Neurocomputing 315 (2018) 394–404

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Graph-based multiple rank regression for image classification

Haoliang Yuan[a,*], Junyu Li[a], Loi Lei Lai[a,*], Yuan Yan Tang[b,c]

[a] School of Automation, Guangdong University of Technology, Guangzhou 510006, China
[b] Department of Computer and Information Science, Faculty of Science and Technology, University of Macau, Macau 999078, China
[c] Beijing Advanced Innovation Center for Big Data and Brain Computing, Beihang University, Beijing 100191, China

ARTICLE INFO

Article history:
Received 24 January 2018
Revised 23 June 2018
Accepted 16 July 2018
Available online 7 August 2018

Communicated by Deng Cai

Keywords:
Multiple rank regression
Graph regularization
Image classification

ABSTRACT

Image classification is one important task in image processing and pattern recognition. Traditional image classification methods commonly transform the image into a vector. However, in essence, image is a matrix data and using vector instead of image loses the correlations of the matrix data. To address this problem, we propose a graph-based multiple rank regression model (GMRR), which employs multiple-rank left and right projecting vectors to regress each matrix data to its label for each category. To exploit the discriminating structure of the data space, a class compactness graph is constructed to constrain these left and right projecting vectors. The extensive experimental results on image classification have demonstrated the effectiveness of our proposed method.

1. Introduction

Matrix data is commonly confronted in many practical applications, such as image processing, pattern recognition, and data mining. For example, in face recognition task, the face data are generally presented as the matrix form. How to classify matrix data is an important research topic. Traditional classification methods are usually based on vector data. Widely used methods are *K*-nearest neighborhoods classifier [1], support vector machine [2], logistic regression [3], least squares regression [4] and sparse (collaborative) representation-based classifier [5,6]. Therefore, to make the matrix data fit for these vector-based methods, one common way is to reformulate a vector by connecting each row (or column) of a matrix. However, the reshaped vectors break the natural structure of the image and ignore the location information of original matrix element.

To address this problem, a natural idea is using the original matrix data for classification directly. Hence, lots of matrix-based methods are proposed for data analysis. Ye et al. [7] develop a new technique called 2-dimensional linear discriminant analysis (2DLDA) for image representation. Hung and Wang [8] propose a matrix variate logistic regression model, which includes the preservation of the inherent matrix structure of covariates and the parsimony of parameters needed. Luo et al. [9] propose a new classification method called support matrix machine, which is defined as a hinge loss plus a so-called spectral elastic net penalty. Chu et al. [10] propose a atom decomposition based subgradient descent method to address the matrix classification problem. Zheng et al. [11] propose a novel matrix classifier to simultaneously leverage the structural information within matrices and select useful features. Yang et al. [12–14] present a two-dimensional image-matrix-based error model named nuclear norm based matrix regression for face representation and classification. Zheng et al. [15] propose a novel classifier to address the multi-class classification of single trial EEG signals in matrix form. Mathematically, a matrix is a two-order tensor. Hence, tensor learning [16–20] can also be used for image analysis.

In this paper, we focus on studying the least squares regression model, which is a typical and fundamental technique in statistics theory [21]. Gabriel [22] proposes a generalized bilinear regression (GBR) model. Recently, Hou et al. [23] propose a novel multiple rank regression model (MRR) for matrix data classification. Comparing with GBR using one left projecting vector together with one right projecting vector, each matrix data in MRR is regressed to its label vector by using multiple rank left and right regression vectors. The mechanism of MRR is to use these multiple rank left and right regression vectors to transform the matrix samples into a feature vector. Hence it is naturally assumed that the matrix samples sharing the same labels should be kept close together in the feature space. However, MRR fails to consider this assumption.

Manifold learning [24–26] aims to ensure that similar data have nearly the same labels by using the graph embedding. It has been shown that learning performance can be significantly enhanced if the graph structure is exploited. Cai et al. [27] propose a graph regularized nonnegative matrix factorization model, which constructs

* Corresponding authors.
E-mail addresses: hunteryuan@126.com (H. Yuan), l.l.lai@ieee.org (L.L. Lai).

https://doi.org/10.1016/j.neucom.2018.07.032
0925-2312/

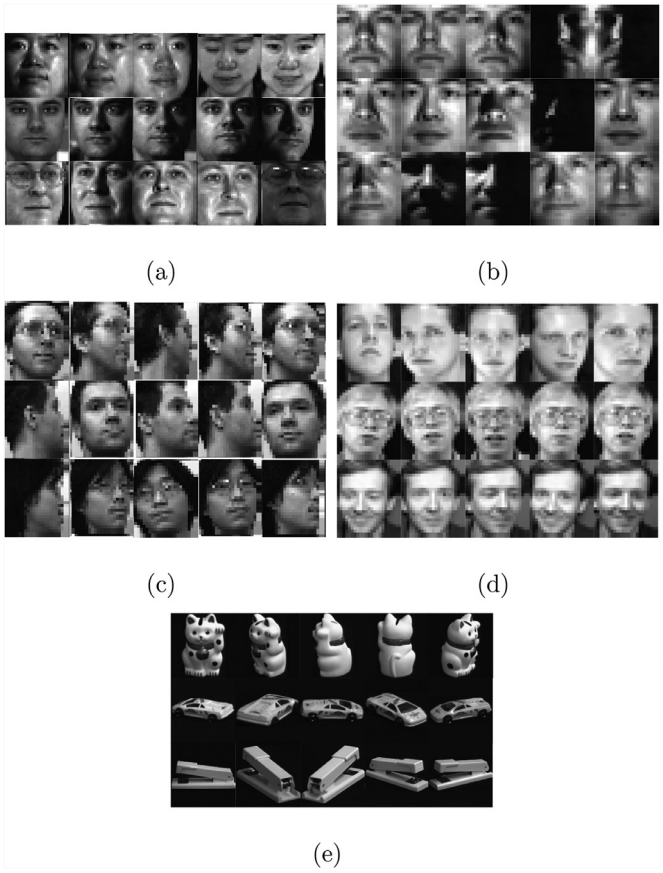

Fig. 1. Examples of images from the five different data sets. Each row denotes one subject. (a) CMU PIE. (b) Extended YaleB. (c) Umist. (d) ORL. (e) COIL-100.

Table 1
Characters of different data sets.

Data set	All samples	Size ($m \times n$)	Class	No. of labeled samples
PIE	11554	32×32	68	$136, 204, \ldots, 748$
YaleB	2414	24×21	38	$152, 190, \ldots, 494$
Umist	575	28×23	20	$40, 60, \ldots, 220$
ORL	400	32×32	40	$80, 120, \ldots, 320$
COIL-100	7200	32×32	100	$200, 300, \ldots, 1200$

a nearest neighbor graph to model the manifold structure. Li et al. [28] present a graph-based local concept coordinate factorization method, which respects the intrinsic structure of the data through manifold kernel learning in the warped Reproducing Kernel Hilbert Space. Zheng et al. [29] propose a graph regularized sparse coding algorithm to learn the sparse representations that explicitly take into account the local manifold structure of the data. Lu et al. [30] propose a graph-regularized $L_{1/2}$-NMF for hyperspectral unmixing, which incorporates a graph regularization to detect the geometrical structure of the hyperspectral data. Li et al. [31] develop a constrained low-rank representation model, which imposes a Laplacian regularizer to pair the low-dimensional representation of original data.

Motivated by the idea of manifold learning, in this paper, we propose a graph-based multiple rank regression model (GMRR), which explicitly exploits the discriminating structure of the data space. We capture the discriminating information of the data space by incorporating a class compactness graph into the objective function of MRR. To effectively optimize our proposed algorithm, we also develop an optimization scheme to solve the objective function of GMRR. The convergence proof of our optimization scheme is provided. Extensive experiments demonstrate the superiority of GMRR. In summary, the contributions of this paper are summarized as follows:

1) Our model exploits the discriminating structure of the data distribution and incorporates it as an additional graph regularization term into the objective function of MRR.

2) Our model constructs a class compactness graph to model the discriminating structure. Through preserving this graph structure in the transformation space, our model can obtain more discriminative power than MRR.

3) Theoretical and experimental analyses of convergence behaviors and computation complexities of our model are given.

The rest of this paper is organized as follows. Section 2 briefly describes the related work. We formulate the GMRR and provide the corresponding optimization procedure in Section 3. The performance analysis including convergence behavior and computational complexity are presented in Section 4. In Section 5, experimental results are conducted and discussed. Finally, the conclusion of this paper is drawn in Section 6.

2. Related work

We briefly describe the MRR model. Denote $\{\mathbf{X}_i \in \mathbb{R}^{m\times n} | i = 1, 2, \ldots, l\}$ as the set of labeled matrix samples and the associated class label vectors are $\{\mathbf{y}_i \in \mathbb{R}^{c\times 1} | i = 1, 2, \ldots, l\}$. Here, $\mathbf{y}_i = [y_{i1}, y_{i2}, \ldots, y_{ic}]^T$, where $y_{ij} = 1$ if and only if $\mathbf{X}_i$ belongs to the jth category and $y_{ij} = 0$ otherwise. c is the number of classes. m and n are the first and second dimensionality of each matrix data. l is the number of labeled matrix samples. The couples of left projecting vectors and right projecting vectors are denoted as $\{\mathbf{u}_j^{(r)}\}_{j=1}^k \in \mathbb{R}^{m\times 1}$ and $\{\mathbf{v}_j^{(r)}\}_{j=1}^k \in \mathbb{R}^{n\times 1}$, respectively. In the following, the objective function of MRR model is formulated as

$$\sum_{r=1}^{c}\left[\sum_{i=1}^{l}\left(\sum_{j=1}^{k}(\mathbf{u}_j^{(r)})^T\mathbf{X}_i\mathbf{v}_j^{(r)} + b^{(r)} - y_{ir}\right)^2 + \alpha\sum_{j=1}^{k}\mathrm{Tr}\left(\mathbf{u}_j^{(r)}(\mathbf{v}_j^{(r)})^T\mathbf{v}_j^{(r)}(\mathbf{u}_j^{(r)})^T\right)\right] \quad (1)$$

where $\alpha > 0$ is the regularization parameter and $b^{(r)}$ is the unknown bias for the rth category.

3. Graph-based multiple rank regression

In this section, we firstly introduce the formulation of our proposed GMRR model. Next, an alternative optimization algorithm is presented to solve GMRR. Finally, we provide a way to classify new matrix samples.

3.1. GMRR

Since MRR can be divided into c separate regression problems, the loss function in model (1) for the rth classifier is represented as

$$\sum_{i=1}^{l}\left(\sum_{j=1}^{k}(\mathbf{u}_j^{(r)})^T\mathbf{X}_i\mathbf{v}_j^{(r)} + b^{(r)} - y_{ir}\right)^2 \quad (2)$$

Denote $\mathbf{U}^{(r)} = [\mathbf{u}_1^{(r)}, \mathbf{u}_2^{(r)}, \ldots, \mathbf{u}_k^{(r)}] \in \mathbb{R}^{m\times k}$ and $\mathbf{V}^{(r)} = [\mathbf{v}_1^{(r)}, \mathbf{v}_2^{(r)}, \ldots, \mathbf{v}_k^{(r)}] \in \mathbb{R}^{n\times k}$, the loss function in model (2) can be reformulated as

$$\sum_{i=1}^{l}\left(\mathrm{Tr}((\mathbf{U}^{(r)})^T\mathbf{X}_i\mathbf{V}^{(r)}) + b^{(r)} - y_{ir}\right)^2 \quad (3)$$

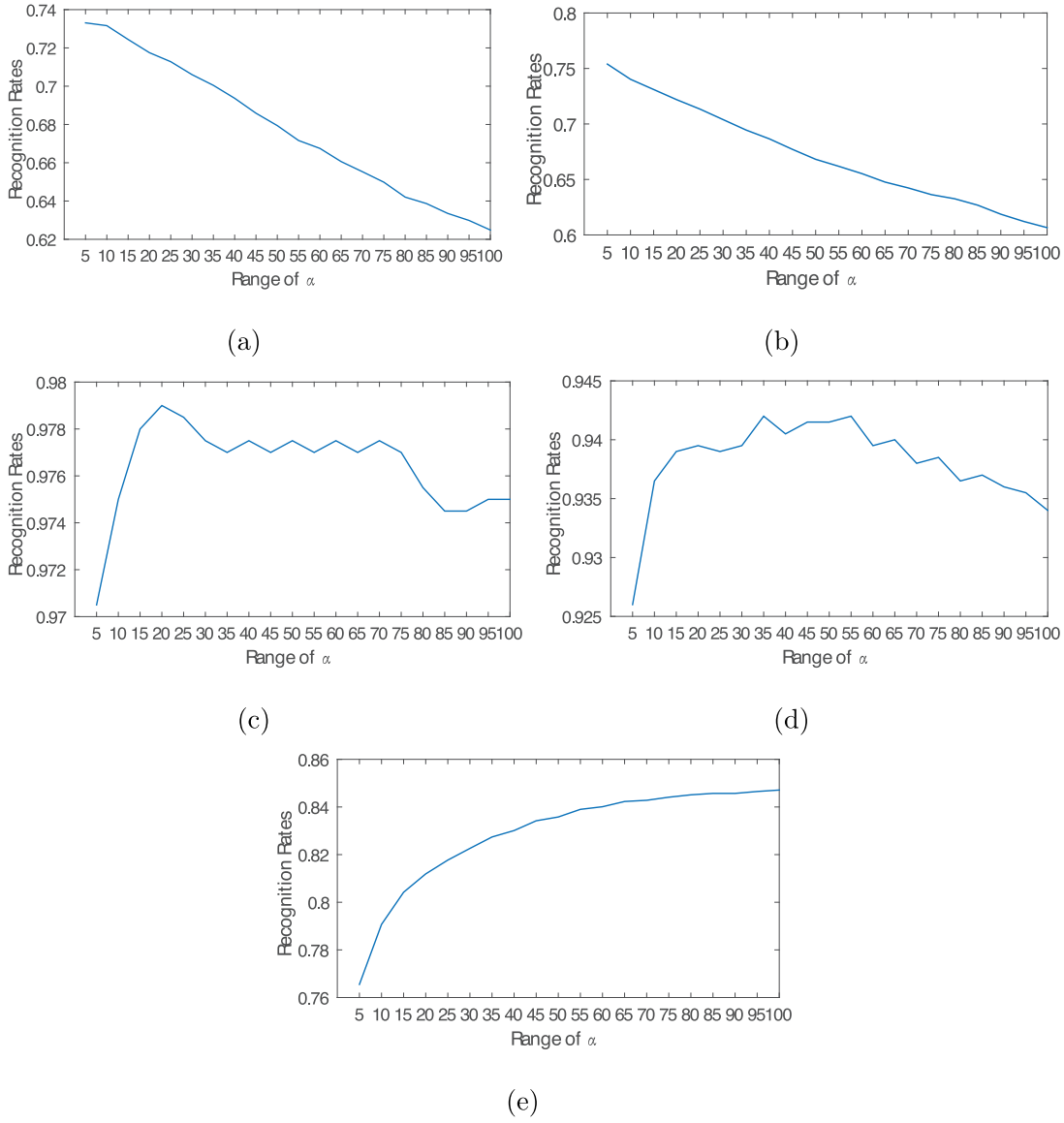

Fig. 2. The impact on different parameters α in MRR. (a) PIE. (b) YaleB. (c) Umist. (d) ORL. (e) COIL-100.

For the rth classifier of MRR, we have

$$\begin{aligned}\mathrm{Tr}((\mathbf{U}^{(r)})^T\mathbf{X}_i\mathbf{V}^{(r)}) &= \mathrm{Tr}(\mathbf{V}^{(r)}(\mathbf{U}^{(r)})^T\mathbf{X}_i) = \mathrm{Tr}((\mathbf{U}^{(r)}(\mathbf{V}^{(r)})^T)^T\mathbf{X}_i)\\ &= (\mathrm{Vec}(\mathbf{U}^{(r)}(\mathbf{V}^{(r)})^T))^T\mathbf{x}_i \end{aligned} \tag{4}$$

where $\mathrm{Vec}(\cdot)$ is an operator which can convert a matrix to a vector by collecting the columns. We denote $\{\mathbf{x}_i = \mathrm{Vec}(\mathbf{X}_i)|i = 1, 2, \ldots, l\}$ as the vector counterparts of the matrix samples. Here, $\mathrm{Vec}(\mathbf{U}^{(r)}(\mathbf{V}^{(r)})^T))^T$ can be viewed as a transformation vector. Here, we denote $\mathbf{p}_r = \mathrm{Vec}(\mathbf{U}^{(r)}(\mathbf{V}^{(r)})^T)$ and $\mathbf{P} = [\mathbf{p}_1, \mathbf{p}_2, \ldots, \mathbf{p}_c] \in \mathbb{R}^{mn\times c}$.

In MRR, it naturally hopes that the matrix samples sharing the same labels should be kept close together in the transformed space. However, this transformation matrix $\mathbf{P}$ fails to achieve it. Inspired by the idea from manifold learning, a class compactness graph is constructed to address this problem. In the class compactness graph, two nodes corresponding to two different matrix samples from the same class are linked by an undirected edge. Moreover, the matrix samples from the same class also exist different similarities. Hence the weight of the class compactness graph is reasonably represented as follows

$$W_{i,j} = \begin{cases} \mathbf{e}^{-\frac{\|\mathbf{x}_i-\mathbf{x}_j\|_F^2}{\sigma}}, & \text{if } \mathbf{X}_i \text{ and } \mathbf{X}_j \text{ are from the same class} \\ 0, & \text{otherwise} \end{cases} \tag{5}$$

where σ is the heat kernel parameter. To preserve this the graph structure in the transformation space, a reasonable criterion for choosing a good mapping is to minimize the following objective function

$$\min_{\mathbf{P}} \sum_{i,j=1}^{l} \|\mathbf{P}^T\mathbf{x}_i - \mathbf{P}^T\mathbf{x}_j\|^2 W_{i,j} \Leftrightarrow \min_{\mathbf{P}} \mathrm{Tr}(\mathbf{P}^T\mathbf{X}\mathbf{L}\mathbf{X}^T\mathbf{P}) \tag{6}$$

where $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_l] \in \mathbb{R}^{mn\times l}$ and $\mathbf{L}$ is the Laplacian matrix, which is defined as $\mathbf{L} = \mathbf{D} - \mathbf{W}$, where $\mathbf{D}$ is a diagonal matrix and its diagonal entries are defined as $D_{ii} = \sum_{j=1}^{l} W_{ij}$. By incorporating the graph regularizer (6) into the MRR model, we can get the fol-

lowing objective function of GMRR as follows

$$\sum_{r=1}^{c}\left[\sum_{i=1}^{l}(\mathrm{Tr}((\mathbf{U}^{(r)})^T\mathbf{X}_i\mathbf{V}^{(r)})+b^{(r)}-y_{ir})^2 + \alpha\sum_{j=1}^{k}\mathrm{Tr}(\mathbf{u}_j^{(r)}(\mathbf{v}_j^{(r)})^T\mathbf{v}_j^{(r)}(\mathbf{u}_j^{(r)})^T)\right]+\beta\mathrm{Tr}(\mathbf{P}^T\mathbf{X}\mathbf{L}\mathbf{X}^T\mathbf{P}) \quad (7)$$

where $\beta>0$ is the regularization parameter. $\mathbf{P}=[\mathbf{p}_1,\mathbf{p}_2,\ldots,\mathbf{p}_c]\in\mathbb{R}^{mn\times c}$ and $\mathbf{p}_r=\mathrm{Vec}(\mathbf{U}^{(r)}(\mathbf{V}^{(r)})^T)$.

3.2. Optimization algorithm

Through observing the optimization problem in (7), we find: 1) $\mathbf{U}^{(r)}$ and $\mathbf{V}^{(r)}$ are coupled in the loss function; 2) $\mathbf{U}^{(r)}$ and $\mathbf{V}^{(r)}$ are coupled together in forming $\mathbf{P}$. Thus, it is difficult to solve them simultaneously. Next, we will design an alternative algorithm to optimize them.

3.2.1. Computing $\boldsymbol{U}^{(r)}$

We first fix $\{\mathbf{V}^{(r)}\}_{r=1}^{c}$ and optimize $\{\mathbf{U}^{(r)}\}_{r=1}^{c}$. Through mathematical derivations, the following equation holds

$$\mathrm{Tr}(\mathbf{P}^T\mathbf{X}\mathbf{L}\mathbf{X}^T\mathbf{P})=\sum_{r=1}^{c}\mathbf{p}_r^T\mathbf{X}\mathbf{L}\mathbf{X}^T\mathbf{p}_r \quad (8)$$

Thus, we can decompose the objective function in (7) into the following c independent sub-problems

$$L(\mathbf{U}^{(r)},\mathbf{V}^{(r)},b^{(r)})=\sum_{i=1}^{l}(\sum_{j=1}^{k}(\mathbf{u}_j^{(r)})^T\mathbf{X}_i\mathbf{v}_j^{(r)}+b^{(r)}-y_{ir})^2 + \alpha\sum_{j=1}^{k}\mathrm{Tr}(\mathbf{u}_j^{(r)}(\mathbf{v}_j^{(r)})^T\mathbf{v}_j^{(r)}(\mathbf{u}_j^{(r)})^T)+\beta\mathbf{p}_r^T\mathbf{X}\mathbf{L}\mathbf{X}^T\mathbf{p}_r \quad (9)$$

To effectively solve $\mathbf{U}^{(r)}$, we need to define

$$\mathbf{g}_i^{(r)}=\begin{bmatrix}\mathbf{X}_i\mathbf{v}_1^{(r)}\\ \mathbf{X}_i\mathbf{v}_2^{(r)}\\ \vdots\\ \mathbf{X}_i\mathbf{v}_k^{(r)}\end{bmatrix}\in\mathbb{R}^{mk\times 1},\ \tilde{\mathbf{u}}^{(r)}=\begin{bmatrix}\mathbf{u}_1^{(r)}\\ \mathbf{u}_2^{(r)}\\ \vdots\\ \mathbf{u}_k^{(r)}\end{bmatrix}\in\mathbb{R}^{mk\times 1},$$

$$\mathbf{A}^{(r)}=\begin{bmatrix}(\mathbf{v}_1^{(r)})^T\mathbf{v}_1^{(r)}\mathbf{I}_{m\times m} & & \\ & \cdots & \\ & & (\mathbf{v}_k^{(r)})^T\mathbf{v}_k^{(r)}\mathbf{I}_{m\times m}\end{bmatrix}\in\mathbb{R}^{mk\times mk}, \quad (10)$$

where $\mathbf{I}_{m\times m}$ is the identity matrix with the size of $m\times m$. Then, the loss function in (9) is equivalent to

$$\sum_{i=1}^{l}\left((\tilde{\mathbf{u}}_i^{(r)})^T\mathbf{g}_i^{(r)}+b^{(r)}-y_{ir}\right)^2 \quad (11)$$

and

$$\sum_{j=1}^{k}\mathrm{Tr}(\mathbf{u}_j^{(r)}(\mathbf{v}_j^{(r)})^T\mathbf{v}_j^{(r)}(\mathbf{u}_j^{(r)})^T)=(\tilde{\mathbf{u}}^{(r)})^T\mathbf{A}^{(r)}\tilde{\mathbf{u}}^{(r)} \quad (12)$$

It is easy to find that $\tilde{\mathbf{u}}^{(r)}=\mathrm{Vec}(\mathbf{U}^{(r)})$. Hence $\mathbf{p}_r$ can be reformulated with respect to $\tilde{\mathbf{u}}^{(r)}$ as follows

$$\begin{aligned}\mathbf{p}_r&=\mathrm{Vec}(\mathbf{U}^{(r)}(\mathbf{V}^{(r)})^T)=\mathrm{Vec}(\mathbf{I}_{m\times m}\mathbf{U}^{(r)}(\mathbf{V}^{(r)})^T)\\&=(\mathbf{V}^{(r)}\otimes\mathbf{I}_{m\times m})\mathrm{Vec}(\mathbf{U}^{(r)})=(\mathbf{V}^{(r)}\otimes\mathbf{I}_{m\times m})\tilde{\mathbf{u}}^{(r)}\end{aligned} \quad (13)$$

where $\otimes$ denotes the Kronecker product. The regularization term in (9) can be reformulated as

$$\begin{aligned}\mathbf{p}_r^T\mathbf{X}\mathbf{L}\mathbf{X}^T\mathbf{p}_r&=(\tilde{\mathbf{u}}^{(r)})^T(\mathbf{V}^{(r)}\otimes\mathbf{I}_{m\times m})^T\mathbf{X}\mathbf{L}\mathbf{X}^T(\mathbf{V}^{(r)}\otimes\mathbf{I}_{m\times m})\tilde{\mathbf{u}}^{(r)}\\&=(\tilde{\mathbf{u}}^{(r)})^T\mathbf{B}^{(r)}\tilde{\mathbf{u}}^{(r)}\end{aligned} \quad (14)$$

where $\mathbf{B}^{(r)}=(\mathbf{V}^{(r)}\otimes\mathbf{I}_{m\times m})^T\mathbf{X}\mathbf{L}\mathbf{X}^T(\mathbf{V}^{(r)}\otimes\mathbf{I}_{m\times m})$.

Next, we denote

$$\mathbf{G}^{(r)}=[\mathbf{g}_1^{(r)},\mathbf{g}_2^{(r)},\ldots,\mathbf{g}_l^{(r)}]\in\mathbb{R}^{mk\times l} \quad (15)$$

$$\mathbf{y}^{(r)}=[y_{1r},y_{2r},\ldots,y_{lr}]\in\mathbb{R}^{1\times l} \quad (16)$$

Then, model (9) becomes

$$L(\tilde{\mathbf{u}}^{(r)})=\|(\tilde{\mathbf{u}}^{(r)})^T\mathbf{G}^{(r)}+b^{(r)}\mathbf{e}-\mathbf{y}^{(r)}\|_2^2+(\tilde{\mathbf{u}}^{(r)})^T\mathbf{C}^{(r)}\tilde{\mathbf{u}}^{(r)} \quad (17)$$

where $\mathbf{e}\in\mathbb{R}^{1\times l}$ is a row vector whose elements are all 1 and $\mathbf{C}^{(r)}=\alpha\mathbf{A}^{(r)}+\beta\mathbf{B}^{(r)}$. Model (17) is a regularized least squares problem. We firstly compute

$$b^{(r)}=\frac{1}{l}\left(\mathbf{y}^{(r)}-(\tilde{\mathbf{u}}^{(r)})^T\mathbf{G}^{(r)}\right)\mathbf{e}^T \quad (18)$$

Combining (17) and (18), we take the derivative of $L(\tilde{\mathbf{u}}^{(r)})$ with respect to $\tilde{\mathbf{u}}^{(r)}$ and set it to zero. The optimal solution for (17) is

$$\tilde{\mathbf{u}}^{(r)}=\left[\mathbf{G}^{(r)}\mathbf{H}(\mathbf{G}^{(r)})^T+\mathbf{C}^{(r)}\right]^{-1}\mathbf{G}^{(r)}\mathbf{H}(\mathbf{y}^{(r)})^T \quad (19)$$

where $\mathbf{H}=\mathbf{I}_{l\times l}-\frac{1}{l}\mathbf{e}^T\mathbf{e}\in\mathbb{R}^{l\times l}$.

3.2.2. Computing $\boldsymbol{V}^{(r)}$ and $b^{(r)}$

We then fix $\mathbf{U}^{(r)}$ and optimize $\mathbf{V}^{(r)}$ and $b^{(r)}$. Similar to optimization of $\mathbf{U}^{(r)}$ in (9), we also want to solve $\mathbf{V}^{(r)}$ and $b^{(r)}$ as a regularized least squares problem just like (17). However, $\mathbf{p}_r$ in (13) is formulated by $\mathrm{Vec}((\mathbf{V}^{(r)})^T)$ not $\mathrm{Vec}(\mathbf{V}^{(r)})$ as

$$\begin{aligned}\mathbf{p}_r&=\mathrm{Vec}(\mathbf{U}^{(r)}(\mathbf{V}^{(r)})^T=\mathrm{Vec}(\mathbf{U}^{(r)}(\mathbf{V}^{(r)})\mathbf{I}_{n\times n})\\&=(\mathbf{I}_{n\times n}\otimes\mathbf{U}^{(r)})\mathrm{Vec}((\mathbf{V}^{(r)})^T)\end{aligned} \quad (20)$$

Since $\mathbf{p}_r$ is not formulated by $\mathrm{Vec}(\mathbf{V}^{(r)})$, we can't perform similar derivation when fixing $\mathbf{U}^{(r)}$. To address this problem, we define $\mathbf{q}_r=\mathrm{Vec}(\mathbf{V}^{(r)}(\mathbf{U}^{(r)})^T)$. Then we have

$$\mathrm{Tr}(\mathbf{p}_r^T\mathbf{X}\mathbf{L}\mathbf{X}^T\mathbf{p}_r)=\mathrm{Tr}(\mathbf{q}_r^T\mathbf{Z}\mathbf{L}\mathbf{Z}^T\mathbf{q}_r) \quad (21)$$

where $\mathbf{Z}=[\mathbf{z}_1,\mathbf{z}_2,\ldots,\mathbf{z}_l]\in\mathbf{R}^{mn\times l}$ and $\mathbf{z}_i=\mathrm{Vec}(\mathbf{X}_i^T)$. Since for any two matrices $\mathbf{A}$ and $\mathbf{B}$, we have $(\mathrm{Vec}(\mathbf{A}))^T\mathrm{Vec}(\mathbf{B})=(\mathrm{Vec}(\mathbf{A}^T))^T\mathrm{Vec}(\mathbf{B}^T)$. Hence, $(\mathrm{Vec}(\mathbf{U}^{(r)}(\mathbf{V}^{(r)})^T))^T\mathrm{Vec}(\mathbf{X}_i)=(\mathrm{Vec}(\mathbf{V}^{(r)}(\mathbf{U}^{(r)})^T))^T\mathrm{Vec}(\mathbf{X}_i^T)$, i.e., $\mathbf{p}_r^T\mathbf{x}_i=\mathbf{q}_r^T\mathbf{z}_i$. Moreover, since $\mathbf{e}^{-\frac{\|\mathbf{x}_i-\mathbf{x}_j\|_F^2}{\sigma}}=\mathbf{e}^{-\frac{\|\mathbf{z}_i-\mathbf{z}_j\|_F^2}{\sigma}}$, Eq. (21) holds.

Considering the results in Eq. (21), we can reformulate the objective function of GMRR in (9) as

$$L(\mathbf{U}^{(r)},\mathbf{V}^{(r)},b^{(r)})=\sum_{i=1}^{l}(\sum_{ji=1}^{k}(\mathbf{v}_j^{(r)})^T\mathbf{X}_i^T\mathbf{u}_j^{(r)}+b^{(r)}-y_{ir})^2 + \alpha\sum_{j=1}^{k}\mathrm{Tr}(\mathbf{v}_j^{(r)}(\mathbf{u}_j^{(r)})^T\mathbf{u}_j^{(r)}(\mathbf{v}_j^{(r)})^T)+\beta\mathbf{q}_r^T\mathbf{Z}\mathbf{L}\mathbf{Z}^T\mathbf{q}_r \quad (22)$$

with $\mathbf{q}_r=\mathrm{Vec}(\mathbf{V}^{(r)}(\mathbf{U}^{(r)})^T)$. We denote

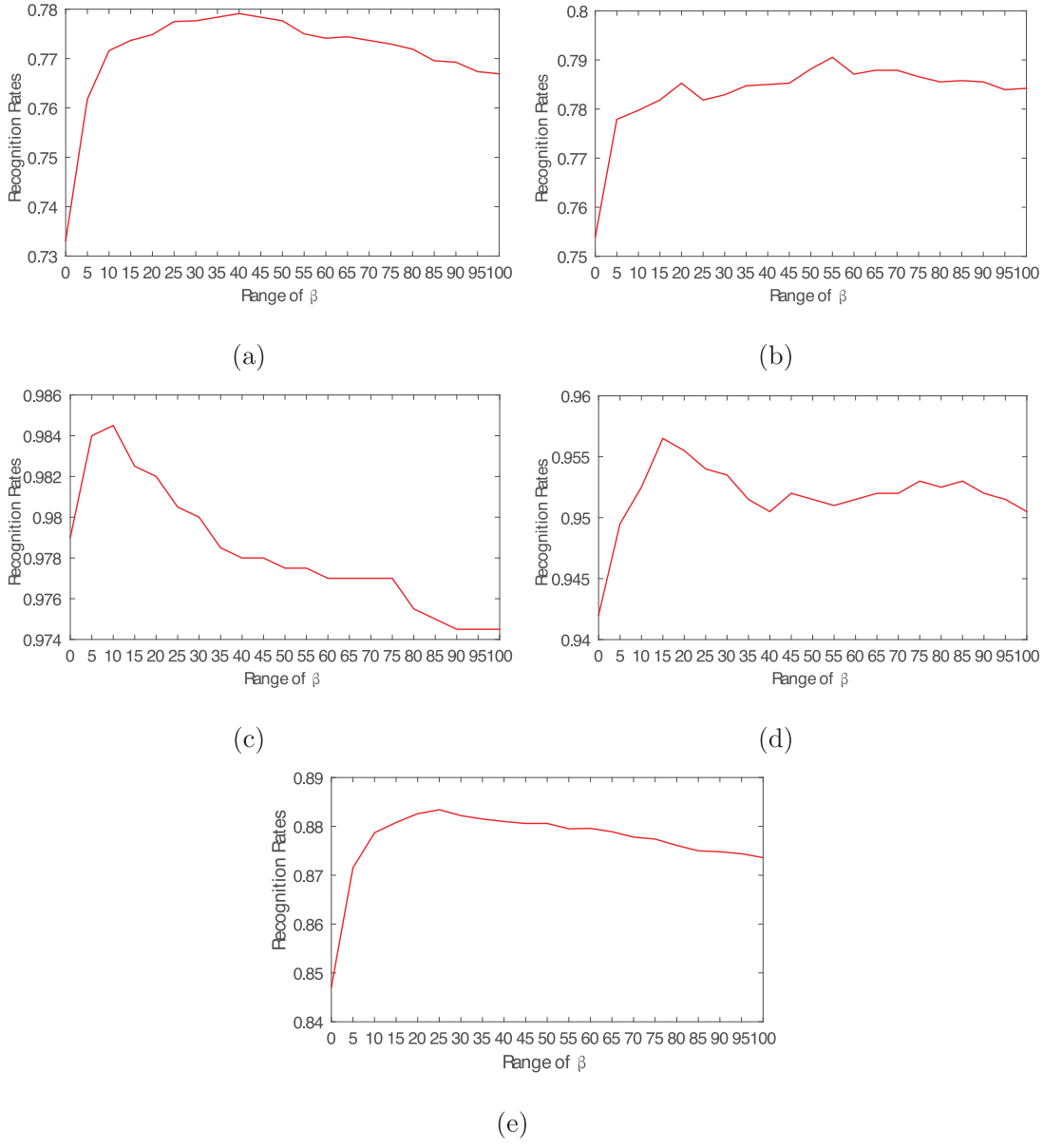

Fig. 3. The impact on different parameters β in GMRR when the optimal α is fixed. (a) PIE. (b) YaleB. (c) Umist. (d) ORL. (e) COIL-100.

$$\mathbf{s}_i^{(r)} = \begin{bmatrix} \mathbf{X}_i^T\mathbf{u}_1^{(r)} \\ \mathbf{X}_i^T\mathbf{u}_2^{(r)} \\ \vdots \\ \mathbf{X}_i^T\mathbf{u}_k^{(r)} \end{bmatrix} \in \mathbb{R}^{nk\times 1}, \tilde{\mathbf{v}}^{(r)} = \begin{bmatrix} \mathbf{v}_1^{(r)} \\ \mathbf{v}_2^{(r)} \\ \vdots \\ \mathbf{v}_k^{(r)} \end{bmatrix} \in \mathbb{R}^{nk\times 1}$$

$$\mathbf{D}^{(r)} = \begin{bmatrix} (\mathbf{u}_1^{(r)})^T\mathbf{u}_1^{(r)}\mathbf{I}_{n\times n} & & \\ & \cdots & \\ & & (\mathbf{u}_k^{(r)})^T\mathbf{u}_k^{(r)}\mathbf{I}_{n\times n} \end{bmatrix} \in \mathbb{R}^{nk\times nk} \tag{23}$$

where $\mathbf{I}_{n\times n}$ is the identity matrix with the size of $n\times n$. Then, the loss function in (22) is equivalent to

$$\sum_{i=1}^{l}\left((\tilde{\mathbf{v}}_i^{(r)})^T\mathbf{s}_i^{(r)} + b^{(r)} - y_{ir}\right)^2 \tag{24}$$

and

$$\sum_{j=1}^{k}\mathrm{Tr}(\mathbf{v}_j^{(r)}(\mathbf{u}_j^{(r)})^T\mathbf{u}_j^{(r)}(\mathbf{v}_j^{(r)})^T)] = (\tilde{\mathbf{v}}^{(r)})^T\mathbf{D}^{(r)}\tilde{\mathbf{v}}^{(r)} \tag{25}$$

Note that $\tilde{\mathbf{v}}^{(r)} = \mathrm{Vec}(\mathbf{V}^{(r)})$, then $\mathbf{q}_r$ can be reformulated with respect to $\tilde{\mathbf{v}}_i^{(r)}$ as follows

$$\begin{aligned}\mathbf{q}_r &= \mathrm{Vec}(\mathbf{V}^{(r)}(\mathbf{U}^{(r)})^T) = \mathrm{Vec}(\mathbf{I}_{n\times n}\mathbf{V}^{(r)}(\mathbf{U}^{(r)})^T) \\ &= (\mathbf{U}^{(r)}\otimes\mathbf{I}_{n\times n})\mathrm{Vec}(\mathbf{V}^{(r)}) = (\mathbf{U}^{(r)}\otimes\mathbf{I}_{n\times n})\tilde{\mathbf{v}}^{(r)}\end{aligned} \tag{26}$$

Then, the graph regularizer in (22) can be reformulated as

$$\begin{aligned}\mathbf{q}_r^T\mathbf{Z}\mathbf{L}\mathbf{Z}^T\mathbf{q}_r &= (\tilde{\mathbf{v}}^{(r)})^T(\mathbf{U}^{(r)}\otimes\mathbf{I}_{n\times n})^T\mathbf{Z}\mathbf{L}\mathbf{Z}^T(\mathbf{U}^{(r)}\otimes\mathbf{I}_{n\times n})\tilde{\mathbf{v}}^{(r)} \\ &= (\tilde{\mathbf{v}}^{(r)})^T\mathbf{E}^{(r)}\tilde{\mathbf{v}}^{(r)}\end{aligned} \tag{27}$$

where $\mathbf{E}^{(r)} = (\mathbf{U}^{(r)}\otimes\mathbf{I}_{n\times n})^T\mathbf{Z}\mathbf{L}\mathbf{Z}^T(\mathbf{U}^{(r)}\otimes\mathbf{I}_{n\times n})$. With these notations, Eq. (22) becomes

$$L(\tilde{\mathbf{v}}^{(r)}, b^{(r)}) = \|(\tilde{\mathbf{v}}^{(r)})^T\mathbf{S}^{(r)} + b^{(r)}\mathbf{e} - \mathbf{y}^{(r)}\|_2^2 + (\tilde{\mathbf{v}}^{(r)})^T\mathbf{F}^{(r)}\tilde{\mathbf{v}}^{(r)}. \tag{28}$$

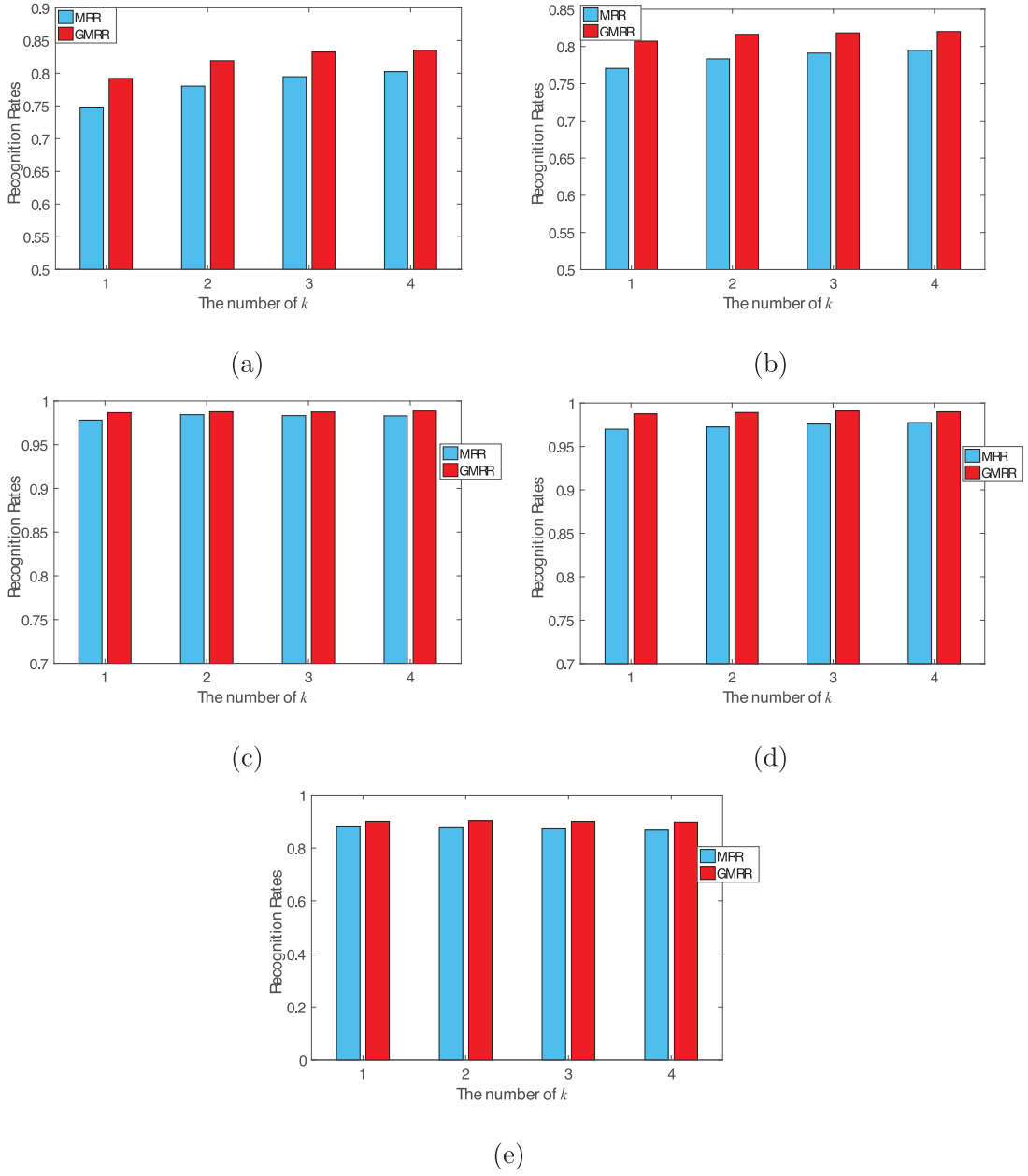

Fig. 4. The impact on different parameters k in GMRR and MRR. (a) PIE. (b) YaleB. (c) Umist. (d) ORL. (e) COIL-100.

where $\mathbf{F}^{(r)} = \alpha\mathbf{D}^{(r)} + \beta\mathbf{E}^{(r)}$ and $\mathbf{S}^{(r)} = [\mathbf{s}_1^{(r)}, \mathbf{s}_2^{(r)}, \ldots, \mathbf{s}_l^{(r)}] \in \mathbb{R}^{nk\times l}$. The optimal solutions should be

$$\bar{\mathbf{v}}^{(r)} = \left[\mathbf{S}^{(r)}\mathbf{H}(\mathbf{S}^{(r)})^T + \alpha\mathbf{F}^{(r)}\right]^{-1}\mathbf{S}^{(r)}\mathbf{H}(\mathbf{y}^{(r)})^T,$$
$$b^{(r)} = \frac{1}{l}\left(\mathbf{y}^{(r)} - (\bar{\mathbf{v}}^{(r)})^T\mathbf{S}^{(r)}\right)\mathbf{e}^T \tag{29}$$

The entire optimization procedure of GMRR is presented in Algorithm 1.

3.3. Classifying new samples

After computing $\{\bar{\mathbf{u}}^r, \bar{\mathbf{v}}^r, b^r\}_{r=1}^c$ yielded by GMRR, we can obtain $\{\mathbf{u}_j^{(r)}\}_{j=1}^k$ and $\{\mathbf{v}_j^{(r)}\}_{j=1}^k$. First of all, we transform all labeled matrix sample $\{\mathbf{X}_i\}_{i=1}^l$ into c-dimensional subspace. Concretely, we compute the c-dimensional regression vector of each labeled matrix samples $\mathbf{X}_i$ as $\mathbf{t}_i = [t_{1i}, t_{2i}, \ldots, t_{ci}]^T$, where $t_{ri} = \sum_{j=1}^k (\mathbf{u}_j^{(r)})^T\mathbf{X}_i\mathbf{v}_j^{(r)} + b^{(r)}$ for $r = 1, 2, \ldots, c$. Here we denote $\mathbf{T} = [\mathbf{t}_1, \mathbf{t}_2, \ldots, \mathbf{t}_l] \in \mathbb{R}^{c\times l}$.

Algorithm 1 Optimization for GMRR.

1. **Input**: Labeled matrix sample set $\{\mathbf{X}_i \in \mathbb{R}^{m\times n} | i = 1, 2, \ldots, l\}$, labeled vectors $\{\mathbf{y}_i \in \mathbb{R}^{c\times 1} | i = 1, 2, \ldots, l\}$, graph Laplacian matrix $\mathbf{L}$, parameters k, σ, α and β.
2. **Initialization**: $\mathbf{V}_0^{(r)} = [\mathbf{I}_{k\times k}, \mathbf{0}_{k\times(n-k)}]^T$ and $\tau = 0$.
3. **Repeat**
   - Update $\bar{\mathbf{u}}_{\tau+1}^{(r)}$ using Eq. (19).
   - Update $\bar{\mathbf{v}}_{\tau+1}^{(r)}$ and $b_{\tau+1}^{(r)}$ using Eq. (29).
   - $\tau = \tau + 1$.
4. **Until** convergence
5. **Output**: $\{\bar{\mathbf{u}}^r, \bar{\mathbf{v}}^r, b^r\}_{r=1}^c = \{\bar{\mathbf{u}}_\tau^{(r)}, \bar{\mathbf{v}}_\tau^{(r)}, b_\tau^{(r)}\}_{r=1}^c$.

When a test matrix sample $\mathbf{X}_{test}$ comes, we compute the c-dimensional regression vector of this test matrix sample $\mathbf{X}_{test}$ as $\mathbf{t} = [t_1, t_2, \ldots, t_c]^T$, where $t_r = \sum_{j=1}^k (\mathbf{u}_j^{(r)})^T\mathbf{X}_{test}\mathbf{v}_j^{(r)} + b^{(r)}$. Finally,

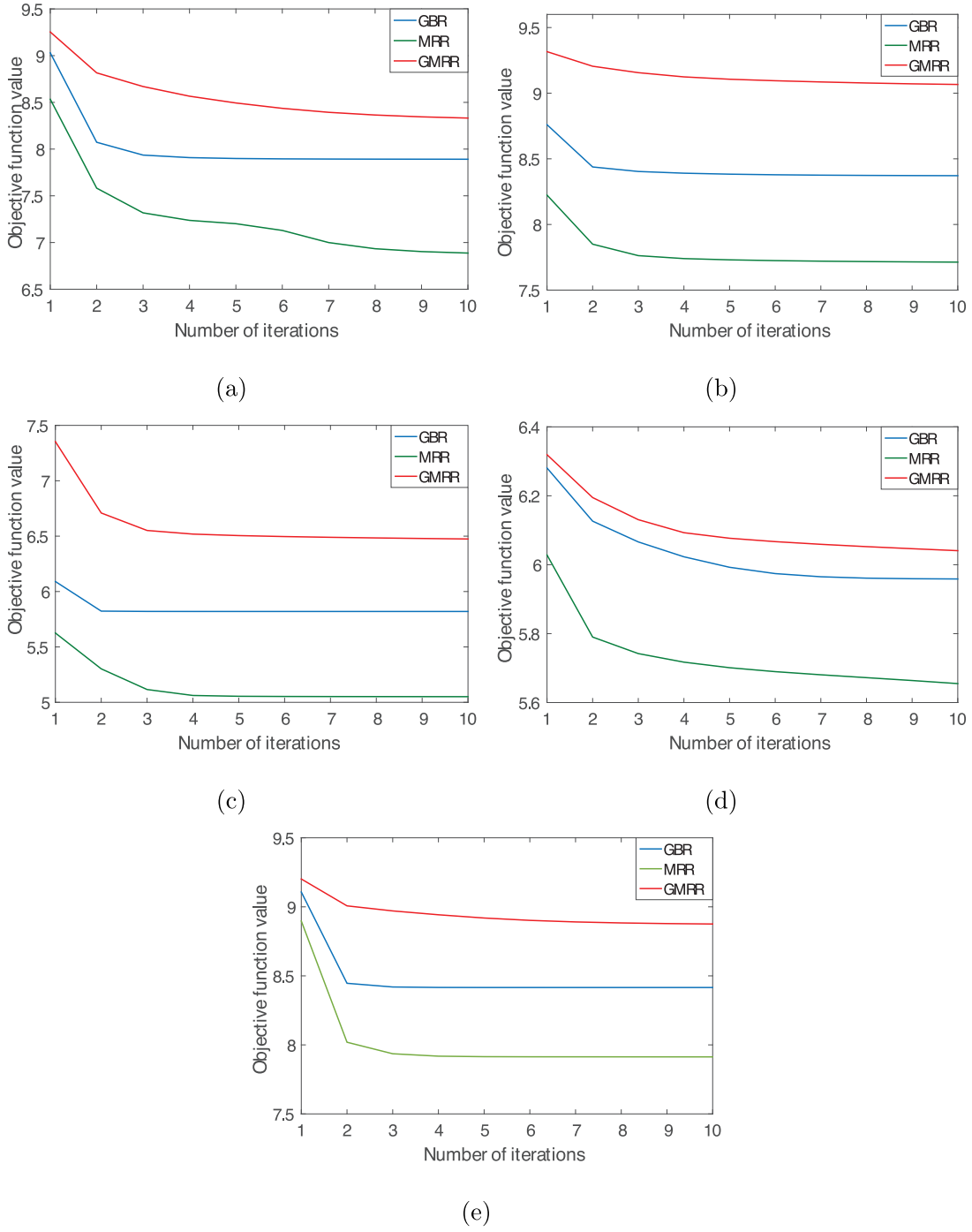

Fig. 5. Convergence curves of GBR, MRR and GMRR on five different databases. (a) PIE. (b) YaleB. (c) Umist. (d) ORL. (e) COIL-100.

we adopt the nearest neighborhood classifier (NN) for classification.

4. Performance analysis

In this section, we will analyze the performance of GMRR in two aspects, i.e., convergence behavior and computational complexity analysis.

4.1. Convergence analysis

For our proposed method, an alternative algorithm of solving the models has been proposed. In order to analyze the convergence behavior, the objective function of GMRR is denoted as $\sum_{r=1}^{c}\mathbf{L}(\mathbf{U}^{(r)},\mathbf{V}^{(r)},b^{(r)})$. Then the following lemma is utilized to show the convergence of GMRR.

Lemma 1. *The alternative optimization algorithm (Algorithm 1) monotonically decreases the value of* $L(\mathbf{U}^{(r)},\mathbf{V}^{(r)},b^{(r)})$.

Proof. Denote the objective function value at the τth iteration as $L(\mathbf{U}_{\tau}^{(r)},\mathbf{V}_{\tau}^{(r)},b_{\tau}^{(r)})$. Through the $(\tau+1)$th iteration, fixing $\mathbf{V}_{\tau}^{(r)}$, we can first obtain $L(\mathbf{U}_{\tau+1}^{(r)},\mathbf{V}_{\tau}^{(r)},b_{\tau}^{(r)})$ via Eq. (19). This subproblem can be represented as $\min_{\mathbf{U}^{(r)}} L(\mathbf{U}^{(r)},\mathbf{V}_{\tau}^{(r)},b_{\tau}^{(r)})$, which is a convex opti-

Table 2
Average recognition rate and standard deviation (%) of different methods on PIE data set with different numbers of labeled samples.

PIE	NN	2DLDA	SMM	GBR	MRR	GMRR
136	12.65 ± 0.81	28.35 ± 1.82	19.38 ± 1.83	31.79 ± 1.28	34.83 ± 1.55	**36.94** ± 1.31
204	16.39 ± 0.56	38.50 ± 1.33	30.66 ± 1.35	43.54 ± 1.62	46.51 ± 1.29	**50.14** ± 1.32
272	19.50 ± 0.48	49.10 ± 2.77	40.19 ± 3.51	52.00 ± 2.00	55.29 ± 2.16	**59.64** ± 2.66
340	22.46 ± 0.68	54.84 ± 1.32	48.45 ± 1.59	57.89 ± 1.31	61.03 ± 0.99	**66.20** ± 1.37
408	25.11 ± 0.65	58.65 ± 1.55	54.25 ± 2.65	62.58 ± 1.33	65.97 ± 1.28	**71.14** ± 0.86
476	27.62 ± 0.29	62.79 ± 0.99	59.19 ± 0.91	66.54 ± 1.13	69.79 ± 1.26	**74.32** ± 0.99
544	29.61 ± 0.70	65.84 ± 1.01	63.60 ± 1.14	69.44 ± 0.90	72.90 ± 1.01	**77.51** ± 0.88
612	32.31 ± 0.97	68.54 ± 1.26	67.09 ± 2.06	71.79 ± 1.14	75.32 ± 1.36	**79.57** ± 1.16
680	34.55 ± 0.67	70.77 ± 1.00	70.31 ± 1.48	74.83 ± 1.72	78.04 ± 1.44	**81.93** ± 1.84
748	36.14 ± 0.88	72.58 ± 0.86	72.08 ± 1.69	76.05 ± 1.24	79.03 ± 1.45	**83.15** ± 1.36

Table 3
Average recognition rate and standard deviation (%) of different methods on YaleB data set with different numbers of labeled samples.

YaleB	NN	2DLDA	SMM	GBR	MRR	GMRR
152	22.79 ± 1.11	**65.78** ± 1.32	53.17 ± 1.57	55.50 ± 1.57	58.54 ± 2.27	62.18 ± 1.80
190	26.55 ± 0.89	**71.38** ± 2.38	60.93 ± 2.83	62.33 ± 2.39	64.88 ± 1.92	68.34 ± 1.63
228	28.52 ± 1.17	**74.45** ± 1.26	66.49 ± 1.63	66.78 ± 1.08	68.84 ± 1.36	72.92 ± 1.60
266	30.88 ± 1.15	**76.76** ± 1.11	70.20 ± 1.40	70.47 ± 1.60	72.33 ± 1.74	75.75 ± 2.10
304	32.05 ± 0.96	**78.84** ± 1.20	72.98 ± 1.18	73.37 ± 1.36	75.12 ± 1.48	78.13 ± 1.46
342	35.43 ± 0.98	**81.06** ± 1.10	75.42 ± 1.51	75.86 ± 1.12	77.00 ± 1.34	80.43 ± 1.07
380	36.48 ± 0.77	81.52 ± 0.97	77.12 ± 0.92	77.06 ± 0.97	78.34 ± 1.25	**81.63** ± 0.84
418	38.00 ± 1.01	**83.16** ± 1.05	78.85 ± 1.09	78.71 ± 1.45	79.88 ± 1.06	82.77 ± 0.77
456	39.79 ± 0.88	83.56 ± 0.95	80.39 ± 1.18	80.44 ± 0.85	81.55 ± 0.94	**84.04** ± 1.05
494	40.59 ± 0.67	84.73 ± 0.88	81.95 ± 0.75	81.59 ± 0.72	82.98 ± 0.64	**85.40** ± 0.84

Table 4
Average recognition rate and standard deviation (%) of different methods on Umist data set with different numbers of labeled samples.

Umist	NN	2DLDA	SMM	GBR	MRR	GMRR
40	57.85 ± 2.51	53.05 ± 3.31	69.93 ± 4.53	67.78 ± 4.71	68.43 ± 4.05	**71.42** ± 4.92
60	70.50 ± 3.38	70.80 ± 5.25	83.50 ± 3.62	81.26 ± 3.51	81.79 ± 3.81	**85.73** ± 2.77
80	78.57 ± 3.75	82.91 ± 1.52	90.24 ± 2.64	90.04 ± 2.84	90.51 ± 2.58	**93.35** ± 1.39
100	81.18 ± 3.00	88.23 ± 1.52	92.95 ± 1.26	91.09 ± 1.95	91.83 ± 1.97	**94.51** ± 2.11
120	86.88 ± 2.91	91.43 ± 1.26	96.22 ± 1.17	93.74 ± 1.93	95.14 ± 1.35	**96.55** ± 1.32
140	89.93 ± 0.54	93.15 ± 1.81	96.97 ± 0.89	95.61 ± 1.36	96.53 ± 1.13	**97.38** ± 1.03
160	91.78 ± 2.40	95.25 ± 1.76	97.54 ± 1.13	97.11 ± 0.98	97.18 ± 1.21	**97.78** ± 1.29
180	94.48 ± 2.15	96.76 ± 1.16	98.20 ± 0.99	98.00 ± 1.01	98.38 ± 0.64	**98.84** ± 0.54
200	95.55 ± 0.97	97.39 ± 0.91	98.56 ± 0.63	97.81 ± 0.89	98.43 ± 0.68	**98.77** ± 0.71
220	94.96 ± 1.04	97.92 ± 0.48	98.62 ± 0.63	97.80 ± 0.59	98.42 ± 0.52	**98.85** ± 0.50

Table 5
Average recognition rate and standard deviation (%) of different methods on ORL data set with different numbers of labeled samples.

ORL	NN	2DLDA	SMM	GBR	MRR	GMRR
80	81.75 ± 2.41	72.63 ± 3.83	84.66 ± 2.60	82.66 ± 2.65	83.22 ± 3.16	**85.09** ± 2.97
120	88.86 ± 1.60	86.96 ± 1.91	92.36 ± 1.01	90.21 ± 2.04	90.86 ± 0.68	**92.71** ± 1.24
160	92.29 ± 1.69	93.08 ± 1.57	95.08 ± 1.29	93.25 ± 2.03	93.92 ± 1.42	**95.25** ± 1.15
200	93.85 ± 2.10	95.15 ± 1.18	95.60 ± 1.91	95.25 ± 1.87	95.65 ± 1.38	**96.85** ± 1.62
240	97.00 ± 1.72	97.37 ± 1.17	98.19 ± 0.95	96.69 ± 0.98	97.19 ± 0.85	**98.31** ± 0.84
280	97.88 ± 1.45	97.75 ± 1.67	98.50 ± 1.23	97.00 ± 2.26	97.25 ± 1.80	**98.92** ± 1.31
320	98.25 ± 2.06	98.12 ± 0.88	98.87 ± 1.09	97.63 ± 1.09	98.13 ± 1.35	**99.00** ± 1.15

Table 6
Average recognition rates and standard deviation (%) of different methods on COIL-100 data set with different numbers of labeled samples.

COIL-100	NN	2DLDA	SMM	GBR	MRR	GMRR
200	59.36 ± 0.97	48.91 ± 1.69	64.55 ± 1.10	63.44 ± 0.96	62.79 ± 1.27	**64.66** ± 0.97
300	64.73 ± 0.80	59.86 ± 0.96	70.69 ± 1.17	69.89 ± 1.50	69.14 ± 1.20	**72.05** ± 1.29
400	69.97 ± 0.88	66.02 ± 1.46	76.24 ± 0.96	75.58 ± 1.16	74.96 ± 1.09	**77.87** ± 1.30
500	73.29 ± 0.52	68.98 ± 0.95	79.47 ± 0.77	79.02 ± 0.68	78.37 ± 0.86	**81.52** ± 0.64
600	75.70 ± 0.56	72.39 ± 1.14	81.89 ± 0.81	81.04 ± 1.05	80.67 ± 1.07	**83.75** ± 1.14
700	78.38 ± 0.88	74.03 ± 1.77	84.20 ± 0.83	83.51 ± 0.74	83.10 ± 0.68	**86.20** ± 0.90
800	80.12 ± 0.33	77.00 ± 1.13	86.25 ± 0.51	85.03 ± 0.32	84.77 ± 0.41	**87.73** ± 0.75
900	82.09 ± 0.63	78.27 ± 1.00	87.97 ± 0.40	87.17 ± 0.60	86.72 ± 0.56	**89.34** ± 0.51
1000	83.42 ± 0.51	79.76 ± 0.62	89.02 ± 0.57	88.02 ± 0.55	87.71 ± 0.52	**90.35** ± 0.54
1100	84.97 ± 0.64	81.09 ± 0.86	90.31 ± 0.62	89.09 ± 0.61	88.95 ± 0.69	**91.11** ± 0.44
1200	86.24 ± 0.51	81.86 ± 1.09	91.23 ± 0.64	90.06 ± 0.56	89.91 ± 0.57	**91.89** ± 0.70

Table 7
Computational time (second) of different methods.

PIE	2DLDA	SMM	GBR	MRR	GMRR
136	**0.0927** ± 0.0019	22.7870 ± 0.2666	0.8166 ± 0.0296	1.0798 ± 0.0395	3.4741 ± 0.0382
204	**0.1216** ± 0.0039	24.8388 ± 0.2585	1.2016 ± 0.0337	1.5035 ± 0.0159	3.9631 ± 0.0193
272	**0.1519** ± 0.0041	27.1337 ± 0.1389	1.6092 ± 0.0674	2.2535 ± 0.1066	4.4628 ± 0.0290
340	**0.1801** ± 0.0051	29.2162 ± 0.2068	2.0700 ± 0.0453	2.5395 ± 0.0304	5.0922 ± 0.1983
408	**0.2111** ± 0.0035	31.4187 ± 0.2167	2.4474 ± 0.0273	3.1925 ± 0.0520	5.7806 ± 0.0565
YaleB	2DLDA	SMM	GBR	MRR	GMRR
152	**0.0624** ± 0.0139	8.9532 ± 0.0997	0.4585 ± 0.0141	0.5716 ± 0.0196	0.8725 ± 0.0182
190	**0.0679** ± 0.0019	9.1347 ± 0.0546	0.5523 ± 0.0141	0.6718 ± 0.0090	1.0154 ± 0.0173
228	**0.0785** ± 0.0014	9.4412 ± 0.0805	0.6737 ± 0.0157	0.8013 ± 0.0114	1.1396 ± 0.0171
266	**0.0914** ± 0.0036	10.0531 ± 0.0456	0.7721 ± 0.0204	0.9516 ± 0.0565	1.2956 ± 0.0270
304	**0.1037** ± 0.0035	10.1750 ± 0.0561	0.8867 ± 0.0311	1.0911 ± 0.0305	1.4725 ± 0.0472
Umist	2DLDA	SMM	GBR	MRR	GMRR
40	**0.0261** ± 0.0077	4.3580 ± 0.1726	0.0865 ± 0.0042	0.1223 ± 0.0058	0.3832 ± 0.0117
60	**0.0335** ± 0.0046	4.4054 ± 0.2817	0.1178 ± 0.0090	0.1519 ± 0.0037	0.4262 ± 0.0141
80	**0.0371** ± 0.0014	4.6139 ± 0.1651	0.1462 ± 0.0066	0.1861 ± 0.0050	0.4635 ± 0.0308
100	**0.0441** ± 0.0017	4.7056 ± 0.1268	0.1845 ± 0.0127	0.2197 ± 0.0092	0.4988 ± 0.0147
120	**0.0517** ± 0.0023	4.8498 ± 0.1503	0.2024 ± 0.0125	0.2481 ± 0.0043	0.5400 ± 0.0139
ORL	2DLDA	SMM	GBR	MRR	GMRR
80	**0.0568** ± 0.0020	12.2900 ± 0.2074	0.3099 ± 0.0150	0.4005 ± 0.0061	1.7982 ± 0.0223
120	**0.0718** ± 0.0017	12.9917 ± 0.1160	0.4273 ± 0.0224	0.5447 ± 0.0074	1.9487 ± 0.0305
160	**0.0883** ± 0.0024	13.5245 ± 0.1600	0.5519 ± 0.0086	0.7124 ± 0.0159	2.1355 ± 0.0442
200	**0.1070** ± 0.0030	14.0613 ± 0.0915	0.6926 ± 0.0258	0.8602 ± 0.0101	2.3785 ± 0.0562
240	**0.1241** ± 0.0035	14.6069 ± 0.1592	0.8296 ± 0.0308	1.0305 ± 0.0.0140	2.5239 ± 0.0229
COIL-100	2DLDA	SMM	GBR	MRR	GMRR
200	**0.2571** ± 0.0424	35.5669 ± 0.4876	1.9429 ± 0.0921	2.4763 ± 0.0601	6.3591 ± 0.0518
300	**0.1599** ± 0.0102	40.3820 ± 0.2170	2.8989 ± 0.0412	3.8264 ± 0.1552	8.0227 ± 0.1120
400	**0.1903** ± 0.0092	44.2715 ± 1.6451	3.9260 ± 0.0226	5.0652 ± 0.0917	9.1712 ± 0.1906
500	**0.2251** ± 0.0069	49.8822 ± 0.7281	5.0571 ± 0.2382	6.2633 ± 0.0650	10.4869 ± 0.2426
600	**0.2571** ± 0.0036	60.1879 ± 4.9278	6.2463 ± 0.0438	8.0539 ± 0.1163	12.0182 ± 0.1152

mization problem. We have

$$L(\mathbf{U}_{\tau+1}^{(r)}, \mathbf{V}_{\tau}^{(r)}, b_{\tau}^{(r)}) \le L(\mathbf{U}_{\tau}^{(r)}, \mathbf{V}_{\tau}^{(r)}, b_{\tau}^{(r)}) \tag{30}$$

Next, fixing $\mathbf{U}_{\tau+1}^{(r)}$, we can obtain $L(\mathbf{U}_{\tau+1}^{(r)}, \mathbf{V}_{\tau+1}^{(r)}, b_{\tau+1}^{(r)})$ via Eq. (29). The subproblem becomes $\min_{\mathbf{V}^{(r)}, b^{(r)}} L(\mathbf{U}^{(r)}, \mathbf{V}_{\tau+1}^{(r)}, b_{\tau+1}^{(r)})$, which is also a convex optimization problem. We have

$$L(\mathbf{U}_{\tau+1}^{(r)}, \mathbf{V}_{\tau+1}^{(r)}, b_{\tau+1}^{(r)}) \le L(\mathbf{U}_{\tau+1}^{(r)}, \mathbf{V}_{\tau}^{(r)}, b_{\tau}^{(r)}) \tag{31}$$

Combining (30) with (31), we have the following conclusion

$$0 \le L(\mathbf{U}_{\tau+1}^{(r)}, \mathbf{V}_{\tau+1}^{(r)}, b_{\tau+1}^{(r)}) \le L(\mathbf{U}_{\tau}^{(r)}, \mathbf{V}_{\tau}^{(r)}, b_{\tau}^{(r)}) \tag{32}$$

That completes the proof. □

Since the objective function of GMRR is $\sum_{r=1}^{c} \mathbf{L}(\mathbf{U}^{(r)}, \mathbf{V}^{(r)}, b^{(r)})$ and these c training procedures are separate, hence Algorithm 1 can confirm that GMRR converges according to the monotone convergence theorem.

4.2. Computational complexity analysis

Let l be the number of labeled matrix samples, m and n be the first and second dimensionality of the matrix sample, and c be the number of classes. We analyze the computational complexity of Algorithm 1. Here, the big O notation is used to represent the computational complexity. Before conducting Algorithm 1, we need to compute the graph weight matrix $\mathbf{W}$. Its computational complexity is $O(l^2mn)$. In Algorithm 1, we need to solve two regularized least square regression problems with the dimensionality mk and nk, respectively. Therefore, its computational complexity is $O(\tau(m^2+n^2)k^2)$, where τ is the number of iterations. Hence, the overall time complexity of solving GMRR is $O(\tau c(m^2+n^2)k^2 + l^2mn)$.

5. Experiments and discussion

In this section, we use several popular data sets to test and verify the effectiveness of GMRR.

5.1. Data set description and experimental setup

Before conducting the experiments, we first describe these five databases used in this paper.

- The CMU PIE face database [32] contains 68 subjects with 41,368 face images as a whole. The face images were captured by 13 synchronized cameras and 21 flashes, under varying pose, illumination, and expression. We used 170 face images for each individual in our experiment and each image is resized to have 32×32 pixels.
- The Extended YaleB data set [33] consists of 2414 single frontal facial images of 38 individuals. These images are taken with varying illumination conditions. In our experiments, each image is resized to have 24×21 pixels.
- The Umist Face Database [34] consists of 575 images of 20 people. Each covers a range of poses from profile to frontal views. Subjects cover a range of race/sex/appearance. The images are cropped to size 28×23.
- The ORL data set [35] includes 400 face images from 40 subjects, each provided with 10 face images. For some subjects, the images are taken at different times, varying the lighting, facial expressions (open/closed eyes, smiling/not smiling) and facial details (glasses / no glasses). In our experiments, each image is resized to have 32×32 pixels.
- Columbia Object Image Library (COIL-100) [36] is a database of images of 100 objects. The objects are placed on a motorized turntable against a black background. Images of the objects are taken at pose intervals of 5 degrees. This corresponds to 72

poses per object. In our experiments, each image is resized to have 32×32 pixels.

Fig. 1 shows some examples of images from these databases and the detailed statistical characters of different data sets are listed in Table 1. To test the effectiveness of GMRR, we consider the competing matrix-based methods including 2DLDA [7], SMM [9], GBR and MRR. We also adopt the NN classifier to conduct the classification on the PCA features with keeping 99% energy of eigenvalues as the benchmark. To avoid any bias, we repeat 10 times to randomly select the labeled matrix samples in all experiments and report the average recognition (standard deviation) to compare these methods. Before conducting classification, we need to describe the parameter settings of these methods. For GBR and MRR, the optimal parameter α is picked from the set $\{5, 10, \ldots, 95, 100\}$ by cross-validation. After the optimal α is chosen in MRR, we fix α and select the parameter β in GMRR from $\{0, 5, 10, \ldots, 95, 100\}$ by cross-validation. The purpose of this parameter setting is to highlight the effect on adding the graph regularization term. Fig. 2 shows the recognition rates in different parameters α of MRR in several databases. After α is selected and fixed, Fig. 3 shows the recognition rates in different parameters β of GMRR in several databases. From observing Fig. 3, one can see that the optimal value of β is nonzero, which states that adding the graph regularization can improve the learning performance.

5.2. Comparison between GMRR and other methods

Tables 2–6 list the average recognition rate and the standard deviation (std) of each method in different data sets. From these results, we find that as the increase of labeled samples, all methods also have increasing trends. This is consistent with intuition since we have more information for learning. Compared with NN, these matrix-based methods such as 2DLDA, SMM, GBR and MRR can obtain higher accuracies in most case. It states that these methods can effectively extract the features from the images for classification. However, in COIL-100, the classification results of NN are better than 2DLDA. This states that 2DLDA does not always helpful for the next classification. Compared with MRR, our GMRR achieves better classification performance. This states that GMRR exploits the discriminating structure of the data space by incorporating a class compactness graph regularizer, which can help to improve the learning ability of MRR. Overall, GMRR achieves the best performance among these methods.

Fig. 4 shows the impact of the number of k in different databases. From observing the results in Fig. 4, one can see that as the increase of k, the classification accuracy does not always increase consistently. This is because k is used to balance the capacity of learning and generalization for the regression model in MRR and GMRR. Moreover, we find that GMRR always obtains the better results than MRR under different k, which confirms the effectiveness of GMRR. From these results in Fig. 4, we determine $k = 2$ in all experiments, which is the same in [23].

5.3. Algorithmic convergence and computational cost

Here, we experimentally verify the convergence of our proposed GMRR, where its objective function is denoted as $J = \sum_{r=1}^{c} \mathbf{L}(\mathbf{U}^{(r)}, \mathbf{V}^{(r)}, b^{(r)})$. Fig. 5 shows the convergence curves of the objective function value on four databases. From seeing the results in Fig. 5, one can see that the objective function value of GMRR can well converge. It confirms the effectiveness of our proposed algorithm to optimize the objective function of GMRR.

Subsequently, we compare the computational cost of different models. All methods are implemented in Matlab. The experiments are carried out on 64-bit, 3.20 GHz In- tel i5-6500 (4 core) CPU computer with 8-GB memory. Since all methods use NN classifier for classification, we don't compute the computational times of NN. Table 7 shows the computational time of different methods. From seeing Table 7, we find that as the increase of labeled samples, all methods spend more time. Since our method needs to construct a class compactness graph, GMRR costs a little more time than MRR.

6. Conclusion

In this paper, we propose a graph-based multiple rank regression model (GMRR), which employs multiple-rank left and right projecting vectors to regress each matrix data set to its label for each category. To exploit the discriminating structure of the data space, we construct a class compactness graph to constrain these left and right projecting vectors. The core of GMRR is to exploit the discriminating structure to improve the learning performance of MRR. Extensive experimental results on image classification have demonstrated the effectiveness of GMRR.

Since GMRR adopts the matrix data as the inputs for classification, this paper can only handle the raw features of images. However, for some challenging databases such as Caltech101/256 and VOC2012, the samples from each category have significant variability. The raw features as the inputs will degrade the performances. To address this issue, many hand-crafted features and deep features are proposed. Hence, how to improve our method to fit for these features is our future work.

Acknowledgments

This research work was supported in part by the Guangdong University of Technology, Guangzhou, China under Grant from the Financial and Education Department of Guangdong Province 2016[202]: Key Discipline Construction Programme, in part by the Education Department of Guangdong Province: New and Integrated Energy System Theory and Technology Research Group (Grant No. 2016KCXTD022), and in part by the Foundation for Distinguished Young Talents in Higher Education of Guangdong (Grant No. 2016KQNCX045).

References

[1] A. Jain, R. Duin, J. Mao, Statistical pattern recognition: a review, IEEE Trans. Pattern Anal. Mach. Intell. 22 (1) (2000) 4–37.
[2] M. Cristianini, J. Shawe-Taylor, An Introduction to Support Vector Machines, Cambridge University Press, Cambridge, U.K., 2000.
[3] D.W. Hosmer, S. Lemeshow, Applied logistic regression, Chichester, Wiley, New York, 2000.
[4] S. Xiang, F. Nie, G. Meng, C. Pan, C. Zhang, Discriminative least squares regression for multiclass classification and feature selection, IEEE Trans. Neural Netw. Learn. Syst. 23 (11) (2012) 1738–1754.
[5] J. Wright, A. Yang, A. Ganesh, S. Sastry, Y. Ma, Robust face recognition via sparse representation, IEEE Trans. Pattern Anal. Mach. Intell. 31 (2) (2009) 210–227.
[6] L. Zhang, M. Yang, X. Feng, Sparse representation or collaborative representation: Which helps face recognition? in: Proceedings of the ECCV, 2011, pp. 471–478.
[7] J. Ye, R. Janardan, Q. Li, Two-dimensional linear discriminant analysis, in: L.K. Saul, Y. Weiss, L. Bottou (Eds.), Proceedings of the Advances in Neural Information Processing Systems, 17, 2005, pp. 1569–1576.
[8] H. Hung, C. Wang, Matrix variate logistic regression model with application to eeg data, Biostatistics 14 (1) (2013) 189–202.
[9] L. Luo, Y. Xie, Z. Zhang, W.-J. Li, Support matrix machines, in: Proceedings of the 32Nd International Conference on International Conference on Machine Learning, 37, pp. 938–947.
[10] W. Chu, Y. Hu, C. Zhao, H. Liu, D. Cai, Atom decomposition based subgradient descent for matrix classification, Neurocomputing 205 (2016) 222–228.
[11] Q. Zheng, F. Zhu, J. Qin, B. Chen, P.-A. Heng, Sparse support matrix machine, Pattern Recognit. 76 (2018) 715–726.
[12] J. Yang, L. Luo, J. Qian, Y. Tai, F. Zhang, Y. Xu, Nuclear norm based matrix regression with applications to face recognition with occlusion and illumination changes, IEEE Trans. Pattern Anal. Mach. Intell. 39 (1) (2017) 156–171.

[13] J. Xie, J. Yang, J.J. Qian, Y. Tai, H.M. Zhang, Robust nuclear norm-based matrix regression with applications to robust face recognition, IEEE Trans. Image Process. 26 (5) (2017) 2286–2295.
[14] J. Chen, J. Yang, L. Luo, J. Qian, W. Xu, Matrix variate distribution-induced sparse representation for robust image classification, IEEE Trans. Neural Netw. Learn. Syst. 26 (10) (2015) 2291–2300.
[15] Q. Zheng, F. Zhu, J. Qin, P.-A. Heng, Multiclass support matrix machine for single trial eeg classification, Neurocomputing 275 (2018) 869–880.
[16] Z. Lai, Y. Xu, J. Yang, J. Tang, D. Zhang, Sparse tensor discriminant analysis, IEEE Trans. Image Process. 22 (10) (2013) 3904–3915.
[17] W. Hu, J. Gao, J. Xing, C. Zhang, S. Maybank, Semi-supervised tensor-based graph embedding learning and its application to visual discriminant tracking, IEEE Trans. Pattern Anal. Mach. Intell. 39 (1) (2017) 172–188.
[18] W. Guo, I. Kotsia, I. Patras, Tensor learning for regression, IEEE Trans. Image Process. 21 (2) (2012) 816–827.
[19] N. Qi, Y. Shi, X. Sun, J. Wang, B. Yin, J. Gao, Multi-dimensional sparse models, IEEE Trans. Pattern Anal. Mach. Intell. 40 (1) (2018) 163–178.
[20] D. Tao, X. Li, X. Wu, S.J. Maybank, General tensor discriminant analysis and Gabor features for gait recognition, IEEE Trans. Pattern Anal. Mach. Intell. 29 (10) (2007) 1700–1715.
[21] C.M. Bishop, Pattern Recognition and Machine Learning, Springer-Verlag, Secaucus, NJ, 2006.
[22] K.R. Gabriel, Generalised bilinear regression, Biometrika 85 (3) (1998) 689–700.
[23] C. Hou, F. Nie, D. Yi, Y. Wu, Efficient image classification via multiple rank regression, IEEE Trans. Image Process. 22 (1) (2013) 340–352.
[24] S. Roweis, L. Saul, Nonlinear dimensionality reduction by locally linear embedding, Science 290 (5500) (2000) 2323–2326.
[25] M. Belkin, P. Niyogi, Laplacian eigenmaps and spectral techniques for embedding and clustering, in: Proceedings of the NIPS, 2001, pp. 585–592.
[26] J. Tenenbaum, V. Silva, J. Langford, A global geometric frame-work for nonlinear dimensionality reduction, Science 290 (5500) (2000) 2319–2323.
[27] D. Cai, X. He, J. Han, T.S. Huang, Graph regularized nonnegative matrix factorization for data representation, IEEE Trans. Pattern Anal. Mach. Intell. 33 (8) (2011) 1548–1560.
[28] P. Li, J. Bu, L. Zhang, C. Chen, Graph-based local concept coordinate factorization, Knowl. Inf. Syst. 43 (1) (2015) 103–126.
[29] M. Zheng, J. Bu, C. Chen, C. Wang, L. Zhang, G. Qiu, D. Cai, Graph regularized sparse coding for image representation, IEEE Trans. Image Process. 20 (5) (2011) 1327–1336.
[30] X. Lu, H. Wu, Y. Yuan, P. Yan, X. Li, Manifold regularized sparse NMF for hyperspectral unmixing, IEEE Trans. Geosci. Remote Sens. 51 (5) (2013) 2815–2826.
[31] P. Li, J. Yu, M. Wang, L. Zhang, D. Cai, X. Li, Constrained low-rank learning using least squares-based regularization, IEEE Trans. Cybern. 47 (12) (2017) 4250–4262.
[32] T. Sim, S. Baker, M. Bsat, The CMU pose, illumination, and expression database, IEEE Trans. Pattern Anal. Mach. Intell. 25 (12) (2003) 1615–1618.
[33] A. Georghiades, P. Belhumeur, D. Kriegman, From few to many: illumination cone models for face recognition under variable lighting and pose, IEEE Trans. Pattern Anal. Mach. Intell. 23 (6) (2001) 643–660.
[34] D. Graham, N. Allison, Characterizing virtual eigensignatures for general purpose face recognition, Face Recognit. Theory Appl. 163 (1998) 446–456.
[35] F. Samaria, A. Harter, Parameterisation of a stochastic model for human face identification, in: Proceedings of the 2nd IEEE Workshop Applications of Computer Vision, Sarasota, FL, 1994, pp. 138–142.
[36] S.A. Nene, S.K. Nayar, H. Murase, object image library (COIL-100), Technical Report, Columbia University, 1996.

Haoliang Yuan received the B.Eng. and M.Sc. degrees from the Hubei University, Wuhan, China, in 2009 and 2012, and the Ph.D. degree from the University of Macau, 2016. Currently, he is an assistant professor in the school of Automation at Guangdong University of Technology.

Junyu Li received the B.Eng. degrees from Guangdong University of Technology, Guangzhou, China, in 2015. Currently, he is working toward his M.Sc. degree at School of Automation, Guangdong University of Technology, China.

Loi Lei Lai received B.Sc., Ph.D. and D.Sc. from University of Aston and City, University of London respectively. Presently he is University Distinguished Professor at Guangdong University of Technology, China. He was Director of Research and Development Centre, Pao Yue Kong Chair Professor, Vice President and Professor & Chair in Electrical Engineering for State Grid Energy Research Institute, China; Zhejiang University, China; IEEE Systems, Man and Cybernetics Society (IEEE/SMCS) and City, University of London respectively. He was awarded an IEEE Third Millennium Medal, IEEE Power and Energy Society (IEEE/PES) UKRI Power Chapter Outstanding Engineer Award in 2000, IEEE/PES Energy Development and Power Generation Committee Prize Paper in 2006 & 2009, IEEE/SMCS Outstanding Contribution Award in 2013 & 2014 and the Most Active Technical Committee Award in 2016. He is a Fellow of IEEE and IET &National Distinguished Expert in China and Distinguished Expert in State Grid Corporation of China.

Yuan Yan Tang is currently a Chair Professor with the Faculty of Science and Technology, University of Macau, and a Professor/an Adjunct Professor/a Honorary Professor with several institutes, including Chongqing University, China; Concordia University, Canada; Hong Kong Baptist University, Hong Kong; and the Advanced Innovation Center for Big Data and Brain Computing, Beihang University. He has authored over 400 academic papers and has authored or co-authored over 25 monographs/books/book chapters. His current interests include wavelets, pattern recognition, image processing, and artificial intelligence. He is a fellow of the International Associate of Pattern Recognition (IAPR). He is the Founder and the Editor-in-Chief of the International Journal on Wavelets, Multiresolution, and Information Processing and an associate editor of several international journals. He is the Founder and the Chair of the Pattern Recognition Committee in the IEEE SMC. He served as the general chair, the program chair, and a committee member for many international conferences. He is the Founder and the General Chair of the series International Conferences on Wavelets Analysis and Pattern Recognition. He is the Founder and the Chair of the Macau Branch of the IAPR.