t} \geq 0$. According to Theorem 1, we have $\rho_t \geq \rho_{t+1}$.*

We first consider that $\rho_t = \rho_{t+1}$. This means that the sparsity ratio (SR) of $\mathbf{v}^t$ remains unchanged. According to Algorithm 1, we have

$$\delta_t = \frac{1}{\rho_t}\left(\sum_{i=1}^{\rho_t} \left(1 + \frac{1}{t}\right)\mathbf{v}_i^t - 1 \right) = \frac{1}{\rho_t t},$$

and

$$\begin{aligned}\delta_{(t+1)} &= \frac{1}{\rho_{(t+1)}}\left(\sum_{i=1}^{\rho_{t+1}}\left(1+\frac{1}{t+1}\right)\mathbf{v}_i^{t+1}-1\right)\\ &= \frac{1}{\rho_{(t+1)}(t+1)}.\end{aligned}$$

Hence, $\delta_t > \delta_{(t+1)}$.

Let $r_t = \left\|\mathbf{W}_t - \left(1+\frac{1}{t}\right)\mathbf{V}_t\right\|_F^2$ *be the objective value of Problem* (21) *at t, where* $\mathbf{V}_t = [\mathbf{v}_1, \mathbf{v}_2, ..., \mathbf{v}_n]$. *Thus,*

$$\begin{aligned}&r_t - r_{t+1}\\ &= \left\|\mathbf{w}^t - \left(1+\frac{1}{t}\right)\mathbf{v}^t\right\|_2^2 - \left\|\mathbf{w}^{t+1} - \left(1+\frac{1}{t+1}\right)\mathbf{v}^{t+1}\right\|_2^2\\ &= \rho_t\left(\delta_t\right)^2 - \rho_{(t+1)}\left(\delta_{(t+1)}\right)^2 > 0.\end{aligned}$$

Hence, r_t *will decrease until convergence as* t *increases.*

Then we consider $\rho_t > \rho_{t+1}$, *which indicates that the SR of* $\mathbf{v}^t$ *decreases as* t *increases. According to Algorithm 1, we have*

$$\begin{aligned}r_t &= \left\|\mathbf{w}^t - \left(1+\frac{1}{t}\right)\mathbf{v}^t\right\|_2^2\\ &= \rho_t\left(\delta_t\right)^2 + \sum_{i=\rho_t}^{\rho_{(t-1)}}\left(\left(1+\frac{1}{t}\right)\mathbf{v}_i^t\right)^2\end{aligned}$$

and

$$\begin{aligned}r_{t+1} &= \left\|\mathbf{w}^{t+1} - \left(1+\frac{1}{t+1}\right)\mathbf{v}^{t+1}\right\|_2^2\\ &= \left(\rho_{(t+1)}\left(\delta_{(t+1)}\right)^2 + \sum_{i=\rho_{(t+1)}}^{\rho_t}\left(\left(1+\frac{1}{t+1}\right)\mathbf{v}_i^{t+1}\right)^2\right).\end{aligned}$$

Let $\forall i \in [\rho_{t+1}, \rho_t]$, *and there exists* $\delta_{(t+1)} > \mathbf{v}_i^{t+1}$. *Suppose* $\delta_t - \delta_{t+1} > 0$, *and we have*

$$\begin{aligned}r_t - r_{t+1} &> \rho_t(\delta_t)^2 - \left(\rho_{(t+1)}\left(\delta_{(t+1)}\right)^2 + \sum_{i=\rho_{(t+1)}}^{\rho_t}\left(\mathbf{v}_i^{t+1}\right)^2\right)\\ &> \rho_{t+1}\left(\delta_t\right)^2 - \rho_{(t+1)}\left(\delta_{(t+1)}\right)^2 > 0.\end{aligned}$$

Clearly, r_t *will decrease until convergence as* t *increases if*

$$\delta_t - \delta_{t+1} > 0.$$

Thus,

$$\begin{aligned}\delta_t - \delta_{t+1} =& \frac{1}{\rho_t}\left(\sum_{i=1}^{\rho_t}\left(1+\frac{1}{t}\right)\mathbf{v}_i^t - 1\right)\\ &- \frac{1}{\rho_{(t+1)}}\left(\sum_{i=1}^{\rho_{t+1}}\left(1+\frac{1}{t+1}\right)\mathbf{v}_i^{t+1} - 1\right)\\ >& \frac{1}{\rho_{(t+1)}}\left(\sum_{i=1}^{\rho_{(t+1)}}\left(1+\frac{1}{t}\right)\mathbf{v}_i^t - 1\right)\\ &- \frac{1}{\rho_{(t+1)}}\left(\sum_{i=1}^{\rho_{(t+1)}}\left(1+\frac{1}{t+1}\right)\mathbf{v}_i^{t+1} - 1\right)\\ =& \frac{1}{\rho_{(t+1)}}\left(\sum_{i=1}^{\rho_{(t+1)}}\left(1+\frac{1}{t}\right)\mathbf{v}_i^t - \left(1+\frac{1}{t+1}\right)\mathbf{w}_i^t\right)\\ &\geq 0.\end{aligned}$$

This means a condition satisfies

$$\mathbf{w}_i^t \leq \frac{(t+1)^2}{t(t+2)}\mathbf{v}_i^t$$

such that the above inequality holds.

In addition, we also consider a special case: $t \to +\infty$. Then, we have $\lim_{t\to+\infty}\left(1+\frac{1}{t}\right)\mathbf{z}_i^t = \mathbf{z}_i^t$, which implies $\sum_{i=1}^{n}\mathbf{z}_i^t = 1$ and $\mathbf{z}_i^t \geq 0$. Thus, the optimal solution to Problem (21) $\mathbf{w}^t$ can be written as $\mathbf{w}_i^t = \mathbf{z}_i^t$. Hence, the objective value of Problem (21) remains zero when $t \to +\infty$. This result verifies that in theory, the objective value of Problem (21) decreases until convergence when $t \to +\infty$.

According to Theorem 1, the sparsity of a sparse representation steadily declines as t gradually increases in Problem (21) if $\mathbf{z}_i^t$ satisfies a certain condition. Hence, t becomes larger as the objective value of Problem (21) decreases. Consequently, $\mathbf{w}^t$ is approximate to $\mathbf{z}^t$ as t increases. Note that we focus on sparse representations for high-dimensional data rather than finding the sparest solution in Problem (18). Consequently, we adopt t as a parameter in Problem (21) instead of τ in Problem (18) to alleviate the rate of change in sparsity.

Compared with the solution procedure for Problem (21), Algorithm 4 includes an additional step: $\mathbf{W}_t = \frac{\left(\mathbf{W}_t + \mathbf{W}_t^T\right)}{2}$. If the initial coefficients in z are closely related to neighboring samples for each sample $\mathbf{x}$, the sparsity of a sparse representation will continue to decline. This result explicitly implies the importance of $\mathbf{z}$ in the initial phase and further explains why locality-constrained linear representation is one of the most appropriate choices for initializing $\mathbf{z}$, which is a critical step in Algorithm 4.

C. Theoretical Analysis

1) Computational Complexity Analysis: Determining the computational complexity of Algorithm 3 mainly consists of two steps. Specifically, the first step is to find the solution to Problem (15), which requires solving the system of standard linear equations. It is easy to prove that $\left(\mathbf{X}^T\mathbf{X} + \lambda\mathbf{S}_i\right)^{-1}$ is a symmetric positive definite matrix. Hence, the computational complexity of the first part is $\mathcal{O}\left(dn^2 + \frac{1}{3}n^3\right)$ based on Cholesky factorization. Moreover, this computational complexity reduces to $\mathcal{O}\left(dn^2 + n^{2.38}\right)$ when applying the Coppersmith–Winograd algorithm [10], [1]. In addition, the computational complexity of the other part, Algorithm 1, is $\mathcal{O}\left(n\log\left(n\right)\right)$. The overall computational complexity of Algorithm 3 is $\mathcal{O}\left(dn^2 + n^{3.38} + n^2\log\left(n\right)\right)$, and the overall computational complexity of Algorithm 4 is $\mathcal{O}\left(dn^2 + n^{3.38} + tn^2\log\left(n\right)\right)$, where t is the number of iterations. The number of samples n is usually large, whereas the dimension d is relatively small in practice. Hence, we usually have $d \ll n$. As a result, the overall computational complexities of Algorithms 3 and 4 are $\mathcal{O}\left(n^{3.38}\right)$ and $\mathcal{O}\left(n^{3.38} + tn^2\log\left(n\right)\right)$, respectively.

2) Comparison with Sparse Representation-Based Techniques: Considering a data sample and a group of basic vectors, a sparse representation can be obtained by solving the l_1-norm-based optimization problem or its variants using

various l_1-norm optimization techniques, e.g., the least angle regression (LARS) algorithm [13]. The computational complexity of LARS is $\mathcal{O}\left(td^2n^2\right)$ for an individual data sample, where t is the number of data splits used in cross-validation. For given data samples, this method is often associated with a high overall computational cost because the solution of the l_1-norm-based optimization problem is found for each individual data sample. To reduce the computational cost, a number of sparse representation techniques for the respective optimization problems, e.g., SSC and LRRSC, are presented to obtain a sparse representation for the whole data set, X. SSC and LRRSC employ different shrinkage-thresholding operators to purse the final overall sparsity of X through iterative computations. The computational complexities of SSC and LRRSC are $\mathcal{O}\left(t\left(dn^2+n^3\right)\right)$ and $\mathcal{O}\left(dn^2+tn^3\right)$, respectively. As a result, SSC and LRRSC effectively have lower computational costs than the above methods, although they require iterative computations that may have a negative impact on individual sparse representations, i.e., real sparsity.

Algorithm 2 aims to find sparse representations for individual data samples and is one of the most representative sparse representation algorithms in terms of real sparsity. Different from -norm-based sparse representation techniques, the LLR model is presented in Algorithm 2. Unfortunately, the computational complexity of Algorithm 2 is $\mathcal{O}\left(tn^{3.38}\right)$. To avoid the high computational complexity of Algorithm 2, the ALLR and ALLR_{SC} methods are presented to learn sparse representations based on the LLR model with probabilistic simplex constraints. First, the ALLR method has a closed-form solution, and the overall computational complexity is $\mathcal{O}\left(n^{3.38}\right)$. In addition, we consider the computational complexity of the ALLR_{SC} approach: $\mathcal{O}\left(n^{3.38}+tn^2\log\left(n\right)\right)$. Thus, the iterative computational cost of the ALLR_{SC} method is a secondary factor under the condition $t \ll n$, although the ALLR_{SC} approach involves iterative computations. The two proposed methods effectively reduce the computational cost compared with that of Algorithm 2. Moreover, the computational complexities of the ALLR and ALLR_{SC} methods are comparable to those of SSC and LRRSC. Consequently, the ALLR and ALLR_{SC} methods display advantages over the representative sparse representation methods discussed above.

V. Experimental Study

In this section, we evaluate the performance of the proposed algorithms[1] in two different types of experiments: clustering and semisupervised classification. All algorithms are implemented in MATLAB 2015b. The experiments are conducted on a Windows platform with an Intel i7-9700k CPU and 32 GB RAM.

A. Experimental Setting

1) Datasets: We consider four publicly available datasets in our experiments, which are summarized below.

- **Extended Yale B Dataset** [24]. This dataset contains 2414 facial images captured from 38 individuals. There are approximately 59-64 images available for each individual. All images are resized to 48×42 pixels.
- **USPS Dataset** [16]. This dataset consists of 7,291 images of ten handwritten digits (0-9), and the image size is 16×16 pixels.
- **COIL-20 Dataset** [36]. This dataset includes 1,440 grayscale images of 20 objects. There are 72 images of each object, and each image is manually cropped to a size of 32×32 pixels.
- **ISOLET Dataset** [15]. This dataset contains 1560 data samples from 26 subjects, and each data sample includes 617 features. Each subject spoke the name of each letter of the alphabet twice. The features are composed of spectral coefficients, contour features, sonorant features, presonorant features, and postsonorant features.

2) Compared Methods: Six related graph-based algorithms were considered baselines: LRR [30], SSC [14], finding good neighbors in the subspace clustering (FGNSC) [49], LRSSC [4], stochastic sparse subspace clustering via orthogonal matching pursuit with consensus (S^3COMP-C) [9] and learnable subspace clustering (LeaSC) [26]. For LeaSC, we chose the l_1-F^2 version for comparison. In addition, a discriminative dictionary learning algorithm called the locality constrained and label embedding dictionary learning (LCLE-DL) algorithm is employed for semisupervised classification[28]. The source codes of the algorithms are provided by their authors. A spectral clustering method, NCuts [38], was employed as a final step in clustering. Moreover, a label propagation method was selected as the last step in semisupervised classification [55]. Hence, the results of the competing algorithms are used to construct graphs for the above methods, except the LCLE-DL algorithm.

3) Parameter Settings: Algorithm 1 is a critical process in Algorithms 3 and 4. The probabilistic simplex constraint on z_i is adopted in Problem (9); i.e., $\sum_{j=1}^{n}\mathbf{z}_{ij}=1, \mathbf{z}_{ij}\geq 0$. Hence, we set $\alpha=1$ in Algorithm 1. The ALLR and ALLR_{SC} methods share the same parameter λ in Problem (15). The parameters λ, τ and t were first selected from $\left\{1,5,10,50,100,500,1e^3\right\}$, $\left\{1e^{-3},5e^{-3},0.01,0.05,0.1,0.5\right\}$ and $\left\{5,10,50,100,300\right\}$ in the ALLR and ALLR_{SC} methods. Then, we slightly adjusted these parameters to obtain the best results for the ALLR and ALLR_{SC} methods in each experiment. For a fair comparison, the best results for the competing algorithms were obtained by manually adjusting their respective parameters.

4) Evaluation Metrics: We employed five standard metrics to evaluate the clustering performance of different methods: clustering accuracy (ACC) [34], normalized mutual information (NMI) [34], the adjusted rand index (Adj-RI) [51], purity [34], and the F-measure [34]. For the semisupervised classification experiments, we used classification accuracy to evaluate the performance of all the methods. A higher value of the evaluation metric indicates better clustering or classification performance. The best and second-best experimental results are shown in bold and underlined, respectively.

[1]https://codeocean.com/capsule/6062760/tree/v1

TABLE I
CLUSTERING RESULTS OF THE COMPETING METHODS (%) FOR THE SUBSETS OF THE EYB AND USPS DATASETS.

Datasets	Metrics	ALLR	ALLR_{SC}	LRR	SSC	FGNSC	LRSSC	S^3COMP-C	LeaSC
EYB	ACC	**99.38**	98.59	79.38	86.88	96.88	97.66	97.19	83.75
	NMI	**98.6**	96.94	84.01	82.27	93.61	95.17	94.11	80.65
	Purity	**99.38**	98.59	83.13	86.88	96.97	96.88	97.19	83.75
	F-measure	**99.37**	98.6	84.87	88.43	96.88	96.95	97.21	85.76
	Adj-RI	**98.61**	96.87	66.69	68.91	93.1	94.82	93.72	63.72
USPS	ACC	90.2	**92.5**	76.1	75.6	65.6	77.6	75.4	77.9
	NMI	85.74	**87.14**	77.02	78.58	61.96	77.75	77.41	78.69
	Purity	90.3	**92.5**	85.2	85.4	76	86.3	84.7	86.3
	F-measure	90.53	**92.5**	79.5	80.04	69.75	79.81	79.15	79.72
	Adj-RI	87.44	**97.7**	69.03	68.36	52.79	70.8	69.17	70.49

TABLE II
COMPUTATIONAL COSTS OF THE COMPETING METHODS (IN SECONDS) FOR THE SUBSETS OF THE EYB AND USPS DATASETS.

Datasets	ALLR	ALLR_{SC}	LRR	SSC	FGNSC	LRSSC	S^3COMP-C	LeaSC
EYB	**0.02**	0.07	15.24	1.48	0.39	1.27	8.72	1.21
USPS	**0.03**	0.71	30.44	3.48	0.91	1.16	4.02	3.28

TABLE III
CLUSTERING RESULTS OF THE COMPETING METHODS (%) FOR THE FOUR DATASETS.

Datasets	Metrics	ALLR	ALLR_{SC}	LRR	SSC	FGNSC	LRSSC	S^3COMP-C	LeaSC
EYB	ACC	**98.3**	94.33	87.37	86.08	89.4	91.88	89.56	87.41
	NMI	**97.74**	95.68	89.51	88.53	94.15	93.37	89.05	89.83
	Purity	**98.3**	94.41	87.45	86.41	89.52	91.88	89.56	87.57
	F-measure	**98.33**	95.85	92.64	91.59	91.65	94.95	91.65	92.25
	Adj-RI	**96.44**	92.02	69.76	67.9	83.86	84.97	70.56	71.85
USPS	ACC	93.17	**93.47**	76.09	77.92	83.48	86.01	88.07	79.45
	NMI	87.92	**89.27**	76.84	78.57	78.35	78.64	88.94	84.4
	Purity	93.17	**93.47**	82.94	84.57	83.31	86.01	89.67	87.19
	F-measure	93.15	**93.35**	80.01	82	84.64	86.06	90.74	84.66
	Adj-RI	88.58	**89.31**	69.12	70.19	72.06	75.38	86.63	77.16
COIL-20	ACC	88.26	**94.03**	75.42	82.22	81.18	79.38	83.13	77.85
	NMI	92	**97.84**	88.83	90.61	88.67	88.38	90.88	91.28
	Purity	88.33	**94.31**	82.29	84.58	83.33	82.43	83.26	83.96
	F-measure	88.32	**95.83**	80.77	84.44	83.2	81.33	83.98	84.12
	Adj-RI	83.74	**93.28**	71.93	75.04	77.05	74.56	76.84	76.82
ISOLET	ACC	72.24	**75.06**	68.4	64.36	61.35	67.95	69.68	66.41
	NMI	80.34	**80.83**	78.33	77.71	66.38	79.05	79.74	78.85
	Purity	74.04	**76.54**	70.06	66.73	63.58	71.28	71.54	68.53
	F-measure	73.92	**75.58**	70.7	67	62.31	70.77	71.08	69.08
	Adj-RI	63.1	**64.66**	60.51	57.37	60.78	62.01	61.95	60

B. Performance Evaluation

1) Clustering Experiments: We performed clustering experiments based on four publicly available datasets. First, subsets of the data samples were chosen from the Extended Yale B (EYB) and USPS datasets. Specifically, the first 10 subjects, which included 640 frontal face images of 10 individuals from the EYB dataset, were considered in the first experiment. In addition, the first 1,000 digit images from the USPS dataset were used in the second experiment. In particular, there were different numbers of images for each digit in the second experiment, ranging from 47-213 images per digit. Hence, the numbers of clusters are 10 and 10 and the numbers of images are 640 and 1,000 for the two experiments, respectively. The two ALRR parameters are (1) $\lambda = 1$ and $\tau = 0.2$ and (2) $\lambda = 1200$ and $\tau = 1e^{-3}$ for experiments 1 and 2, respectively. In addition, the two ALLR_{SC} parameters are (1) $\lambda = 0.8$ and $t = 3$ and (2) $\lambda = 25$ and $t = 3$ and for experiments 1 and 2, respectively. The clustering results are reported in Table I.

We observed that the ALLR and ALLR_{SC} methods perform better than the competing methods in terms of the five standard metrics: ACC, NMI, purity, the F-measure and Adj-RI. Specifically, the ALLR and ALLR_{SC} methods achieve the best and the second-best clustering results for the subset of the EYB dataset, respectively. For example, in terms of ACC, NMI, purity, the F-measure and Adj-RI, the ALLR approach achieves at least 1.72%, 3.43%, 2.19%, 2.16% and 3.79% improvements over the other competing methods, except ALLR_{SC}. Similarly, the ALLR_{SC} and ALLR methods yield the best and second-best clustering results for the subset of the USPS dataset, respectively. In addition, the computational cost is shown in Table II. Notably, the ALLR and ALLR_{SC} methods yield the lowest and the second-lowest computational costs in the experiments, respectively.

TABLE IV
COMPUTATIONAL COSTS OF THE COMPETING METHODS (IN SECONDS) FOR THE FOUR DATASETS.

Datasets	ALLR	ALLR_{SC}	LRR	SSC	FGNSC	LRSSC	S^3COMP-C	LeaSC
EYB	**0.15**	0.41	25.3	31.26	58.54	49.52	50.1	10.48
USPS	**5**	87.87	143.72	533.91	1531.79	878.74	108.39	616.44
COIL-20	**0.06**	0.45	93.08	6.48	10.33	4.84	7.25	7.73
ISOLET	**0.07**	0.77	87.92	9.98	18.6	5.2	5.61	10.05

TABLE V
SEMISUPERVISED CLASSIFICATION PERFORMANCE (MEAN ACCURACY AND STANDARD DEVIATION) OF THE COMPETING METHODS FOR THE FOUR DATASETS.

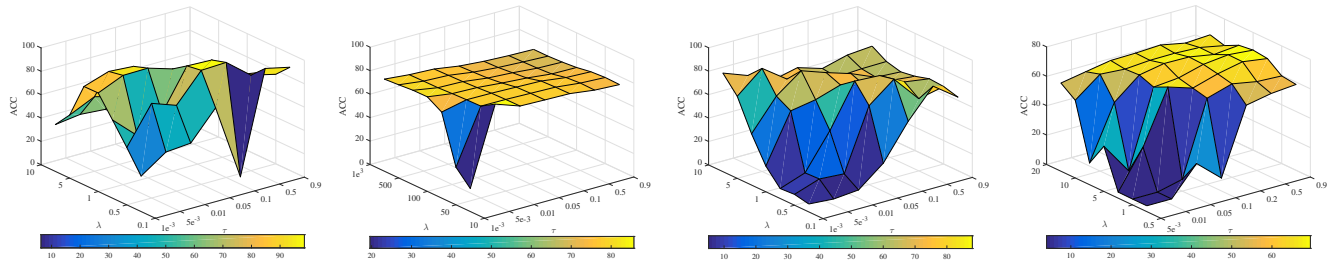
Datasets	Ratio	ALLR	ALLR_{SC}	LRR	SSC	FGNSC	LRSSC	S^3COMP-C	LeaSC	LCLE-DL
EYB	5%	97.25 (0.48)	**97.33** (0.5)	96.56 (0.43)	75.1 (4.81)	69.96 (1.91)	91.77 (1.4)	81.52 (1.49)	92.46 (1.33)	94.83 (1.59)
	10%	97.46 (0.85)	**97.94** (0.19)	96.72 (0.81)	84.66 (1.82)	78.06 (1.19)	94.03 (0.78)	86.08 (1.04)	94.61 (0.6)	95.8 (1.14)
	20%	**98.5** (0.2)	98.28 (0.18)	97.58 (1.06)	88.97 (1.5)	84.23 (1.7)	95.3 (0.5)	89.75 (0.71)	95.84 (0.39)	96.5 (0.78)
	50%	**99.01** (0.29)	98.7 (0.25)	98.28 (1.27)	95.64 (0.75)	90.86 (0.96)	96.97 (0.68)	92.03 (0.64)	96.62 (0.32)	97.48 (0.44)
USPS	5%	**95.21** (0.44)	93.14 (0.33)	92.22 (0.76)	83.24 (0.59)	41.33 (0.73)	87.29 (0.51)	91.38 (0.72)	78.23 (1.43)	85.94 (1.73)
	10%	**96.31** (0.17)	95.16 (0.36)	94.41 (0.37)	88.59 (0.72)	53.69 (5.69)	90.1 (0.35)	92.8 (0.39)	83.88 (1.02)	90.01 (1.13)
	20%	**97.15** (0.17)	96.16 (0.27)	95.47 (0.38)	92.56 (0.44)	68.21 (0.78)	91.95 (0.29)	93.94 (0.22)	87.34 (1.21)	94.43 (0.89)
	50%	**97.8** (0.13)	97.02 (0.22)	96.37 (0.17)	96.13 (0.44)	86.94 (0.73)	93.35 (0.32)	95.12 (0.24)	93.12 (1.02)	96.35 (0.63)
COIL-20	5%	**90.15** (2.74)	89.4 (1.98)	89.23 (1.13)	87.97 (3.5)	48.44 (3.7)	86.17 (3.95)	87.62 (7.12)	78.38 (2.4)	78.53 (2.67)
	10%	**94.44** (1.5)	94.13 (0.9)	92.22 (1.52)	93.52 (1.98)	60.61 (4.62)	89.99 (3.21)	93.14 (2.67)	88.01 (2.57)	88.11 (1.93)
	20%	**97.03** (0.62)	96.24 (0.71)	94.71 (15.3)	96.18 (1.06)	73..47 (2.49)	95.72 (1.27)	95.6 (4.06)	93.14 (2.25)	95.86 (1.56)
	50%	**98.89** (0.31)	98.5 (0.55)	97.04 (0.74)	97.29 (1.54)	91.34 (2.71)	97.56 (0.95)	98.07 (0.94)	98.06 (0.8)	97.59 (1.29)
ISOLET	5%	82.46 (1.97)	**84.51** (1.59)	79.84 (1.45)	81.28 (1.9)	30.51 (2.62)	76.65 (1.71)	78.23 (1.43)	81.48 (1.61)	77.12 (1.35)
	10%	87.88 (0.86)	**88.12** (1.82)	85.8 (1.32)	85.4 (1.59)	42.44 (2.88)	82.48 (1.15)	83.88 (1.02)	86.2 (1.34)	85.19 (1.3)
	20%	**91.48** (0.68)	91.16 (1.04)	90.18 (0.33)	90.07 (0.73)	56.49 (4.2)	86.88 (0.92)	87.34 (1.21)	90.93 (0.67)	90.58 (1.13)
	50%	**94.91** (1.26)	94.4 (0.79)	93.74 (0.9)	93.72 (0.79)	79.63 (4.23)	91.32 (0.73)	93.12 (1.02)	94.21 (0.83)	93.37 (0.89)

Then, we further evaluated the clustering performance of the proposed methods as the number of data samples or clusters increased. Hence, we considered all data samples from each dataset employed in the experimental evaluation. The parameters of the ALLR and ALLR_{SC} methods are set as (1) $\lambda = 2$, $\tau = 0.05$; (2) $\lambda = 200$, $\tau = 1e^{-3}$; (3) $\lambda = 30$, $\tau = 5e^{-3}$; (4) $\lambda = 20$, $\tau = 0.3$; (5) $\lambda = 2$, $t = 25$; (6)$\lambda = 1400$, $t = 150$; (7) $\lambda = 50$, $t = 300$; and (8) $\lambda = 70$, $t = 14$ for the EYB, USPS, COIL-20 and ISOLET datasets, respectively. Table III shows the clustering results of the experiments. We can see that the ALLR approach outperforms the other methods for the EYB dataset. For example, in terms of ACC, NMI, purity, the F-measure and Adj-RI, the ALLR approach achieves significant improvements of at least 6.42%, 3.59%, 6.42%, 3.38% and 11.47% over the competing methods, except ALLR_{SC}. In addition, the ALLR method achieves the second-best results for the other three datasets based on the five metrics. The ALLR_{SC} method achieves the best clustering results for the USPS, COIL and ISOLET datasets. Notably, the ALLR_{SC} approach achieves at least 5.4%, 10.9%, and 5.38% improvements in ACC when compared with the state-of-the-art methods, except ALLR, for the USPS, COIL and ISOLET datasets, respectively. Moreover, the ALLR_{SC} method consistently outperforms the other methods based on the other four metrics. In addition, LRRSC and S^3COMP-C yield satisfactory clustering results. Table IV shows the computational costs of different methods for all four datasets. The ALLR and ALLR_{SC} methods exhibit higher computational efficiency than the competing algorithms.

The number of clusters or samples dramatically increases when we consider all data samples from the EYB and USPS datasets in clustering experiments. The ALLR and ALLR_{SC} methods yield impressive experimental results compared to those of the competing algorithms for the subsets of the two datasets. As the number of clusters or samples increases, the ALLR_{SC} and ALLR methods consistently display distinct advantages in clustering ability. This result validates the importance of considering both the low-dimensional structure of high-dimensional data and the sparsity of data representations. Compared with the ALLR approach, the ALLR_{SC} method benefits from sparse representation learning during iterative computations. In addition, the computational cost of the ALLR method is much lower than that of the other algorithms in the clustering experiments. Moreover, the ALLR_{SC} method achieves relatively low computational costs for sparse representations. Consequently, the ALLR and ALLR_{SC} methods exhibit excellent efficiency in obtaining sparse representations.

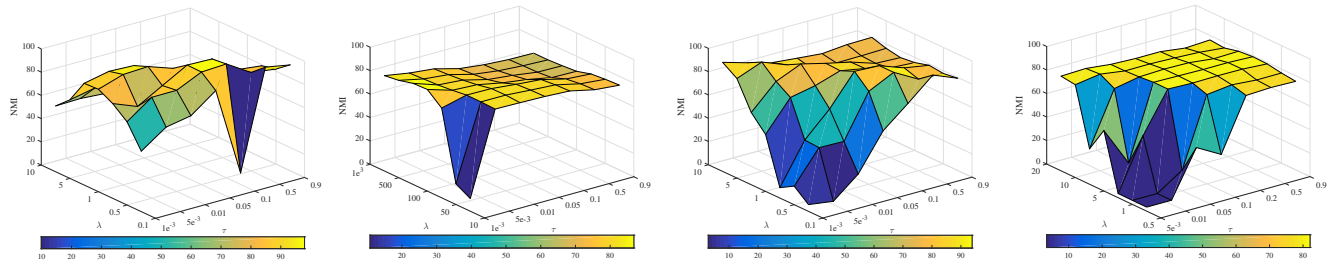
2) Semisupervised Classification Experiments: We performed semisupervised classification experiments with varying percentages of samples to evaluate the performance of the proposed methods. Specifically, 5%, 10%, 20% and 50% of data samples in each class were randomly selected and labeled in the experiments. We performed the experiments 10 times for all methods based on each dataset. The parameters of the ALLR and ALLR_{SC} methods were set as (1) $\lambda = 1$, $\tau = 0.1$; (2) $\lambda = 1,000$, $\tau = 0.01$; (3) $\lambda = 1$, $\tau = 0.1$; (4) $\lambda = 2$, $\tau = 0.6$; (5) $\lambda = 2e^6$, $t = 3$; (6) $\lambda = 700$, $t = 20$; (7) $\lambda = 15$, $t = 2$; and (8) $\lambda = 50$, $t = 30$ for the EYB, USPS, COIL-20 and ISOLET datasets, respectively.

The classification accuracies and deviations of all the meth-



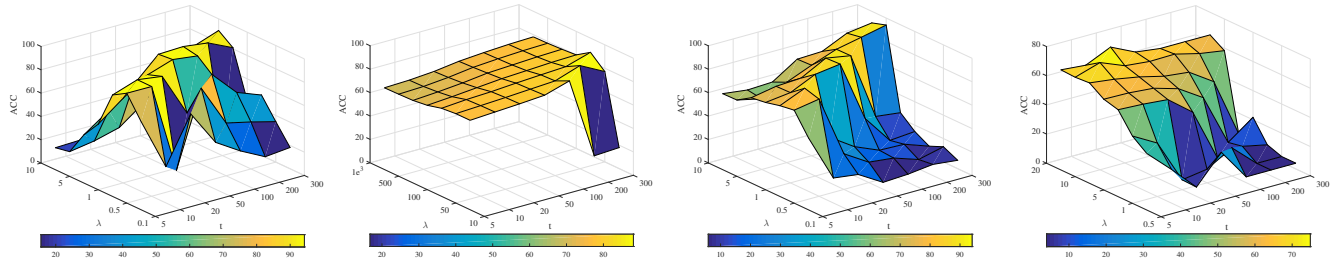
(a) The EYB dataset (b) The USPS dataset (c) The COIL-20 dataset (d) The ISOLET dataset

Fig. 2. ACC of the ALLR method (%) with variations in λ and τ for different datasets.



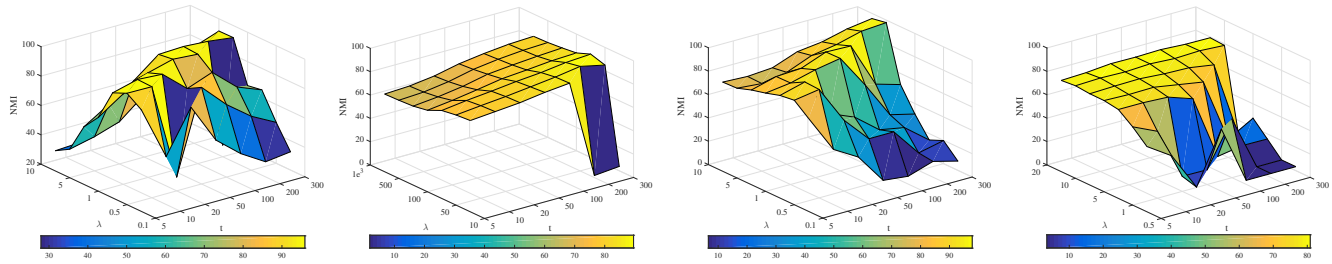
(a) The EYB dataset (b) The USPS dataset (c) The COIL-20 dataset (d) The ISOLET dataset

Fig. 3. NMI of the ALLR method (%) with variations in λ and τ for different datasets.



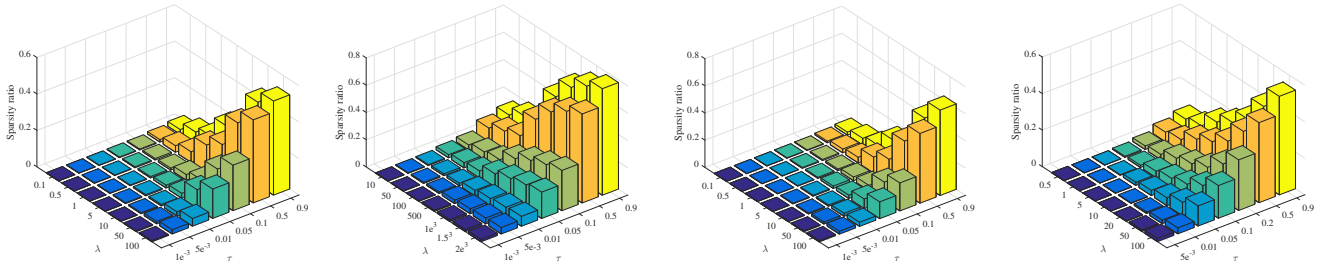
(a) The EYB dataset (b) The USPS dataset (c) The COIL-20 dataset (d) The ISOLET dataset

Fig. 4. ACC of the ALLR_{SC} method (%) with variations in λ and t for different datasets.



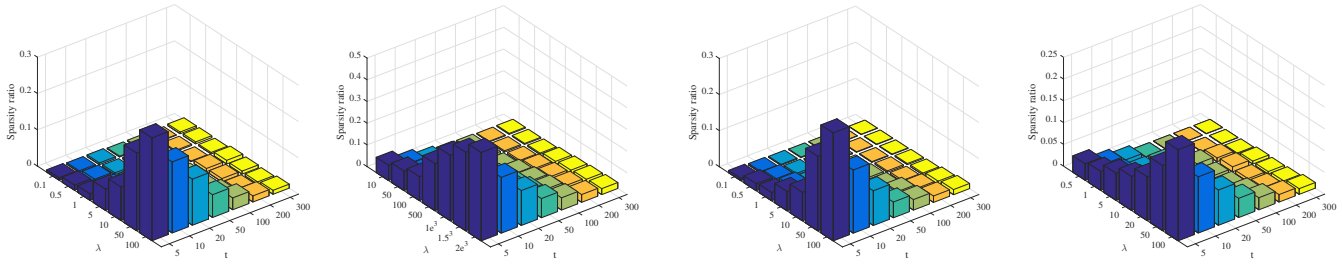
(a) The EYB dataset (b) The USPS dataset (c) The COIL-20 dataset (d) The ISOLET dataset

Fig. 5. NMI of the ALLR_{SC} method (%) with variations in λ and t for different datasets.



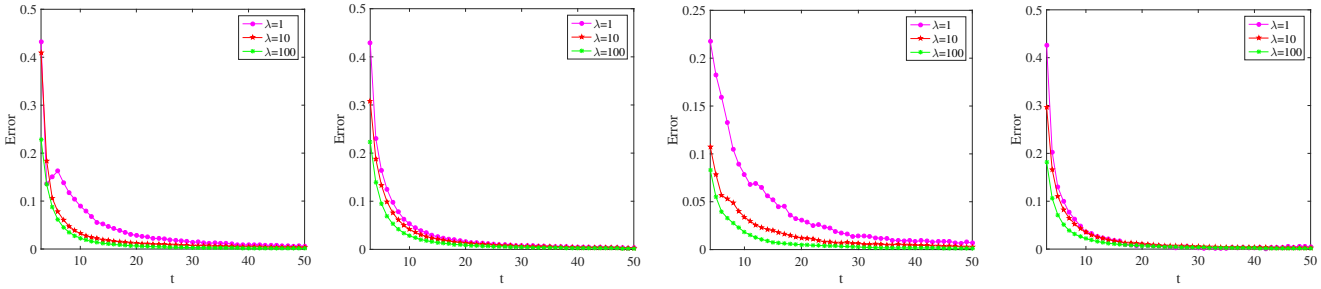
(a) The EYB dataset (b) The USPS dataset (c) The COIL-20 dataset (d) The ISOLET dataset

Fig. 6. Changes in the sparsity ratio in the ALLR method for the four datasets.



(a) The EYB dataset (b) The USPS dataset (c) The COIL-20 dataset (d) The ISOLET dataset

Fig. 7. Changes in the sparsity ratio in the ALLR_{SC} method for the four datasets.



(a) The EYB dataset (b) The USPS dataset (c) The COIL-20 dataset (d) The ISOLET dataset

Fig. 8. Convergence results for the ALLR_{SC} method based on the four datasets.

ods for the four different real-world datasets are given in Table V. As expected, the classification accuracy steadily increases as the percentage of selected data samples gradually increases. We can see that the ALLR and ALLR_{SC} methods significantly outperform the other methods in Table V. For example, the mean classification accuracy of the ALLR approach is at least 0.69%, 2.99%, 0.92% and 0.98% better than that of the competing methods, except ALLR_{SC}, for 5% of data samples from the EYB, USPS, COIL and ISOLET datasets, respectively. We can observe the same advantages for the ALLR method as the percentage of data samples increases from 5% to 50%. In addition, the ALLR_{SC} method obtained higher mean classification accuracy than the other algorithms for 5% and 10% of the EYB and ISOLET datasets. LRRSC does not yield stable competitive results compared with LRR and SSC. These classification results confirm that the affinity matrix calculated from the symmetric sparse representation effectively improves the classification accuracy.

C. Parameter Sensitivity Analysis

There are two parameters, λ and τ or t, in the ALLR and ALLR_{SC} methods. The parameter λ usually depends on the prior data distribution. Parameter τ is chosen in the range of $\left(1e^{-3}, 1\right)$ to effectively control the sparsity of the ALLR result. The parameter t is a strictly positive integer, and the number of iterations can be controlled in the experiments. In this experiment, we investigate the sensitivities of the two parameters in the ALLR and ALLR_{SC} methods and the corresponding effects on model performance.

The effects of different combinations of the parameters are reported in terms of ACC and NMI in Figs. 2 to 5. Figs. 2 and 3 show the effects of the parameters λ and τ in the ALLR method with respect to ACC and NMI, respectively. The ALLR approach performs well for a wide range of values

of λ with a fixed range of τ according to Figs. 2 and 3. In addition, from Figs. 4 and 5, we can observe that the ALLR$_{\mathrm{SC}}$ method usually performs stably with different values of t and a relatively large λ. This finding illustrates that t can be slightly controlled without much effort for parameter tuning in practical applications. Consequently, the computational cost of the ALLR$_{\mathrm{SC}}$ method can be reduced in a given period of time. These experimental results demonstrate that the ALLR and ALLR$_{\mathrm{SC}}$ methods achieve stable performance for the four datasets and that λ should be carefully chosen with a fixed range of τ or t.

According to the analysis of parameter sensitivity, the performance of ALLR and ALLR$_{\mathrm{SC}}$ has a high correlation with the value of the sparsity ratio. The performance of ALLR and ALLR$_{\mathrm{SC}}$ will degrade dramatically when the estimation of the sparsity ratio is too high or too low. From Figs. 6 and 7, we suggest that the sparsity ratio varies from 0.05 to 0.2. Moreover, we usually set $\tau = 0.1$ and $t = 50$ for ALLR and ALLR$_{\mathrm{SC}}$, respectively. Once the sparsity ratio and τ or t have been assigned, it becomes easy to determine the other parameter λ for ALLR and ALLR$_{\mathrm{SC}}$. Empirically speaking, the estimation of the sparsity ratio is an effective way to determine the combinations of λ and τ or t for ALLR and ALLR$_{\mathrm{SC}}$ in practical applications.

D. Sparsity and Convergence

Sparsity is a critical issue for sparse representations of high-dimensional data. Here, we investigate the sparsity of the results produced by the ALLR and ALLR$_{\mathrm{SC}}$ methods. Figs. 6 and 7 show the effects of the parameters on the sparsity ratios of the ALLR and ALLR$_{\mathrm{SC}}$ methods for the four datasets. We first observe that the sparsity ratio usually remains low when τ is relatively small and λ widely varies in the ALLR approach, as shown in Fig. 6. This finding suggests that τ is a key parameter for controlling the sparsity ratio. Notably, the sparsity of the results of the ALLR method is demonstrated under wide ranges of λ and τ in the experiments. In addition, the sparsity ratio remains low when t is relatively large in the ALLR$_{\mathrm{SC}}$ approach, as shown in Fig. 7. Moreover, we can see that the sparsity ratio slowly declines as t gradually increases with a fixed λ. This finding confirms that the changes in the sparsity ratio in the experiments are consistent with the theoretical results for the ALLR$_{\mathrm{SC}}$ approach. Combined with Figs. 2 - 5, a relatively low value of the sparsity ratio often leads to high performance for the ALLR and ALLR$_{\mathrm{SC}}$ methods, and vice versa. In practice, choosing the optimal parameters is difficult without prior knowledge of the data distribution, and the results demonstrate the importance of sparse representations for learning with high-dimensional data.

We further examine the convergence of the ALLR$_{\mathrm{SC}}$ method in the experiments. The F-norm was adopted to compute the iterative error between $\mathbf{W}^t$ and $\mathbf{W}^{(t-1)}$; i.e., $e = \left\|\mathbf{W}^{t+1} - \mathbf{W}^t\right\|_F^2$. We have already verified the convergence property of the ALLR$_{\mathrm{SC}}$ approach. In addition, Fig. 8 shows the plots of the iterative errors versus the iteration number t for the four datasets. We find that the ALLR$_{\mathrm{SC}}$ method usually gradually converges as t increases for all datasets. These results indicate that the ALLR$_{\mathrm{SC}}$ method can converge quickly for different values of the parameter λ. Additionally, the convergence condition of the ALLR$_{\mathrm{SC}}$ approach can generally be satisfied when learning sparse representations for the four datasets.

E. Discussion

We discuss several critical differences among the proposed methods and the sparse representation-based methods in terms of the experimental results. First, the ALLR method provides a closed-form solution, and the parameter t in ALLR$_{\mathrm{SC}}$ can limit the overall computational cost. However, the number of iterations in the other sparse representation methods is unknown before convergence. Iterative computations without a reasonable run time constraint may lead to a high computation cost. Hence, the ALLR and ALLR$_{\mathrm{SC}}$ methods are efficient for obtaining sparse representations. The experimental results demonstrate the computational efficiency of the ALLR and ALLR$_{\mathrm{SC}}$ methods.

Second, a number of sparse representation techniques focus on overall sparsity after iterative computations, and the sparsity of each data sample cannot be theoretically guaranteed in pursuing the final overall sparsity of a sparse representation. In contrast, learning based on individual sparse representations promotes overall sparsity. Hence, the final results of the ALLR and ALLR$_{\mathrm{SC}}$ methods provide actual sparse representations. The experimental results verify that maximizing individual sparsity for each data sample is an effective way to achieve sparse representations.

Finally, sparsity plays an important role in sparse representations. The sparsity ratio is an effective way to measure sparsity. However, sparse representation-based methods do not consider representation matrix data when calculating sparsity, and changes in the sparsity ratio during iterative computations are ignored. We investigated the sparsity ratio of ALLR$_{\mathrm{SC}}$ in theory and based on experiments. The decline in the sparsity ratio in ALLR$_{\mathrm{SC}}$ can be theoretically guaranteed under certain conditions. The experimental results illustrate steady changes in the sparsity ratio for different combinations of ALLR$_{\mathrm{SC}}$ parameters. By comparing the clustering results in Figs. 2 - 5 and the sparsity ratio results in Figs. 6 and 7, a relatively low sparsity ratio often leads to satisfactory clustering results. Hence, the experimental results demonstrate that theoretical analyses of the sparsity ratio are important.

VI. Conclusions

In this paper, we proposed two efficient sparse representation algorithms for learning with high-dimensional data; the algorithms are based on locality-constrained linear representation learning with a probabilistic simplex constraint and fully use a locality-constrained linear representation to adaptively choose appropriate neighbors in learning sparse representations. For the ALLR approach, we showed that the optimization problem has a closed-form solution, thus dramatically reducing the computational cost of learning sparse representations. For the ALLR$_{\mathrm{SC}}$ method, a sparse solution can be obtained with a limited number of computations. The

sparse representation results of the ALLR_{SC} approach can be obtained at a relatively low computational cost. In addition, we provided rigorous proofs of the sparsity and convergence of the ALLR_{SC} method, and the critical characteristics of this approach can be effectively guaranteed under specific conditions. Moreover, the effects of the two key parameters in the two algorithms on the sparsity ratio results were investigated for different combinations of the parameters in the experiments. Consequently, the ALLR and ALLR_{SC} methods have three advantages over other methods in obtaining sparse representations: they efficiently yield sparse representations, they provide real sparse representation, and they consider steady changes in the sparsity ratio. The experimental results for four publicly available datasets indicate that the ALLR and ALLR_{SC} methods are competitive sparse representation methods for learning with high-dimensional data.

The approximated local linear representation models effectively capture the local structures in high-dimensional data. However, there may be inadequate high-quality neighbors for some data samples in a local linear representation if the corresponding high-dimensional data are corrupted. Therefore, the global structure of high-dimensional data should be further explored to alleviate these problems. In future work, we will further investigate these problems by capturing the inherent structures of high-dimensional data, i.e., the local and global structures of data.

REFERENCES

[1] A. Ambainis, Y. Filmus, and F. L. Gall, "Fast matrix multiplication: limitations of the coppersmith-winograd method," in *Proc. 47th Annu. ACM Symp. Theory Comput. (ACM STOC)*, Portland, Oregon, USA, june 2015, pp. 585–593.

[2] K. Axiotis and M. Sviridenko, "Sparse convex optimization via adaptively regularized hard thresholding," in *Proc. 38th Int. Conf. Mach. Learn. (ICML)*, Vienna, Austria, july 2020, pp. 452–462.

[3] D. Boob, Q. Deng, G. Lan, and Y. Wang, "A feasible level proximal point method for nonconvex sparse constrained optimization," in *Advances in neural information processing systems (NIPS)*, Vancouver, British Columbia, Canada, december 2020, pp. 1–12.

[4] M. Brbić and I. Kopriva, "l_0-motivated low-rank sparse subspace clustering," *IEEE Trans. Cybern.*, vol. 50, no. 4, pp. 1711–1725, april 2020.

[5] J. Chen, H. Mao, Y. Sang, and Z. Yi, "Subspace clustering using a symmetric low-rank representation," *Knowl. Based Syst.*, vol. 127, no. 1, pp. 46–57, july 2017.

[6] J. Chen, H. Mao, Z. Wang, and X. Zhang, "Low-rank representation with adaptive dictionary learning for subspace clustering," *Knowl. Based Syst.*, vol. 223, pp. 1–12, July 2021.

[7] J. Chen, H. Mao, H. Zhang, and Z. Yi, "Symmetric low-rank preserving projections for subspace learning," *Neurocomputing*, vol. 315, no. 13, pp. 381–393, november 2018.

[8] J. Chen, H. Zhang, H. Mao, Y. Sang, and Z. Yi, "Symmetric low-rank representation for subspace clustering," *Neurocomputing*, vol. 173, no. 15, pp. 1192–1202, january 2016.

[9] Y. Chen, C. G. Li, and C. You, "Stochastic sparse subspace clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, june 2020, pp. 4155–4164.

[10] D. Coppersmith and S. Winograd, "Matrix multiplication via arithmetic progressions," *Journal of Symbolic Computation*, vol. 9, no. 3, pp. 251–280, March 1990.

[11] G. Davis, S. Mallat, and M. Avellaneda, "Adaptive greedy approximations," *Constructive approximation*, vol. 13, no. 1, pp. 57–98, 1997.

[12] J. Duchi, S. Shalev-Shwartz, Y. Singer, and T. Chandra, "Efficient projections onto the l_1-ball for learning in high dimensions," in *Proc. 25th Int. Conf. Mach. Learn. (ICML)*, Helsinki, Finland, july 2008, pp. 272–279.

[13] B. Efron, T. Hastie, I. Johnstone, and R. Tibshirani, "Least angle regression," *Annals of statistics*, vol. 32, no. 2, pp. 407–499, April 2004.

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

[15] M. Fanty and R. Cole, "Spoken letter recognition," in *Advances in neural information processing systems (NIPS)*, Denver, Colorado, USA, december 1991, pp. 220–226.

[16] J. Friedman, T. Hastie, and R. Tibshirani, *The Elements of Statistical Learning*. Springer series in statistics, 2001.

[17] W. Fu, S. Li, L. Fang, and J. A. Benediktsson, "Adaptive spectral–spatial compression of hyperspectral image with sparse representation," *IEEE Trans Geosci. Remote Sens.*, vol. 55, no. 2, pp. 671–682, february 2017.

[18] D. Gamarnik and J. Gaudio, "Sparse high-dimensional isotonic regression," in *Advances in neural information processing systems (NIPS)*, Vancouver, British Columbia, Canada, december 2019, pp. 12 852–12 862.

[19] Y. Gao, J. Ma, and A. L. Yuille, "Semi-supervised sparse representation based classification for face recognition with insufficient labeled samples," *IEEE Trans. Image Process.*, vol. 26, no. 5, pp. 2545–2560, 2017.

[20] X. Han, B. Shi, and Y. Zheng, "Self-similarity constrained sparse representation for hyperspectral image super-resolution," *IEEE Trans. Image Process.*, vol. 27, no. 11, pp. 5625–5637, november 2017.

[21] C. Huang, C. C. Loy, and X. Tang, "Sparse supervised representation-based classifier for uncontrolled and imbalanced classification," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 5, pp. 1503–1513, 2020.

[22] J. Lai and X. Jiang, "Classwise sparse and collaborative patch representation for face recognition," *IEEE Trans. Image Process.*, vol. 25, no. 7, pp. 3261–3272, july 2016.

[23] H. Lee, A. Battle, R. Raina, and A. Y. Ng, "Efficient sparse coding algorithms," in *Advances in neural information processing systems (NIPS)*, Vancouver, British Columbia, Canada, december 2007, pp. 801–808.

[24] K. Lee, J. Ho, and D. Kriegman, "Acquiring linear subspaces for face recognition under variable lighting," *IEEE Trans. Pattern Anal. and Mach. Intell.*, vol. 27, no. 5, pp. 1537–1544, may 2005.

[25] C. Li, Y. Shao, W. Yin, and M. Liu, "Robust and sparse linear discriminant analysis via an alternating direction method of multipliers," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 3, pp. 915–926, 2019.

[26] J. Li, H. Liu, Z. Tao, H. Zhao, and Y. Fu, "Learnable subspace clustering," *IEEE Trans. Neural Netw. Learn. Syst.*, pp. 1–15, december 2020.

[27] W. Li, J. Mao, Y. Zhang, and S. Cui, "Low-rank-sparse subspace representation for robust regression," in *Advances in neural information processing systems (NIPS)*, Vancouver, British Columbia, Canada, december 2018, pp. 176–184.

[28] Z. Li, Z. Lai, Y. Xu, J.Yang, and D. Zhang, "A locality-constrained and label embedding dictionary learning algorithm for image classification," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 2, pp. 278–293, 2017.

[29] Z. Lin, R. Liu, and Z. Su, "Linearized alternating direction method with adaptive penalty for low-rank representation," in *Advances in neural information processing systems (NIPS)*, Vancouver, British Columbia, Canada, december 2011, pp. 612–620.

[30] 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. and Mach. Intell.*, vol. 35, no. 1, pp. 171–184, january 2013.

[31] L. Liu, L. Chen, C. L. P. Chen, Y. Y. Tang, and C. M. Pun, "Weighted joint sparse representation for removing mixed noise in image," *IEEE Trans. Cybern.*, vol. 47, no. 3, pp. 600–611, march 2017.

[32] N. Liu, Z. Lai, X. Li, Y. Chen, D. Mo, H. Kong, and L. Shen, "Locality preserving robust regression for jointly sparse subspace learning," *IEEE Trans. Circuits Syst. Video Technol.*, pp. 1–17, august 2020.

[33] J. Lu, Z. Lai, H. Wang, Y. Chen, J. Zhou, and L. Shen, "Generalized embedding regression: A framework for supervised feature extraction," *IEEE Trans. Neural Netw. Learn. Syst.*, pp. 1–15, november 2020.

[34] C. D. Manning, P. Raghavan, and H. Sch´utze, *Introduction to Information Retrieval*. New York, NY, USA: Cambridge University Press, 2008.

[35] S. Matsushima and M. Brbić, "Selective sampling-based scalable sparse subspace clustering," in *Advances in neural information processing systems (NIPS)*, Vancouver, British Columbia, Canada, december 2019, pp. 12 416–12 425.

[36] S. A. Nene, S. K. Nayar, and H. Murase, "Columbia object image library (coil-20)," Department of Computer Science, Columbia University, New York, USA, Tech. Rep., 1996.
[37] J. Peng, L. Li, and Y. Tang, "Maximum likelihood estimation-based joint sparse representation for the classification of hyperspectral remote sensing images," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 6, pp. 1790–1802, 2019.
[38] J. Shi, J. Malik, and S.Sastry, "Normalized cuts and image segmentation," *IEEE Trans. Pattern Anal. and Mach. Intell.*, vol. 22, no. 8, pp. 888–905, 2000.
[39] T. Shu, B. Zhang, and Y. Tang, "Discriminative sparse neighbor approximation for imbalanced learning," *IEEE Trans. Neural Netw. Learn. Syst.*, pp. 1–10, 2018.
[40] Y. Sun, Z. Zhang, W. Jiang, Z. Zhang, L. Zhang, S. Yan, and M. Wang, "Discriminative local sparse representation by robust adaptive dictionary pair learning," *IEEE Trans. Neural Netw. Learn. Syst.*, pp. 1–15, 2020.
[41] R. Tibshirani, "Regression shrinkage and selection via the lasso: a retrospective," *Journal of the Royal Statistical Society: Series B*, vol. 58, no. 1, pp. 267–288, april 1996.
[42] S. Verma and Z. Zhang, "Hunt for the unique, stable, sparse and fast feature learning on graphs," in *Advances in neural information processing systems (NIPS)*, Long Beach Convention Center, Long Beach, CA, USA, december 2017, pp. 88–98.
[43] J. Wan, Z. Lai, J. Li, J. Zhou, and C. Gao, "Robust facial landmark detection by multiorder multiconstraint deep networks," *IEEE Trans. Neural Netw. Learn. Syst.*, pp. 1–14, january 2021.
[44] J. Wang, J. Yang, K. Yu, F. Lv, and Y. Gong, "Locality-constrained linear coding for image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Vancouver, British Columbia, Canada, june 2010, pp. 3360–3367.
[45] L. Wang, R. Chan, and T. Zeng, "Probabilistic semi-supervised learning via sparse graph structure learning," *IEEE Trans. Neural Netw. Learn. Syst.*, pp. 1–15, 2020.
[46] Y. Wang, Q. Yao, J. T. Kwok, and L. M. Ni, "Scalable online convolutional sparse coding," *IEEE Trans. Image Process.*, vol. 27, no. 10, pp. 4850–4859, 2018.
[47] J. Wright, Y. Ma, J. Mairal, G. Sapiro, T. S. Huang, and S. Yan, "Sparse representation for computer vision and pattern recognition," *Proceedings of the IEEE*, vol. 98, no. 6, pp. 1031–1044, 2010.
[48] Z. J. Xiang, H. Xu, and P. J. Ramadge, "Learning sparse representations of high dimensional data on large scale dictionaries," in *Advances in neural information processing systems (NIPS)*, Vancouver, British Columbia, Canada, december 2011, pp. 612–620.
[49] J. Yang, J. Liang, K. Wang, P. L. Rosin, and M. Yang, "Subspace clustering via good neighbors," *IEEE Trans. Pattern Anal. and Mach. Intell.*, vol. 42, no. 6, pp. 1537–1544, june 2020.
[50] K. Yu, T. Zhang, and Y. Gong, "Nonlinear learning using local coordinate coding," in *Advances in neural information processing systems (NIPS)*, Vancouver, British Columbia, Canada, december 2009, pp. 2223–2231.
[51] S. Zhang, H. S. Wong, and Y. Shen, "Generalized adjusted rand indices for cluster ensembles," *Pattern Recognit.*, vol. 45, no. 6, pp. 2214–2226, Jun. 2012.
[52] Y. Zhang, D. Shi, J. Gao, and D. Cheng, "Low-rank-sparse subspace representation for robust regression," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, Hawaii, USA, july 2017, pp. 7445–7554.
[53] Z. Zhang, J. Ren, S. Li, R. Hong, Z. Zha, and M. Wang, "Robust subspace discovery by block-diagonal adaptive locality-constrained representation," in *Proc. 27th ACM Int. Conf. Multimedia (ACM MM)*, Vancouver, British Columbia, Canada, december 2019, pp. 1569–1577.
[54] X. Zhou, M. Zhu, S. Leonardos, and K. Daniilidis, "Sparse representation for 3d shape estimation: A convex relaxation approach," *IEEE Trans. Pattern Anal. and Mach. Intell.*, vol. 39, no. 8, pp. 1648–1661, august 2017.
[55] X. Zhu and Z. Ghahramani, "Learning from labeled and unlabeled data with label propagation," The Center for Automated Learning and Discovery, Carnegie Mellon University, Pittsburgh, Pennsylvania, USA, Tech. Rep., 2002.
[56] W. Zuo, D. Meng, L. Zhang, X. Feng, and D. Zhang, "A generalized iterated shrinkage algorithm for non-convex sparse coding," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Honolulu, Hawaii, USA, july 2013, pp. 217–224.

Jie Chen received the BSc degree in Software Engineering, MSc degree and PhD degree in Computer Science from Sichuan University, Chengdu, China, in 2005, 2008 and 2014, respectively. From 2008 to 2009, he was with Huawei Technologies Co., Ltd. as a software engineer. He is currently an Associate Professor with the College of Computer Science, Sichuan University, China. His current research interests include machine learning, big data analysis, and deep neural networks.

Shengxiang Yang (M'00–SM'14) received the PhD degree from Northeastern University, Shenyang, China in 1999. He is currently a Professor in Computational Intelligence and Director of the Centre for Computational Intelligence, School of Computer Science and Informatics, De Montfort University, Leicester, U.K. He has over 340 publications with an H-index of 61 according to Google Scholar. His current research interests include evolutionary computation, swarm intelligence, artificial neural networks, data mining and data stream mining, and relevant real-world applications. He serves as an Associate Editor/Editorial Board Member of a number of international journals, such as the *IEEE Transactions on Evolutionary Computation*, *IEEE Transactions on Cybernetics*, *Information Sciences*, and *Enterprise Information Systems*.

Zhu Wang received the B.M., MSc., and LLD. degrees in Civil and Commercial Law from Renmin University of China, China in 2003, 2006, and 2009, respectively. He is currently a Professor in Law and Director of Institute of Rule of Law of Market Economy, Sichuan University, Chengdu, China. His research interests are mainly in Tort, insurance law, constitution and big data analysis of law.

Hua Mao received the B.S. degree and M.S. degree in Computer Science from University of Electronic Science and Technology of China (UESTC) in 2006 and 2009, respectively. She received her Ph.D. degree in Computer Science and Engineering from Aalborg University, Denmark in 2013. She is currently a Senior Lecturer in Department of Computer and Information Sciences, Northumbria University, U.K. Her current research interests include Deep Neural Networks and Big Data.