Cognitive Computation (2019) 11:697–710
https://doi.org/10.1007/s12559-019-09650-2

Joint Sparse Regularization for Dictionary Learning

Jianyu Miao[1] · Heling Cao[1] · Xiao-Bo Jin[2] · Rongrong Ma[3,4] · Xuan Fei[1] · Lingfeng Niu[4,5]

Received: 5 December 2018 / Accepted: 10 May 2019 / Published online: 10 June 2019

Abstract

As a powerful data representation framework, dictionary learning has emerged in many domains, including machine learning, signal processing, and statistics. Most existing dictionary learning methods use the ℓ_0 or ℓ_1 norm as regularization to promote sparsity, which neglects the redundant information in dictionary. In this paper, a class of joint sparse regularization is introduced to dictionary learning, leading to a compact dictionary. Unlike previous works which obtain sparse representations independently, we consider all representations in dictionary simultaneously. An efficient iterative solver based on ConCave-Convex Procedure (CCCP) framework and Lagrangian dual is developed to tackle the resulting model. Further, based on the dictionary learning with joint sparse regularization, we consider the multi-layer structure, which can extract the more abstract representation of data. Numerical experiments are conducted on several publicly available datasets. The experimental results demonstrate the effectiveness of joint sparse regularization for dictionary learning.

Keywords Dictionary learning · Joint sparse regularization · Multi-layer structure

Introduction

Dictionary learning (DL) aims to estimate a collection of basis vectors, over which the given data can be accurately reconstructed, often with a sparse coefficient. It has a variety of applications, such as feature extraction, clustering, image denoising, image reconstruction, image recovery, outlier identification, and face recognition. In the past decades, due to the simplicity, a predefined dictionary is appealing, which usually leads to simple and fast algorithms for the evaluation of the sparse representation. This is indeed the case for over-complete discrete cosine transformation, steerable wavelet filters, curvelets, and short-time Fourier transform [1]. Learning the dictionary instead of using off-the-shelf ones has been shown to improve signal reconstruction significantly and has better performance compared with predefined dictionary-based methods. However, these methods are usually complex and time-consuming.

Sparsity is significantly important for both predefined based and learned based dictionary learning methods. Among the overcomplete atoms set, it can help identify the most discriminative ones for each observed example. An intuitive sparse metric is the ℓ_0, which indicates the number of non-zero elements in a vector. Several recent researches [1, 13, 19, 34, 52] have made attempts to build a model based on the ℓ_0 constrained. The most representative dictionary learning based on ℓ_0 is the K-SVD algorithm [1], which has been successfully used in image denoising. To make the learned dictionary more discriminative, discriminative KSVD [52] and label consistent KSVD [13] incorporated the label information of data into the procedure of learning dictionaries. Liu et al. [19] proposed to use structured dictionary learning for discovering topical objects from video.

In addition, a majority of existing DL methods also use the ℓ_1 as regularization to promote the sparsity [12, 14, 42, 43]. Considering the total training set as the dictionary, sparse representation based classification (SRC) [43] solved the ℓ_1 minimization problem to determine the class to which a new test sample belongs. Later, several works [26, 27, 44] have tried to improve SRC from different perspectives, including group-sparsity and non-linear extension. Further, Zhang et al. [50] and Chen et al. [6] proposed to use the kernel trick to handle the non-linear extension. Jiang et al. [12] proposed the robust dictionary learning with emphasizing joint minimization on the capped ℓ_1 loss and ℓ_1 regularization. As indicated in [29], the ℓ_1 usually introduce extra bias in estimation, and cannot recover a signal with the

✉ Lingfeng Niu
niulf@ucas.ac.cn

Extended author information available on the last page of the article.

least measurements when applied to compressed sensing. Thus, a great number of efforts are dedicated to improving from the ℓ_1. Among such efforts, an improvement is the suggestion of the use of $\ell_p(0 < p < 1)$ regularization [30, 41]. To better approximate ℓ_0, Wang et al. [41] proposed to use the ℓ_p to improve the capability of data representation.

Motivated by the great success of the multi-layer architecture in deep learning, deep matrix factorization (DMF) approaches [18, 22, 39, 46], in which the dictionary is factorized in a way such that each factor adds an extra layer of abstraction, have been proposed and studied. Liu et al. [22] proposed a deep non-negative matrix factorization for estimating the illuminant of an image, and demonstrated the better performance compared with the state-of-the-art methods. As shown in [39], factorizing the data into multiple factors is able to extract the various attributes of the data. Using deep sparse representation (DSR), Sharma et al. [33] derived a feature representation for speech recognition. Also, Tariyal et al. [37] proposed the concept of deep dictionary learning, which combined dictionary learning and deep learning. The corresponding model can be solved in a greedy layer-by-layer manner.

Practically, most existing approaches utilize the ℓ_0 or ℓ_1 to promote the sparsity of the coefficient. Due to the flat nature of the ℓ_0 and ℓ_1, all the atoms in the learned dictionary are evenly treated and subsequently used for sparse representation. However, because the underlying high-level patterns of the input data are unknown in advance, the dictionary may contain redundant information. In addition, they appear to require learning relatively a large dictionary to obtain good performance, leading to a high computation cost. Therefore, the compact dictionary is highly desirable in real applications.

In this paper, we do not promote the sparsity of coefficient of each instance independently, but all coefficients together, that is the coefficient matrix. To the best of our knowledge, there is rare work considering the sparsity of the coefficient matrix in dictionary learning. To be more specific, we propose to enforce row sparsity on the coefficient matrix, leading that some atoms would be simultaneously removed for all the instances in the sparse representation stage. This strategy can effectively handle the redundancy in the dictionary and obtain a compact dictionary at the same time. Thus, we propose a class of joint sparse regularized dictionary learning (JSDL) methods. An effective optimization algorithm is presented in the case that $\ell_{2,1-2}$ is chosen as regularization, where CCCP and Lagrangian dual method are adopted to optimize the subproblem w.r.t. the sparse coefficient and dictionary, respectively. An adopted alternating direction method of multipliers (ADMM) is embedded to solve the sequence of convex subproblems in CCCP efficiently. Inspired by the great success of multi-layer architecture, we extend JSDL to the scenario of multiple layers and name it M-JSDL for convenience. And, the corresponding algorithm can be implemented based on that of JSDL. Extensive experiments are conducted to test the effectiveness of the proposed methods.

The main contributions are summarized as follows:

1. To obtain a more compact dictionary, we propose to consider all sparse representations simultaneously and introduce a class of dictionary learning methods based on joint sparse regularization.
2. To solve the resulting non-convex and non-smooth model, we design an optimization algorithm based on CCCP and Lagrangian dual method.
3. Motivated by the great success of multi-layer architecture in deep learning methods, we propose a joint sparse regularization dictionary learning approach with multi-layer structure and give the training algorithm at the same time.
4. Extensively experimental studies on publicly available datasets are performed to demonstrate the effectiveness of the proposed methods.

The rest of this paper is organized as follows. In "Related Works," we give related works. The section "Dictionary Learning with Joint Sparse Regularization" presents a class of joint sparse regularization and proposes the regularized dictionary learning model. Also, a tailored optimization algorithm based on CCCP and Lagrangian dual is designed. We give the joint sparse regularization dictionary learning with multi-layer structure and the corresponding algorithm in "Multi-Level JSDL." The section "Numerical Experiment" presents the detailed experimental results on several real-life datasets. The section "Conclusions" concludes this paper.

Related Works

In this section, we give a brief overview from both dictionary learning and structured sparsity regularization. Before we begin, we list notations to be used in this paper in Table 1.

Dictionary Learning

Suppose we have a collection of training samples $\mathcal{X} = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n\}$, where each sample $\mathbf{x}_i$ is represented by d features. Dictionary learning aims to find a dictionary $\mathbf{D} = [\mathbf{d}_1, \mathbf{d}_2, \cdots, \mathbf{d}_K] \in \Re^{d \times K}$ such that each sample $\mathbf{x}_i \in \mathcal{X}$ can be represented by a linear combination of $\{\mathbf{d}_i\}_{i=1}^{K}$, and most combination coefficients are 0 or close to 0. Let $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n] \in \Re^{d \times n}$ be the data

Table 1 The list of notations and their definitions in this paper

Symbol	Description
$\mathbf{x}$	A column vector in $\Re^n$
x_i	The ith entry of vector x
$\Vert\mathbf{x}\Vert_0$	The ℓ_0 of vector $\mathbf{x} \in \Re^n$, $\Vert\mathbf{x}\Vert_0 = \sum_{i=1}^{n} \vert x_i\vert^0$, where for a scalar x, $\vert x_i\vert^0 \triangleq \begin{cases} 1, & x_i \neq 0; \\ 0, & x_i = 0. \end{cases}$
$\Vert\mathbf{x}\Vert_2$	The Euclidean norm of vector $\mathbf{x} \in \Re^n$, $\Vert\mathbf{x}\Vert_2 = \sqrt{\sum_{i=1}^{n} x_i^2}$
$\Vert\mathbf{x}\Vert_p$	The ℓ_p norm of vector $\mathbf{x} \in \Re^n$, $\Vert\mathbf{x}\Vert_p = \left(\sum_{i=1}^{n} \vert x_i\vert^p\right)^{1/p}$
$\mathbf{X}$	A matrix in $\Re^{m\times n}$
$\mathbf{X}^{\mathrm{T}}$	The transpose of $\mathbf{X}$
X_{ij}	The entry at the ith row and the jth column of matrix $\mathbf{X}$
$\Vert\mathbf{X}\Vert_F$	The Frobenius norm of matrix $\mathbf{X} \in \Re^{m\times n}$, $\Vert\mathbf{X}\Vert_F = \sqrt{\sum_{i=1}^{m} \sum_{j=1}^{n} X_{ij}^2}$
$\Vert\mathbf{X}\Vert_{2,1}$	The $\ell_{2,1}$ norm of matrix $\mathbf{X} \in \Re^{m\times n}$, $\Vert\mathbf{X}\Vert_{2,1} = \sum_{i=1}^{m} \sqrt{\sum_{j=1}^{n} X_{ij}^2}$
$\mathrm{Tr}(\mathbf{X})$	The trace of squire matrix $\mathbf{X} \in \Re^{m\times m}$, $\mathrm{Tr}(\mathbf{X}) = \sum_{i=1}^{m} X_{ii}$
$\langle\mathbf{X}, \mathbf{Y}\rangle$	The Euclidean inner product between two same scale matrix $\mathbf{X}$ and $\mathbf{Y}$, $\langle\mathbf{X}, \mathbf{Y}\rangle = \sum_{i,j} X_{ij} Y_{ij}$
$\mathbf{I}$	The identity matrix of compatible size

matrix; dictionary learning can be typically formulated as the following optimization problem:

$$\min_{\mathbf{A},\mathbf{D}\in\mathcal{D}} loss(\mathbf{X}, \mathbf{D}\mathbf{A}) + \lambda\Omega(\mathbf{A}), \tag{1}$$

where the first term $loss(\cdot, \cdot)$ is the data fidelity term which encourages accurate representation, the second term is the sparse regularization term which enforces the sparsity on coefficient matrix $\mathbf{A}$, and λ is a positive regularization parameter which balances these two terms in the formulation. Noticing that the product between $\mathbf{D}$ and $\mathbf{A}$ is invariant to simultaneously scaling $\mathbf{D}$ by a scalar and $\mathbf{A}$ by its inverse. To prevent $\mathbf{D}$ from being arbitrarily large, which would lead to arbitrarily small values of $\mathbf{a}_i$, it is common to constrain its columns $(\mathbf{d}_1, \mathbf{d}_2, \cdots, \mathbf{d}_K)$ to have ℓ_2 norm less than or equal to 1. For ease of notation, we denote $\mathcal{D}$ the convex set of matrices verifying the constraint:

$$\mathcal{D} = \{\mathbf{D} \in \Re^{d\times K} : \|\mathbf{d}_j\|_2 \leq 1,\ \ j = 1, 2, \cdots, K\}. \tag{2}$$

In sparse representation, the coefficient with the fewest number of non-zero entries is highly appealing. Therefore, the most desirable sparse regularization is the ℓ_0, which counts the number of non-zero entries in the coefficient $\mathbf{a}_i$. In general, the ℓ_0 based dictionary learning model can be formulated as:

$$\min_{\mathbf{D},\mathbf{A}} \|\mathbf{X} - \mathbf{D}\mathbf{A}\|_F^2, \quad s.t. \quad \forall i,\ \ \|\mathbf{a}_i\|_0 \leq T_0, \tag{3}$$

where T_0 is a predefined positive constant, which controls the sparsity of $\mathbf{a}_i$. The above optimization would lead to a combinatorial optimization problem and has been proved to be NP-hard [7]. Therefore, approximate solutions are considered instead, and several efficient pursuit algorithms have been proposed. A straightforward way is to adopt a greedy strategy, such as the matching pursuit (MP) and orthogonal matching pursuit (OMP), which selects the dictionary atoms sequentially. K-SVD [1] updates the columns of dictionary with SVD sequentially, and the corresponding coefficients with any pursuit algorithm. As extensions of K-SVD, LC-KSVD [13] and discriminative K-SVD [52] introduced label information into the procedure of learning dictionaries. To address the noise in the observation data, Zhou et al. [55] proposed using the low-rank technique to capture the global structure and the ℓ_0 to promote the sparsity, and then improved the quality of learned dictionary.

To achieve sparsity, another alternative way is to employ the surrogate of the ℓ_0, which can be convex or non-convex. Among these surrogates, the most common one is the ℓ_1, which has been used in many works [5, 17, 24, 53, 54]. The corresponding models usually can be referred to as *basis pursuit* (BP) or LASSO. Mairal et al. [24] proposed an online dictionary learning model based on the ℓ_1, which can be handled efficiently by stochastic approximations methods. It is particularly suitable for processing the large-scale datasets with millions of training samples. To identify pedestrian effectively, Chen et al. [5] adopted hierarchical matching pursuit (HMP) to extract features and order preserving sparse coding (OPSC) for classification. Li et al. [17] combined non-negative matrix factorization and sparse coding to detect the overlapping community structure of the brain functional network.

Recently, Tariyal et al. [37] proposed the deep dictionary learning (DDL) model, which learned multiple levels dictionaries in a greedy fashion and has empirically demonstrated the better performance compared with the other methods. Based on DDL, several recent studies [20, 28, 36] have further developed algorithms varying from different applications, such as medical disease analysis, cross-modal retrieval, and silicone mask detection.

Joint Sparse Regularization

Joint sparse regularization has been playing an important role in many machine learning and data mining tasks, including feature selection [9], multi-task learning [2], structural learning [32], and multi-view learning [47]. From the sparsity perspective, although the $\ell_{2,0}$ is more desirable, it would result in a NP-hard problem. As a convex proxy of $\ell_{2,0}$, the $\ell_{2,1}$ has been frequently used as a sparse regularizer to enforce row sparsity in many works [11, 21, 31, 48]. Huang et al. [11] proposed to utilize the $\ell_{2,1}$ for sparse metric learning (SML), which is a generalization of the existing SML methods. Further, Ying et al. proposed [48] a smooth optimization approach for the $\ell_{2,1}$ regularized SML model.

To achieve better sparsity, Zhang et al. [51] proposed using $\ell_{2,p}$ with $0 < p < 1$ to perform feature selection. As stated in [51], experimental results demonstrate that $\ell_{2,p}(0 < p < 1)$ outperforms the $\ell_{2,1}$. In addition, the smaller p is, the better is the performance. However, due to the non-convexity and non-Lipschitz continuity of $\ell_{2,p}(0 < p < 1)$, it is very challenging to be solved. Most recently, we have proposed a novel sparse metric $\ell_{2,1-2}$ on matrices, which is non-convex yet Lipschitz continuous [35]. Although both the $\ell_{2,p}(0 < p < 1)$ and $\ell_{2,1-2}$ are the non-convex joint sparse regularization and have been successfully used in feature selection, compared with the $\ell_{2,p}(0 < p < 1)$, the $\ell_{2,1-2}$ is parameter-free and Lipschitz continuous, which leads to an easy optimization with global convergence.

Dictionary Learning with Joint Sparse Regularization

JSDL model

Currently, a variety of regularizations with joint sparsity have been proposed and studied in many works. To unify them, we give the $\ell_{r,p}(r > 0, p > 0)$ norm of a matrix $\mathbf{A} \in \Re^{K\times n}$, which is defined as the p norm of the vector containing of the r norm of the matrix rows. Mathematically, the formulation is as follows:

$$\|\mathbf{A}\|_{r,p} = \|(\|\mathbf{a}_1\|_r, \|\mathbf{a}_2\|_r, \cdots, \|\mathbf{a}_K\|_r)\|_p . \tag{4}$$

Figure 1 gives an illustration of computing the $\ell_{r,p}$-norm of matrix $\mathbf{A}$.

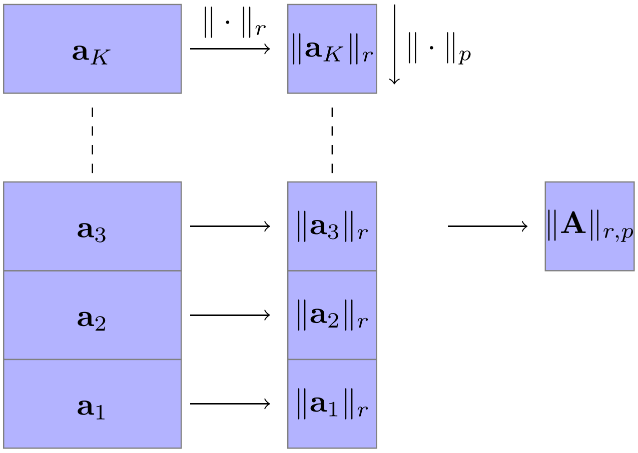

Fig. 1 Computation of the $\ell_{r,p}$-norm of matrix $\mathbf{A}$. $\mathbf{a}_i$ is the ith row of $\mathbf{A}$

Considering a linearly separable multi-class problem, we suppose the classifier is $\mathbf{A} \in \Re^{d\times c}$, where d and c are the number of features and the number of classes, respectively. In general, we would like to require most rows of $\mathbf{A}$ to be 0, which can be obtained by setting $0 < p \le 1$ in Eq. 4. Note that only the case of $p = 1$ is convex. The value of r depends on the kind of assumption between different classes, i.e., positive correlation and negative correlation. The positive correlation means that instances from different classes share as many identical features as possible. In this case, the range of r can be set to $1 < r < \infty$. In addition, increasing r corresponds to allowing more classes to share the same features. The other one is negative correlation which indicates that if a feature is important for one or few classes, it becomes less important for the other classes. This assumption can be implemented by setting $0 \le r \le 1$.

Considering the fact that $r = 2$ is the most commonly used among the choices of r, we give a more general form to measure the joint sparsity of the matrices.

Definition 1 For any matrix $\mathbf{A} = \left[\mathbf{a}_1^{\mathrm{T}}, \mathbf{a}_2^{\mathrm{T}}, \cdots, \mathbf{a}_K^{\mathrm{T}}\right]^{\mathrm{T}} \in \Re^{K\times n}$, let

$$\mathbf{a} = (\|\mathbf{a}_1\|_2, \|\mathbf{a}_2\|_2, \cdots, \|\mathbf{a}_K\|_2) \in \Re^K, \tag{5}$$

Then, the joint sparsity of $\mathbf{A}$ can be measured by:

$$\|\mathbf{A}\|_{2,\varphi} = \varphi(\mathbf{a}), \tag{6}$$

where $\varphi(\cdot)$ is a sparsity-promoting function on the vector.

Similar to $\ell_{r,p}$, we refer to the proposed joint sparse regularization as $\ell_{2,\varphi}$. Several sparsity-promoting functions on the vector are possible, including the ℓ_1 norm [38], $\ell_{1/2}$ [45], ℓ_{1-2} [23], MCP [49], and SCAD [8]. They all admit the sparse solution and have been proven to have very good performance. Noticing that, apart from the ℓ_1 norm, the rest are non-convex. It is worth mentioning that $\ell_{2,\varphi}$ becomes the famous $\ell_{2,1}$ norm when $\varphi(\cdot)$ is ℓ_1 norm, $\ell_{2,p}$ when $\varphi(\cdot)$ is ℓ_p norm, and $\ell_{2,1-2}$ when $\varphi(\cdot)$ is ℓ_{1-2}.

Because the joint sparse regularization imposes sparsity along with row of matrix, it can effectively remove the redundancy of dictionary. In this paper, we propose to use the regularization with joint sparsity for dictionary learning, which results in learning a compact dictionary. Different from the previous deep learning works which consider the sparsity of coefficient independently, we conduct the sparsity on all coefficients, that is the coefficient matrix, simultaneously. Thus we consider the following optimization problem:

$$\min_{\mathbf{D}\in\mathcal{D},\mathbf{A}} \frac{1}{2}\|\mathbf{X}-\mathbf{DA}\|_F^2 + \lambda\|\mathbf{A}\|_{2,\varphi}, \tag{7}$$

where $\|\mathbf{A}\|_{2,\varphi}$ is defined in Eq. 6, and λ is a positive regularization parameter which balances the tradeoff between reconstruction error and sparsity.

The Optimization Algorithm

As stated in “JSDL model,” several possible sparsity promoting functions can be used in our model (7). Recent years have witnessed a great success of non-convex regularization in many areas, such as dictionary learning [41], feature selection [35, 51], and remote sensing [40]. Since the algorithm strongly relies on the specific formulation of the sparsity-promoting function, we choose the non-convex ℓ_{1-2} to present the algorithm. The resulting joint sparse regularization $\ell_{2,1-2}$ has been proposed and studied in our previous work [35], which has shown good performance in feature selection. To be more specific, when the ℓ_{1-2} is used as the sparsity-promoting function, the dictionary learning with joint sparse regularization (7) becomes:

$$\min_{\mathbf{D}\in\mathcal{D},\mathbf{A}} \frac{1}{2}\|\mathbf{X}-\mathbf{DA}\|_F^2 + \lambda\|\mathbf{A}\|_{2,1-2} \tag{8}$$

where $\mathcal{D}$ is defined as in Eq. 2. To solve the above problem, following [1], we adopt a two-step iterative scheme, where the sparse approximations are found with the dictionary fixed and the dictionary is subsequently optimized based on the current sparse coefficients.

Updating Dictionary D

Keeping coefficient matrix $\mathbf{A}$ fixed, optimization problem becomes:

$$\min_{\mathbf{D}} \frac{1}{2}\|\mathbf{X}-\mathbf{DA}\|_F^2, \ s.t. \ \|\mathbf{d}_j\|_2 \le 1, \ j=1,2,\cdots,K, \tag{9}$$

which is a quadratically constrained quadratic program (QCQP). It can be effectively solved by the Lagrange dual method. Noticing that $\forall j$, the quadratic constraint $\|\mathbf{d}_j\|_2 \le 1$ is equivalent to $\|\mathbf{d}_j\|_2^2 \le 1$. To derive the dual problem, we consider the Lagrangian of problem (9):

$$\mathcal{L}(\mathbf{D},\boldsymbol{\lambda}) = \frac{1}{2}\|\mathbf{X}-\mathbf{DA}\|_F^2 + \sum_{j=1}^{K}\lambda_j\left(\|\mathbf{d}_j\|_2^2 - 1\right), \tag{10}$$

where $\lambda = (\lambda_1, \lambda_2, \cdots, \lambda_K)$ and λ_j is the Lagrange multiplier corresponding to the jth inequality constraint $\|\mathbf{d}_j\|_2^2 \le 1$. Then, the dual problem of Eq. 9 can be formulated as:

$$\max_{\boldsymbol{\lambda}} \inf_{\mathbf{D}\mathcal{L}}(\mathbf{D},\boldsymbol{\lambda}), \quad s.t. \quad \lambda_j \ge 0, \ \forall j = 1,2,\cdots,K. \tag{11}$$

Since $\mathcal{L}(\mathbf{D},\boldsymbol{\lambda})$ is strictly convex with respect to $\mathbf{D}$, the solution of $\inf_{\mathbf{D}\mathcal{L}}(\mathbf{D},\boldsymbol{\lambda})$ must satisfy:

$$\nabla_{\mathbf{D}\mathcal{L}}(\mathbf{D},\boldsymbol{\lambda}) = (\mathbf{DA}-\mathbf{X})\mathbf{A}^{\mathrm{T}} + 2\mathbf{D}\boldsymbol{\Lambda} = \mathbf{0}, \tag{12}$$

from which, we have:

$$\mathbf{D} = \mathbf{XA}^{\mathrm{T}}(\mathbf{AA}^{\mathrm{T}} + 2\boldsymbol{\Lambda})^{-1}, \tag{13}$$

where $\boldsymbol{\Lambda} = diag(\lambda_1, \lambda_2, \cdots, \lambda_K) \in \Re^{K\times K}$. Substituting Eq. 13 into problem (11), we have the Lagrangian dual problem:

$$\begin{aligned} &\min_{\boldsymbol{\Lambda}} \ \mathrm{Tr}\left(\mathbf{XA}^{\mathrm{T}}(\mathbf{AA}^{\mathrm{T}}+2\boldsymbol{\Lambda})^{-1}\mathbf{AX}^{\mathrm{T}}\right) + \mathrm{Tr}(\boldsymbol{\Lambda}) \\ &s.t. \, \forall j, \ \lambda_j \ge 0. \end{aligned} \tag{14}$$

To solve the above problem, we implement the algorithm described as in [16]. Once the optimal of Lagrangian dual problem (14) is obtained, according to Eq. 13, we can get the optimal dictionary. Note that there are $K \times n$ variables to be optimized in primal problem (9), while only K ones in dual problem (14). Thus, the Lagrange dual method can significantly reduce the computational cost.

Updating Coefficient Matrix A

With dictionary $\mathbf{D}$ fixed, optimization problem is reduced to:

$$\min_{\mathbf{A}} \frac{1}{2}\|\mathbf{X}-\mathbf{DA}\|_F^2 + \lambda\|\mathbf{A}\|_{2,1-2}, \tag{15}$$

which is non-convex and in general difficult to be solved. As an import technique in non-convex optimization, the concave-convex procedure (CCCP) has been extensively used and studied in many machine learning tasks. To be specific, CCCP is a majorization-minimization (MM) algorithm that solves d.c. (difference of convex functions) programs as a sequence of convex programs. Considering the popularity and global convergence demonstrated in our previous work [35], we use CCCP to solve the subproblem (15). Denote by $F(\mathbf{A})$ the objective function in Eq. 15, we find:

$$F(\mathbf{A}) = \underbrace{\left(\frac{1}{2}\|\mathbf{X}-\mathbf{DA}\|_F^2 + \lambda\|\mathbf{A}\|_{2,1}\right)}_{convex} - \underbrace{\lambda\|\mathbf{A}\|_F}_{convex}. \tag{16}$$

As seen from Eq. 16, $F(\mathbf{A})$ can be expressed as the difference of the two convex functions. According to CCCP, the linearized subproblem at the kth step is given by:

$$\min_{\mathbf{A}} \left(\frac{1}{2}\|\mathbf{X}-\mathbf{D}\mathbf{A}\|_F^2 + \lambda\|\mathbf{A}\|_{2,1} \right) - \lambda\langle \mathbf{A}, \widehat{\mathbf{A}^k} \rangle, \tag{17}$$

where $\widehat{\mathbf{A}^k} \in \partial\|\mathbf{A}\|_F$. $\partial\|\mathbf{A}\|_F$ is the subdifferential of $\|\cdot\|_F$ at point $\mathbf{A}$ and given by:

$$\partial\|\mathbf{A}\|_F = \begin{cases} \{\frac{\mathbf{A}}{\|\mathbf{A}\|_F}\}, & \text{if } \mathbf{A} \neq \mathbf{0}; \\ \{\mathbf{M} : \|\mathbf{M}\|_F \leq 1\} & \text{otherwise} \end{cases} \tag{18}$$

For ease of computation, in this paper, we set $\widehat{\mathbf{A}^k} = \mathbf{0}$, when $\mathbf{A}^k = \mathbf{0}$ and obtain:

$$\widehat{\mathbf{A}^k} = \begin{cases} \mathbf{0}, & \mathbf{A}^k = \mathbf{0}; \\ \frac{\mathbf{A}^k}{\|\mathbf{A}^k\|_F}, & \mathbf{A}^k \neq \mathbf{0} \end{cases} \tag{19}$$

We summarize the process of CCCP for solving the coefficient $\mathbf{A}$ in Algorithm 1.

Algorithm 1 CCCP for solving problem (15).

```
Input: data matrix $\mathbf{X}$, dictionary $\mathbf{D}$ and regularization parameter $\lambda$
repeat
    Calculate $\widehat{\mathbf{A}^k}$ by Eq. 19
    $\mathbf{A}^{k+1} := \arg\min_{\mathbf{A}} \frac{1}{2}\|\mathbf{X}-\mathbf{D}\mathbf{A}\|_F^2+\lambda\|\mathbf{A}\|_{2,1}-\lambda\langle \mathbf{A}, \widehat{\mathbf{A}^k}\rangle$
    $k := k+1$
until CCCP stopping criterion is satisfied
Output: Converged $\mathbf{A}^*$
```

Next, we discuss how to optimize the CCCP linearized subproblem (17). Recall the subproblem:

$$\min_{\mathbf{A}} \frac{1}{2}\|\mathbf{X}-\mathbf{D}\mathbf{A}\|_F^2 + \lambda\|\mathbf{A}\|_{2,1} - \lambda\langle \mathbf{A}, \widehat{\mathbf{A}^k} \rangle, \tag{20}$$

which is an unconstrained convex optimization. Noticing that in the case that some rows of matrix $\mathbf{A}$ are all with zero entries, the term $\|\mathbf{A}\|_{2,1}$ in Eq. 20 is not differentiable. Thus, the gradient-based approaches cannot be directly applied. Here, we use ADMM [4] to solve (20), which is a convex optimization algorithm dating back to the early 1980s. It has attracted attention due to its applicability to various machine learning and image processing problems. To this end, we introduce an auxiliary variable $\mathbf{U}$ and set $\mathbf{U} = \mathbf{A}$, then reformulate problem (20) as the following equality-constrained problem with two parts of variables $\mathbf{A}$ and $\mathbf{U}$:

$$\begin{aligned} &\min_{\mathbf{A},\mathbf{U}} \frac{1}{2}\|\mathbf{X}-\mathbf{D}\mathbf{A}\|_F^2 + \lambda\|\mathbf{U}\|_{2,1} - \lambda\langle \mathbf{A}, \widehat{\mathbf{A}^k} \rangle \\ &s.t.\ \mathbf{U} = \mathbf{A}. \end{aligned} \tag{21}$$

The augmented Lagrangian for problem (21) is formulated as:

$$\begin{aligned} L_\rho(\mathbf{A}, \mathbf{U}) = {} & \frac{1}{2}\|\mathbf{X}-\mathbf{D}\mathbf{A}\|_F^2 + \lambda\|\mathbf{U}\|_{2,1} - \lambda\langle \mathbf{A}, \widehat{\mathbf{A}^k} \rangle \\ & +\langle \mathbf{\Sigma}, \mathbf{A}-\mathbf{U} \rangle + \frac{\rho}{2}\|\mathbf{A}-\mathbf{U}\|_F^2, \end{aligned} \tag{22}$$

where $\mathbf{\Sigma} \in \Re^{K\times n}$ is the Lagrangian multiplier matrix corresponding to the equality constraint $\mathbf{U} = \mathbf{A}$ and $\rho > 0$ is a penalty parameter that determines the penalty for infeasibility of the equality constraint. According to [4], ADMM consists of the following iterations:

$$\mathbf{A}^{k+1} : \; = \arg\min_{\mathbf{A}} L_\rho\left(\mathbf{A}, \mathbf{U}^k\right); \tag{23a}$$

$$\mathbf{U}^{k+1} : \; = \arg\min_{\mathbf{U}} L_\rho\left(\mathbf{A}^{k+1}, \mathbf{U}\right); \tag{23b}$$

$$\mathbf{\Sigma}^{k+1} : \; = \mathbf{\Sigma}^k + \rho\left(\mathbf{A}^{k+1} - \mathbf{U}^{k+1}\right), \tag{23c}$$

where k is the iteration index. We describe the details of Eqs. 23a and 23b in the follow-up of this part. To simplify the notation, we omit the superscripts in Eqs. 23a and 23b.

The A-Update When fixing $\mathbf{U}$ and $\mathbf{\Sigma}$ in Eq. 23a, the problem is reduced to:

$$\min_{\mathbf{A}} \frac{1}{2}\|\mathbf{X}-\mathbf{D}\mathbf{A}\|_F^2 - \lambda\langle \mathbf{A}, \widehat{\mathbf{A}^k} \rangle + \langle \mathbf{\Sigma}, \mathbf{A}-\mathbf{U} \rangle + \frac{\rho}{2}\|\mathbf{A}-\mathbf{U}\|_F^2 \tag{24}$$

It can be verified that the objective function in Eq. 24 is convex and differentiable with respect to $\mathbf{A}$, and thus the optimal solution of Eq. 24 is just the stationary point of the objective function. Setting the derivative of the objective function in problem (24) to be 0, we have:

$$\left(\mathbf{D}^{\mathrm{T}}\mathbf{D} + \rho\mathbf{I}\right)\mathbf{A} = \mathbf{D}^{\mathrm{T}}\mathbf{X} + \lambda\widehat{\mathbf{A}^k} - \mathbf{\Sigma} + \rho\mathbf{U}, \tag{25}$$

where $\mathbf{I} \in \Re^{K\times K}$ is the identity matrix. As we know, the matrix $\mathbf{D}^{\mathrm{T}}\mathbf{D} + \rho\mathbf{I}(p > 0)$ is positive definite; its inverse exists. Thus, we have:

$$\mathbf{A} = \left(\mathbf{D}^{\mathrm{T}}\mathbf{D} + \rho\mathbf{I}\right)^{-1}\left(\mathbf{D}^{\mathrm{T}}\mathbf{X} + \lambda\widehat{\mathbf{A}^k} - \mathbf{\Sigma} + \rho\mathbf{U}\right) \tag{26}$$

In the following, we give a theorem which can be used for solving the subproblem w.r.t $\mathbf{U}$.

Theorem 1 *Given a positive scalar* λ *and* $\mathbf{a}, \mathbf{b} \in \Re^n$*, the optimal solution of*

$$\min_{\mathbf{w}\in\Re^n} f(\mathbf{w}) = \frac{1}{2}\|\mathbf{w}-\mathbf{a}\|_2^2 + \lambda\|\mathbf{w}\|_2 - \mathbf{w}^{\mathrm{T}}\mathbf{b} \tag{27}$$

is

$$\mathbf{w}^* = \begin{cases} (1 - \frac{\lambda}{\|\mathbf{a}+\mathbf{b}\|_2})(\mathbf{a}+\mathbf{b}), & \textit{if } \|\mathbf{a}+\mathbf{b}\|_2 > \lambda \\ 0, & \textit{otherwise} \end{cases} \tag{28}$$

Proof Since problem (27) is an unconstrained convex problem, we know that its optimal solution $\mathbf{w}^*$ will satisfy the optimal condition that is $0 \in \partial f(\mathbf{w}^*)$. Note that when $\mathbf{w} \neq 0$:

$$\partial f(\mathbf{w}) = \nabla f(\mathbf{w}) = \lambda \frac{\mathbf{w}}{\|\mathbf{w}\|_2} + \mathbf{w} - \mathbf{a} - \mathbf{b} \tag{29}$$

we know that if $\mathbf{w}^* \neq 0$,

$$\lambda \frac{\mathbf{w}^*}{\|\mathbf{w}^*\|_2} + \mathbf{w}^* - \mathbf{a} - \mathbf{b} = 0, i.e., \mathbf{a}+\mathbf{b}-\mathbf{w}^* = \lambda \frac{\mathbf{w}^*}{\|\mathbf{w}^*\|_2} \tag{30}$$

from which we obtain the theorem. □

The U-Update Similarly, optimization problem (23b) with respect to $\mathbf{U}$ can be simplified as:

$$\min_{\mathbf{U}} \|\mathbf{U}\|_{2,1} + \langle \mathbf{\Sigma}, \mathbf{A} - \mathbf{U} \rangle + \frac{\rho}{2} \|\mathbf{A} - \mathbf{U}\|_F^2. \tag{31}$$

Expanding the objective function in problem (31) and removing the terms that are irrelevant of $\mathbf{U}$, let $\mathbf{N} = \mathbf{A} + \mathbf{\Sigma}/\mu$, we arrive at:

$$\min_{\mathbf{U}} \frac{1}{2} \|\mathbf{U} - \mathbf{N}\|_F^2 + \frac{\lambda}{\mu} \|\mathbf{U}\|_{2,1}, \tag{32}$$

which can be rewritten into the following vector-wise form

$$\min_{\mathbf{u}_i} \sum_{i=1}^{K} \left(\frac{1}{2} \|\mathbf{u}_i - \mathbf{n}_i\|_2^2 + \frac{\lambda}{\mu} \|\mathbf{u}_i\|_2 \right), \tag{33}$$

where $\mathbf{u}_i$ and $\mathbf{n}_i$ are the ith row of $\mathbf{U}$ and $\mathbf{N}$, respectively. Furthermore, we have:

$$\sum_{i=1}^{K} \min_{\mathbf{u}_i} \left(\frac{1}{2} \|\mathbf{u}_i - \mathbf{n}_i\|_2^2 + \frac{\lambda}{\mu} \|\mathbf{u}_i\|_2 \right). \tag{34}$$

Then, the optimal solution can be obtained by simultaneously solving d independent subproblems. Using Theorem 1, the optimal solution to the ith subproblem is easily presented as:

$$\mathbf{u}_i = \begin{cases} \frac{\|\mathbf{n}_i\|_2 - \alpha}{\|\mathbf{n}_i\|_2} \mathbf{n}_i, & \text{if } \alpha < \|\mathbf{n}_i\|_2 \\ 0 & \text{otherwise,} \end{cases} \tag{35}$$

where $\mathbf{N} = \mathbf{A} + \mathbf{\Sigma}/\mu$ and $\alpha = \lambda/\mu$.

With the above updating rules, we summarize our ADMM in Algorithm 2.

Algorithm 2 ADMM for solving problem (20).

```
Input: data matrix $\mathbf{X}$ and $\mathbf{D}$
Initialization $\mathbf{A}^0$, $k = 0$
repeat
    $\mathbf{A}^{k+1} := \left(\mathbf{D}^{\mathrm{T}}\mathbf{D} + \rho\mathbf{I}\right)^{-1} (\mathbf{D}^{\mathrm{T}}\mathbf{X} + \lambda\widehat{\mathbf{A}^k} - \mathbf{\Sigma}^k + \rho\mathbf{U}^k)$
    $\mathbf{N}^k = \mathbf{A}^{k+1} + \mathbf{\Sigma}^k/\mu$
    for $i = 1, 2, \cdots, d$ do
        if $\alpha < \rho\|\mathbf{n}_i^k\|_2$ then
            $\mathbf{u}_i^{k+1} = \left(1 - \frac{\lambda}{\rho\|\mathbf{n}_i^k\|_2}\right)\mathbf{n}_i^k$
        else
            $\mathbf{u}_i^{k+1} = \mathbf{0}$
        end if
    end for
    $\mathbf{\Sigma}^{k+1} := \mathbf{\Sigma}^k + \rho(\mathbf{A}^{k+1} - \mathbf{U}^{k+1})$
    $k = k + 1$
until ADMM stopping criterion is satisfied
Output: Converged $\mathbf{A}^*$
```

Complexity Analysis

To more clearly analyze the computational complexity, we revisit the symbolic representation of some scalars. K and n are the numbers of atoms in dictionary and instances, respectively. The computation comes from two aspects, i.e., updating $\mathbf{A}$ and $\mathbf{D}$. To update $\mathbf{A}$, we need to compute $\left(\mathbf{D}^{\mathrm{T}}\mathbf{D} + \rho\mathbf{I}\right)^{-1}$, which requires $\mathcal{O}(K^3)$ operations. However, since the inverse is only related to $\mathbf{D}$, which is a constant in Algorithm 1. Another main cost is the matrix multiplication, which needs $\mathcal{O}(nN_1N_2K^2)$, where N_1 and N_2 are the numbers of iterations CCCP and ADMM, respectively. To update $\mathbf{D}$, the main cost is to compute $(\mathbf{A}\mathbf{A}^{\mathrm{T}} + 2\mathbf{\Lambda})^{-1}$, which requires $\mathcal{O}(K^3)$ operations. Then, the overall cost is $\mathcal{O}(K^3 + nN_1N_2K^2)$.

Multi-Level JSDL

Before starting multi-level JSDL, we first show the simpler neural network architecture with the four layers in Fig. 2. Noticing that, unlike a neural network which is directed from the input to the representation, the dictionary learning kind of network points in the other direction from representation to the input. In multi-level dictionary learning, instead of interpreting the columns as atoms, we can think of them as connections between the input and the representation layer. As seen from the figure, $\mathbf{D}_i$ is the

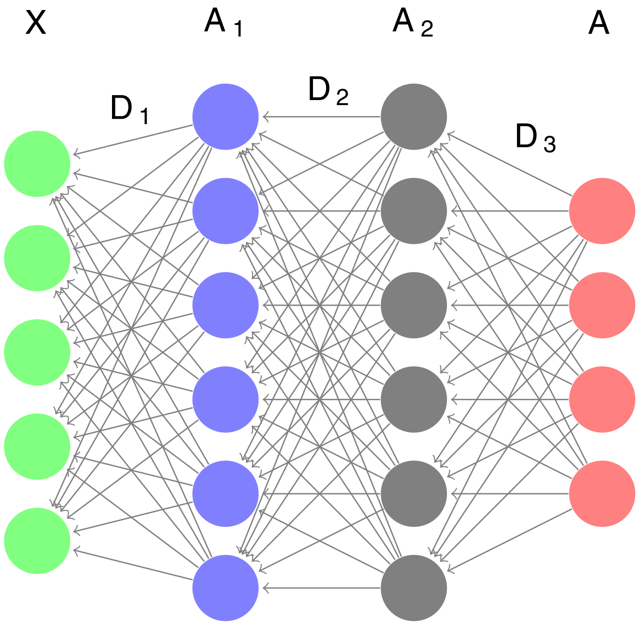

Fig. 2 An illustration of 4-layer dictionary learning

dictionary at the i-layer and can be viewed as a mapping between representation $\mathbf{A}$ and data matrix $\mathbf{X}$ containing complex hierarchical and structural information. In the network, it is natural to expect the following equations hold:

$$\begin{cases} \mathbf{A}_2 = (\text{or} \approx)\mathbf{D}_3\mathbf{A}, & \text{for the third layer;} \quad (36a) \\ \mathbf{A}_1 = (\text{or} \approx)\mathbf{D}_2\psi(\mathbf{A}_2), & \text{for the second layer;} \quad (36b) \\ \mathbf{X} = (\text{or} \approx)\mathbf{D}_1\psi(\mathbf{A}_1), & \text{for the first layer,} \quad (36c) \end{cases}$$

where $\psi(\cdot)$ is an activation function at each hidden layer. Combining Eqs. 36a, 36b, and 36c, we have:

$$\mathbf{X} = (\text{or} \approx)\mathbf{D}_1\psi(\mathbf{D}_2\psi(\mathbf{D}_3\mathbf{A})) \tag{37}$$

For the case of the network with 4 layers, according to Eq. 37, we give the loss function:

$$loss(\mathbf{D}_1, \mathbf{D}_2, \mathbf{D}_3, \mathbf{A}) = \frac{1}{2}\,\|\mathbf{X} - \mathbf{D}_1\psi\,(\mathbf{D}_2\psi\,(\mathbf{D}_3\mathbf{A}))\|_F^2 \tag{38}$$

Furthermore, we can easily extend the the four-layer dictionary learning to the scenario of $m(m > 4)$ layers. Similarly, we derive the overall loss function over the m-layer as follows:

$$loss(\mathbf{D}_1, \cdots, \mathbf{D}_m, \mathbf{A}) = \frac{1}{2}\,\|\mathbf{X} - \mathbf{D}_1\psi\,(\mathbf{D}_2\psi\,(\cdots\psi\,(\mathbf{D}_m\mathbf{A})))\|_F^2\,, \tag{39}$$

By combing the regularization term $\ell_{2,1-2}$, with the loss function in Eq. 39, the multi-layer dictionary learning model can be formulated as:

$$\min_{\mathbf{D}_1,\cdots,\mathbf{D}_m,\mathbf{A}} loss(\mathbf{D}_1, \cdots, \mathbf{D}_m, \mathbf{A}) + \lambda\|\mathbf{A}\|_{2,1-2}. \tag{40}$$

As seen from Eq. 40, the problem is highly non-convex with $m+1$ parts of variables, i.e., m dictionaries and 1 coefficient matrix, which is usually difficult to be solved globally. We adopt a greedy approach, which was introduced by Hinton et al. [10] for unsupervised learning algorithm for deep belief networks (DBN). This training strategy for such networks was subsequently analyzed by Bengio et al. [3] who concluded that it is an important ingredient in effective optimization and training of deep networks. To this end, we learn a standard dictionary and coefficients at the first layer, and in subsequent layers the coefficients from the previous layer act as inputs for dictionary learning. We summarize the training process of multi-layer JSDL in Algorithm 3.

Algorithm 3 Greedy layer-wise training M-JSDL.

```
Input: training data matrix $\mathbf{X}_{train}$
Initialization $\mathbf{D}_1, \mathbf{D}_2, \cdots, \mathbf{D}_m$
For the first layer
    $\{\mathbf{A}_1^*, \mathbf{D}_1^*\} = \arg\min_{\mathbf{D},\mathbf{A}} \frac{1}{2}\|\mathbf{X}_{train} - \mathbf{D}\mathbf{A}\|_F^2$
From the second layer to penultimate level
    $\{\mathbf{A}_i^*, \mathbf{D}_i^*\} = \arg\min_{\mathbf{D},\mathbf{A}} \frac{1}{2}\|\psi^{-1}(\mathbf{A}_{i-1}^*) - \mathbf{D}\mathbf{A}\|_F^2$
For the last layer
    $\{\mathbf{A}^*, \mathbf{D}_m^*\} = \arg\min_{\mathbf{D},\mathbf{A}} \frac{1}{2}\|\psi^{-1}(\mathbf{A}_{m-1}^*) - \mathbf{D}\mathbf{A}\|_F^2 + \lambda\|\mathbf{A}\|_{2,1-2}$
Output: $\mathbf{D}_1^*, \mathbf{D}_2^*, \cdots, \mathbf{D}_m^*, \mathbf{A}^*$
```

Algorithm 4 Testing algorithm with M-JSDL.

```
Input: testing data $\mathbf{X}_{test}$, dictionary $\{\mathbf{D}_1^*, \cdots, \mathbf{D}_m^*\}$ and the
    regularization parameter $\lambda^*$
For the first layer
    $\mathbf{A}_1^* = \arg\min_{\mathbf{A}} \frac{1}{2}\|\mathbf{X}_{test} - \mathbf{D}_1^*\mathbf{A}\|_F^2$
From the second layer to penultimate level
    $\mathbf{A}_i^* = \arg\min_{\mathbf{A}} \frac{1}{2}\|\psi^{-1}\left(\mathbf{A}_{i-1}^*\right) - \mathbf{D}_i^*\mathbf{A}\|_F^2$
For the last layer
    $\mathbf{A}^* = \arg\min_{\mathbf{A}} \frac{1}{2}\|\psi^{-1}\left(\mathbf{A}_{m-1}^*\right) - \mathbf{D}_m^*\mathbf{A}\|_F^2 + \lambda^*\|\mathbf{A}\|_{2,1-2}$
Output: $\mathbf{A}^*$
```

As observed in Algorithm 3, during the training process of multi-layer JSDL, we only need to solve the following optimization problem at each layer:

$$\min_{\mathbf{D},\mathbf{A}} \frac{1}{2}\|\mathbf{Y} - \mathbf{D}\mathbf{A}\|_F^2 + \lambda\|\mathbf{A}\|_{2,1-2}, \tag{41}$$

where $\mathbf{Y}$ is independent of the current layer and can be viewed as a constant. In fact, problem (41) is a shallow JSDL model, whose optimization has been in detail presented in "The Optimization Algorithm." Once we get the optimal dictionary $\mathbf{D}_i^*$ at the i-layer, the sparse representation of the test samples can be obtained by Algorithm 4.

Numerical Experiment

In this section, we conduct experiments on publicly available datasets to evaluate the performance of the proposed methods. We first compare our proposed JSDL with several state-of-the-art shallow algorithms. Then, we compare M-JSDL with the deep dictionary learning method, deep belief network (DBN), and stacked autoassociators (SAA) models.

Dataset Descriptions

The MNIST dataset is composed of gray-scale images of ten handwritten digits. Each image is 28 × 28 pixels. Based on MNIST, Larochelle et al. [15] developed four variations[1], i.e., *basic-rotated*, *background-random*, *background-image*, and *background-image-rotated*. They can be generated by adding different variational factors, which are rotation angle between 0 and 2π radians, random background pixels between 0 and 255, random patch from 20 black and white images, and factors of *mnist-rot* and *mnist-back-image*, respectively. These four databases have 10,000, 2000, and 50,000 instances in their training, validation, and test sets, respectively. Figure 3 shows some samples from these databases.

The extended Yale B database contains 2,414 images of 38 human frontal faces under 64 illumination conditions and expressions. There are about 64 images for each person. The original images were cropped to 192 × 168 pixels. Following [43], we utilize a random matrix to project each face image into a 504-dimensional feature vector.

The AR face dataset consists of over 4,000 color images from 126 individuals. For each individual, 26 pictures were taken in two separate sessions. Following the standard evaluation procedure from [43], we only used a subset consisting of 2,600 images from 50 male and 50 female subjects. Each image is cropped to 165 × 120 and then projected into a 540-dimensional feature vector.

Classification Comparison

To begin, we choose the seven state-of-the-art algorithms as compared methods of JSDL. They are two traditional classifiers, including nearest neighbor (NN) and support vector machine (SVM), and four dictionary learning methods, including SDL [25], SRC [43], D-KSVD [52], and LC-KSVD [13]. For SVM, we consider two kernels, i.e., Gaussian kernel (SVM$_{rbf}$) and polynomial kernel (SVM$_{poly}$).

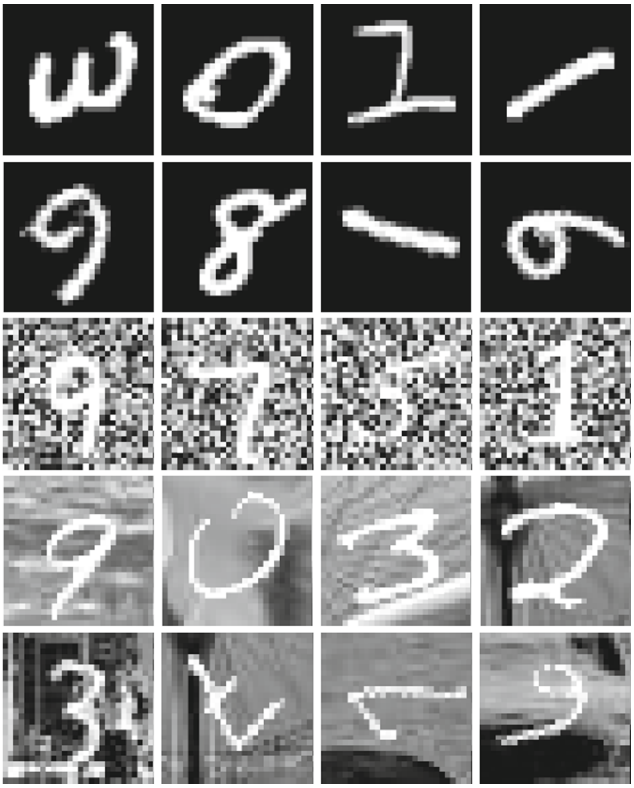
Fig. 3 Samples from the MNIST database and extended MNIST databases. From top to bottom, samples are from *basic*, *basic-rotated*, *background-random*, *background-image*, and *background-image-rotated*

For all the methods, the optimal parameters corresponding to the best classification performances can be obtained by 5-fold cross validation on the training set. For SVM with Gaussian kernel, we adopt a two-stage grid search for the parameters width of the kernel and the soft-margin. We first search through a coarse logarithmic grid ranging from $\sigma = 10^{-7}$ to 1 and $C = 0.1$ to 10^5. Then, we perform a more fine-grained search in the vicinity of tuple that gives the best performance. In the case of SVM with the polynomial kernel, we tune the degree of the polynomial from $d = 1$ to 20. In k-nearest neighbor classification algorithm, we simply set $k = 1$.

For the JSDL, the regularization parameter is from the range of $\{0.001, 0.01, 0.1, 1, 10\}$. We have also observed that a larger dictionary usually means a better performance, but a higher computational cost. Setting the size of the dictionary is therefore often a tradeoff between the quality of results and efficiency. In our experiments, we vary the values in $\{50, 100, 200, 300, 400\}$. Once the sparse

[1] http://www.iro.umontreal.ca/~lisa/twiki/bin/view.cgi/Public/DeepVsShallowComparisonICML2007

Table 2 The classification accuracy of different methods on the six datasets

Dataset	Acc.(%)							
	SVM$_{ploy}$	SVM$_{rbf}$	NN	SDL	SRC	LC-KSVD	D-KSVD	JSDL
MNIST	93.32	94.06	95.31	**98.95**	**96.55**	93.30	93.60	96.21
mnist–basic	96.31	**96.97**	94.81	95.14	**97.01**	92.70	92.20	96.70
mnist–rotated	86.39	**89.62**	82.33	51.98	81.22	48.66	50.01	**88.22**
mnist–background–random	83.38	85.42	83.24	**88.23**	84.21	87.70	87.70	**88.65**
mnist–background–image	75.99	77.39	73.61	80.92	78.91	80.65	**81.20**	**81.54**
mnist-rotated-background-image	62.71	67.38	65.57	74.31	66.71	**75.40**	**75.40**	**76.22**
AR	94.00	94.71	89.70	95.25	94.70	94.56	88.18	**96.87**
Extended Yale B	92.35	94.01	90.70	94.51	96.10	94.48	93.58	**97.31**
Average rank	5.6667	3.6667	5.8333	**3.333**	3.8333	5.3333	4.6667	**1.8333**

The top two ranks are highlighted in boldface

representations of the input data are learned by these methods, SVM is used for classification. We implement SVM with the LIBSVM[2] package.

The classification performances for the different learning algorithms on the different datasets are reported in Table 2. The last row in the table presents the average rank of each algorithm over all datasets, which is based on its classification accuracy. The top two ranks corresponding to the performances are highlighted in boldface. As seen from the table, dictionary learning based methods outperform traditional classification algorithms, which demonstrates effectiveness of the sparse representation. Furthermore, among dictionary learning methods, our proposed method is able to obtain the best performance on most of datasets, except two datasets. Also, the cases of the datasets MNIST and mnist-basic are slightly worse than the best ones.

To further verify the statistical significance of JSDL, we conduct the Wilcoxon signed-rank test, which is a pairwise test to detect significant differences between two methods. Table 3 reports the R^+, R^-, and p value of all the pairwise comparisons concerning JSDL. As seen from this table, JSDL can achieve a significant improvement over all the benchmark methods except SDL with a level of significance $\alpha = 0.05$.

To test the performance of M-JSDL, we compare it with deep dictionary learning [37], which used the ℓ_1 norm as sparse regularization. In the experiment, following [37], we consider the architecture with up to three layers. Additionally, we set the dictionary size to be 300, 150, and 50 for the first, second, and last layers, respectively. For DDL and our proposed M-JSDL, we use a linear activation function on all the datasets. In addition, we compare it with DBN and SAA models. The results are summarized in Table 4. From the table, we can see that compared with baselines, our method can achieve the better performance in many cases.

Moreover, we list the classification results with different layers in Table 5. As shown in the table, the results in the second column are the accuracies for the dictionary learning with shallow architecture. And the results from the third column to the last column present the accuracies for deep architecture corresponding to different layers. The results show that for all the datasets, dictionary learning with multi-layer architecture indeed improves the performance of the shallow dictionary learning. The improvement is possibly due to more abstract representation by learning from multi-layer. Depending on the complexity of the dataset, the difference in performance varies from 1.7% to more than 5%.

Parameter Analysis

As we know, the proposed model JSDL has two parameters, i.e., the dictionary size K and the regularization parameter

Table 3 Wilcoxon signed ranks test results ($\alpha = 0.05$)

Comparison	R^+	R^-	p value
JSDL vs. SVM$_{ploy}$	36.00	0.00	0.0078
JSDL vs. SVM$_{rbf}$	33.00	3.00	0.0391
JSDL vs. NN	36.00	0.00	0.0078
JSDL vs. SDL	30.00	6.00	0.1094
JSDL vs. SRC	33.00	3.00	0.0391
JSDL vs. LC-KSVD	36.00	0.00	0.0078
JSDL vs. D-KSVD	36.00	0.00	0.0078

[2] http://www.csie.ntu.edu.tw/~cjlin/libsvm/

Table 4 Comparison of M-JSDL with other models with a multi-layer structure

Dataset	Acc.(%)			
	DDL	DBN	SAA	M-JSDL
MNIST	98.64	98.53	96.72	**98.67**
mnist–basic	**97.28**	96.06	96.54	97.01
mnist–rotated	90.34	87.89	88.57	**91.21**
mnist–background–random	**92.38**	90.20	88.72	90.41
mnist–background–image	86.17	83.85	77.00	**87.06**
mnist–rotated-background–image	63.85	68.16	**75.91**	65.37

The best performance is highlighted in boldface

Table 5 The classification accuracy of M-JSDL with different layers

Dataset	Acc.(%)			
	Shallow (50)	1-layer (300)	2-layer (300-150)	3-layer (300-150-50)
MNIST	94.31	94.64	95.97	96.01
mnist–basic	93.22	96.76	97.03	98.72
mnist–rotated	85.53	85.98	86.01	87.34
mnist–background–random	84.78	84.21	85.46	87.12
mnist–background–image	90.10	90.67	91.22	94.56
mnist–rotated-background–image	73.65	74.34	74.92	76.03

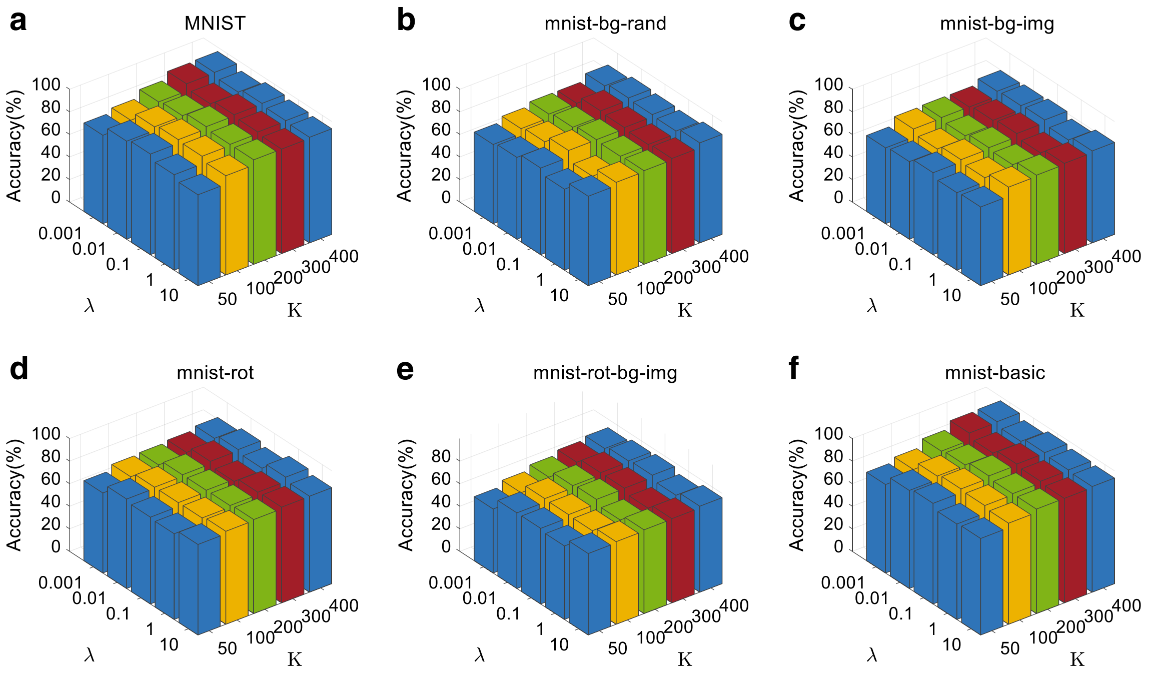

Fig. 4 Performance variation of the proposed method w.r.t. different values of the parameters λ and K. **a** MNIST; **b** mnist-bg-rand(mnist-background-random); **c** mnist-bg-image(mnist-background-image); **d** mnist-rot(mnist-rotated); **e** mnist-rot-bg-img(mnist-rotated-background-image); **f** mnist-basic

λ, which need to be set in advance and have important effects on the performance of the algorithm. In this subsection, we examine how the performance of the proposed JSDL changes with the variation of K and λ. To be specific, these two parameters are tuned from $\{50, 100, 200, 300, 400\} \times \{0.001, 0.01, 0.1, 1, 10\}$. The results on the six datasets are presented in Fig. 4. It can be observed that the classification accuracies increase as the dictionary size K increases. This observation validates that the better performance of dictionary learning methods can be obtained by a larger dictionary. At the same time, we can see from the figure, comparing with the value of regularization parameter λ, our method is comparatively sensitive to the dictionary size.

Convergence Study

To solve the proposed JSDL, we develop an iterative update algorithm. In this subsection, we experimentally study the speed of convergence of JSDL. In the experiment, the algorithm is terminated when the relative change of the objective function value per iteration is less than 0.001, that is, $|(obj(k) - obj(k+1))| / obj(k) < 0.001$, where $obj(k)$ represents the objective function value at the kth iteration. Figure 5 shows the change of the objective function value of our proposed method as the iteration proceeds.

From the figure, we can observe that the objective function value of our JSDL does not increase during the iteration, and decrease very fast at the first 5 iterations. More importantly, for all datasets, the algorithm finds the minimizers within about 30 iterations, which means that the convergence of our JSDL is relatively fast.

Conclusions

While it has been shown in many works that joint sparse regularization is beneficial to many tasks, it is rarely used in dictionary learning. In this paper, we unify the existing joint spare metric on matrices and give a class of joint sparse metric. Further, we propose a joint sparse regularization dictionary learning method, which is able to remove the redundancy in dictionary and then learn a compact dictionary. To find the solution of the resulting model, we design an efficient iterative algorithm based on CCCP and Lagrangian dual method. We also propose a multi-layer dictionary learning approach with joint sparse regularization. The solution of the multi-layer model can be

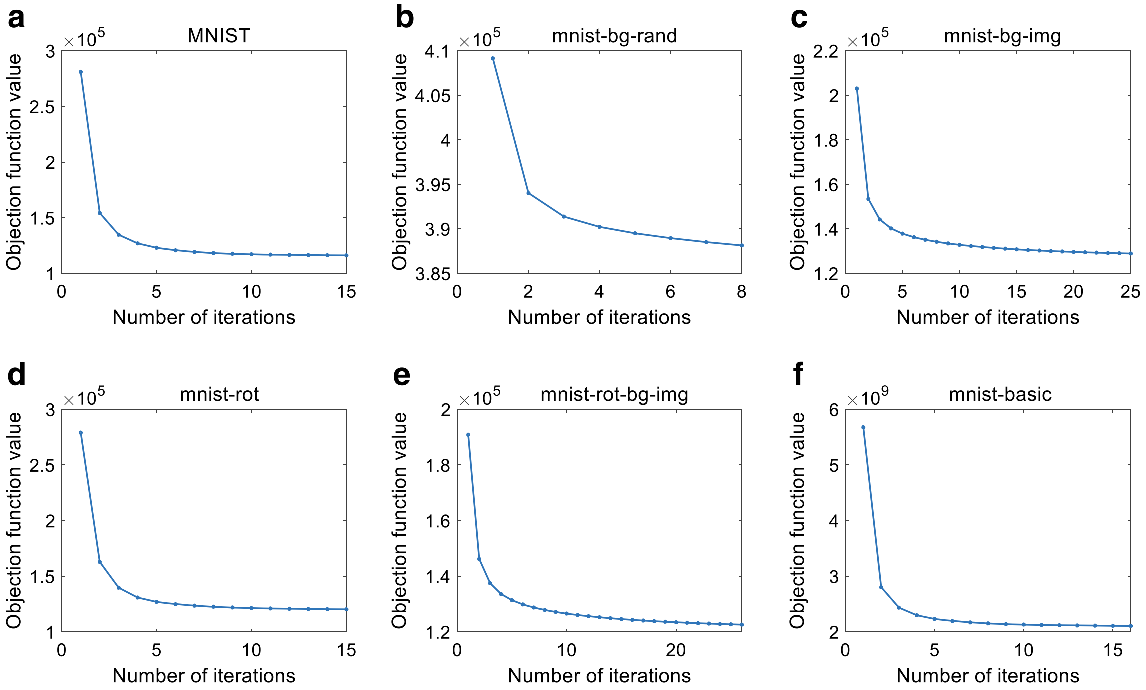

Fig. 5 Convergence behavior of our proposed JSDL method, where the x-axis represents the number of iterations, and the y-axis represents the objective function values. **a** MNIST; **b** mnist-bg-rand(mnist-background-random); **c** mnist-bg-image(mnist-background-image); **d** mnist-rot(mnist-rotated); **e** mnist-rot-bg-img(mnist-rotated-background-image); **f** mnist-basic

achieved by directly employing the algorithm of the shallow model. Extensive experiments on publicly available datasets have validated the effectiveness of the proposed method.

Funding Information This work was supported by National Natural Science Foundation of China (11671379, 61602154, and U1804159), by High-level Talent Fund Project of Henan University of Technology (31401155), by Natural Science Research Project of Henan Provincial Department of Science and Technology (182102210092), and by Fundamental Research Funds for Henan Provincial Colleges and Universities in Henan University of Technology (2016RCJH06).

Compliance with Ethical Standards

Conflict of Interest The authors declare that they have no conflict of interest.

Ethical Approval This article does not contain any studies with human participants performed by any of the authors.

References

1. Aharon M, Elad M. Bruckstein, a.: *k*-svd: an algorithm for designing overcomplete dictionaries for sparse representation. IEEE Trans Signal Process. 2006;54(11):4311–22.
2. Argyriou A, Evgeniou T, Pontil M. Multi-task feature learning. In: Advances in neural information processing systems; 2007. p. 41–8.
3. Bengio Y, Lamblin P, Popovici D, Larochelle H. Greedy layer-wise training of deep networks. In: Advances in neural information processing systems; 2007. p. 153–60.
4. Boyd S, Parikh N, Chu E, Peleato B, Eckstein J. Distributed optimization and statistical learning via the alternating direction method of multipliers. Foundations and Trends®, in Machine Learning. 2011;3(1):1–122.
5. Chen SB, Xin Y, Luo B. Action-based pedestrian identification via hierarchical matching pursuit and order preserving sparse coding. Cogn Comput. 2016;8(5):797–805.
6. Chen Y, Nasrabadi NM, Tran TD. Hyperspectral image classification via kernel sparse representation. IEEE Trans Geosci Remote Sens. 2013;51(1):217–31.
7. Davis G, Mallat S, Avellaneda M. Adaptive greedy approximations. Constr Approx. 1997;13(1):57–98.
8. Fan J, Li R. Variable selection via nonconcave penalized likelihood and its oracle properties. J Am Stat Assoc. 2001;96(456):1348–60.
9. Gui J, Sun Z, Ji S, Tao D, Tan T. Feature selection based on structured sparsity: a comprehensive study. IEEE Trans Neural Netw Learn Syst. 2017;28(7):1490–1507.
10. Hinton GE, Osindero S, Teh YW. A fast learning algorithm for deep belief nets. Neural Comput. 2006;18(7):1527–54.
11. Huang K, Ying Y, Campbell C. Gsml: a unified framework for sparse metric learning. In: 2009 ninth IEEE international conference on data mining. IEEE; 2009. p. 189–98.
12. Jiang W, Nie F, Huang H. Robust dictionary learning with capped l1-norm. In: IJCAI; 2015. p. 3590–96.
13. Jiang Z, Lin Z, Davis LS. Label consistent k-svd: learning a discriminative dictionary for recognition. IEEE Trans Pattern Anal Mach Intell. 2013;35(11):2651–64.
14. Kasiviswanathan SP, Wang H, Banerjee A, Melville P. Online ℓ_1 -dictionary learning with application to novel document detection. In: International conference on neural information processing systems; 2012. p. 2258–66.
15. Larochelle H, Erhan D, Courville A, Bergstra J, Bengio Y. An empirical evaluation of deep architectures on problems with many factors of variation. In: Proceedings of the 24th international conference on Machine learning. ACM; 2007. p. 473–80.
16. Lee H, Battle A, Raina R, Ng AY. Efficient sparse coding algorithms. In: Advances in neural information processing systems; 2007. p. 801–8.
17. Li X, Hu Z, Wang H. Combining non-negative matrix factorization and sparse coding for functional brain overlapping community detection. Cogn Comput. 2018;991–1005.
18. Li Z, Tang J. Weakly supervised deep matrix factorization for social image understanding. IEEE Trans Image Process. 2017;26(1):276–88.
19. Liu H, Sun F. Discovery of topical objects from video: a structured dictionary learning approach. Cogn Comput. 2016;8(3):519–28.
20. Liu H, Wang F, Zhang X, Sun F. Weakly-paired deep dictionary learning for cross-modal retrieval. Pattern Recogn Lett. 2018. https://doi.org/10.1016/j.patrec.2018.06.021.
21. Liu X, Wang L, Zhang J, Yin J, Liu H. Global and local structure preservation for feature selection. IEEE Transactions on Neural Networks and Learning Systems. 2014;25(6):1083–95.
22. Liu X, Zhong G, Dong J. Natural image illuminant estimation via deep non-negative matrix factorisation. IET Image Process. 2017;12(1):121–5.
23. Lou Y, Yin P, He Q, Xin J. Computing sparse representation in a highly coherent dictionary based on difference of l_1 and l_2. J Sci Comput. 2015;64(1):178–96.
24. Mairal J, Bach F, Ponce J, Sapiro G. Online dictionary learning for sparse coding. In: Proceedings of the 26th annual international conference on machine learning. ACM; 2009. p. 689–96.
25. Mairal J, Ponce J, Sapiro G, Zisserman A, Bach FR. Supervised dictionary learning. In: Advances in neural information processing systems; 2009. p. 1033–40.
26. Majumdar A, Ward RK. Improved group sparse classifier. Pattern Recogn Lett. 2010;31(13):1959–64.
27. Majumdar A, Ward RK. Robust classifiers for data reduced via random projections. IEEE Trans Syst Man Cybern B Cybern. 2010;40(5):1359–71.
28. Manjani I, Tariyal S, Vatsa M, Singh R, Majumdar A. Detecting silicone mask-based presentation attack via deep dictionary learning. IEEE Trans Inf Forensics Secur. 2017;12(7):1713–23.
29. Meinshausen N, Yu B. Lasso-type recovery of sparse representations for high-dimensional data. Ann Stat. 2009;37:246–70.
30. Mukherjee S, Basu R, Seelamantula CS. ℓ_1 -k-svd: a robust dictionary learning algorithm with simultaneous update. Signal Process. 2016;123:42–52.
31. Nie F, Huang H, Cai X, Ding CH. Efficient and robust feature selection via joint $\ell_{2,1}$-norms minimization. In: Advances in neural information processing systems; 2010. p. 1813–21.
32. Schmidt MW, Murphy KP, Fung G, Rosales R. Structure learning in random fields for heart motion abnormality detection. In: CVPR; 2008. p. 2.
33. Sharma P, Abrol V, Sao AK. Deep-sparse-representation-based features for speech recognition. IEEE/ACM Transactions on Audio Speech & Language Processing. 2017;25(11):2162–75.
34. Shen Y, Li J, Zhu Z, Cao W, Song Y. Image reconstruction algorithm from compressed sensing measurements by dictionary learning. Neurocomputing. 2015;151:1153–62.
35. Shi Y, Miao J, Wang Z, Zhang P, Niu L. Feature selection with $\ell_{2,1-2}$ regularization. IEEE Trans Neural Netw Learn Syst. 2018;29(10):4967–82.

36. Singhal V, Khurana P, Majumdar A. Class-wise deep dictionary learning. In: 2017 international joint conference on neural networks (IJCNN). IEEE; 2017. p. 1125–32.
37. Tariyal S, Majumdar A, Singh R, Vatsa M. Deep dictionary learning. IEEE Access. 2016;4:10,096–109.
38. Tibshirani R. Regression shrinkage and selection via the lasso. J R Stat Soc Ser B Methodol. 1996;58:267–88.
39. Trigeorgis G, Bousmalis K, Zafeiriou S, Schuller BW. A deep matrix factorization method for learning attribute representations. IEEE Trans Pattern Anal Mach Intell. 2017;39(3):417–29.
40. Tuia D, Flamary R, Barlaud M. Nonconvex regularization in remote sensing. IEEE Trans Geosci Remote Sens. 2016;54(11):6470–80.
41. Wang H, Nie F, Cai W, Huang H. Semi-supervised robust dictionary learning via efficient $\ell_{2,0+}$-norms minimization. In: Proceedings of the IEEE international conference on computer vision. IEEE; 2013. p. 1145–52.
42. Wang S, Liu Q, Xia Y, Dong P, Luo J, Huang Q, Feng DD. Dictionary learning based impulse noise removal via l1–l1 minimization. Signal Process. 2013;93(9):2696–708.
43. Wright J, Yang AY, Ganesh A, Sastry SS, Ma Y. Robust face recognition via sparse representation. IEEE Trans Pattern Anal Mach Intell. 2009;31(2):210–27.
44. Xu J, Yang G, Yin Y, Man H, He H. Sparse-representation-based classification with structure-preserving dimension reduction. Cogn Comput. 2014;6(3):608–21.
45. Xu Z, Chang X, Xu F, Zhang H. l_{1/2} regularization: a thresholding representation theory and a fast solver. IEEE Trans Neural Netw Learn Syst. 2012;23(7):1013–27.
46. Xue HJ, Dai XY, Zhang J, Huang S, Chen J. Deep matrix factorization models for recommender systems. In: International joint conference on artificial intelligence; 2017. p. 3203–9.
47. Yang ZX, Tang L, Zhang K, Wong PK. Multi-view cnn feature aggregation with elm auto-encoder for 3d shape recognition. Cogn Comput. 2018;10(6):908–21.
48. Ying Y, Huang K, Campbell C. Sparse metric learning via smooth optimization. In: Advances in neural information processing systems; 2009. p. 2214–22.
49. Zhang C et al. Nearly unbiased variable selection under minimax concave penalty. Ann Statist. 2010;38(2):894–942.
50. Zhang L, Zhou WD, Chang PC, Liu J, Yan Z, Wang T, Li FZ. Kernel sparse representation-based classifier. IEEE Trans Signal Process. 2012;60(4):1684–95.
51. Zhang M, Ding CH, Zhang Y, Nie F. Feature selection at the discrete limit. In: AAAI; 2014. p. 1355–61.
52. Zhang Q, Li B. Discriminative k-svd for dictionary learning in face recognition. In: 2010 IEEE conference on computer vision and pattern recognition (CVPR). IEEE; 2010. p. 2691–98.
53. Zhang Z, Xiahou J, Bai ZJ, Hancock ER, Zhou D, Chen SB, Chen L. Discriminative lasso. Cogn Comput. 2016;8(5):847–55.
54. Zheng A, Xu M, Luo B, Zhou Z, Li C. Class: Collaborative low-rank and sparse separation for moving object detection. Cogn Comput. 2017;9(2):180–93.
55. Zhou P, Fang C, Lin Z, Zhang C, Chang EY. Dictionary learning with structured noise. Neurocomputing. 2018;273:414–23.

Affiliations

Jianyu Miao[1] · Heling Cao[1] · Xiao-Bo Jin[2] · Rongrong Ma[3,4] · Xuan Fei[1] · Lingfeng Niu[4,5]

Jianyu Miao
jymiao@haut.edu.cn

Heling Cao
caohl@haut.edu.cn

Xiao-Bo Jin
xbjin9801@gmail.com

Rongrong Ma
marongrong16@mails.ucas.ac.cn

Xuan Fei
feixuan@haut.edu.cn

[1] College of Information Science and Engineering, Henan University of Technology, Zhengzhou, 450001, China

[2] Department of Electrical and Electronic Engineering, Xi'an Jiaotong-Liverpool University, Suzhou, 215123, China

[3] School of Mathematical Sciences, University of Chinese Academy of Sciences, Beijing, 100049, China

[4] Key Laboratory of Big Data Mining and Knowledge Management, Chinese Academy of Sciences, Beijing 100190, China

[5] School of Economics and Management, University of Chinese Academy of Sciences, Beijing, 100190, China