Linear Manifold Regularization with Adaptive Graph for Semi-supervised Dimensionality Reduction

Kai Xiong[1], **Feiping Nie**[1,2], **Junwei Han**[1]
[1]Northwestern Ploytechnical University, Xi'an, 710072, P. R. China
[2]University of Texas at Arlington, USA
{bearkai1992, feipingnie, junweihan2010}@gmail.com

Abstract

Many previous graph-based methods perform dimensionality reduction on a pre-defined graph. However, due to the noise and redundant information in the original data, the pre-defined graph has no clear structure and may not be appropriate for the subsequent task. To overcome the drawbacks, in this paper, we propose a novel approach called linear manifold regularization with adaptive graph (LMRAG) for semi-supervised dimensionality reduction. LMRAG directly incorporates the graph construction into the objective function, thus the projection matrix and the adaptive graph can be simultaneously optimized. Due to the structure constraint, the learned graph is sparse and has clear structure. Extensive experiments on several benchmark datasets demonstrate the effectiveness of the proposed method.

1 Introduction

Dimensionality reduction is a significant topic in machine learning and other related fields. It is reasonable to presume that the naturally generated high dimensional data have a much more compact description, *i.e.*, the high dimensional data probably lie on or close to a smooth low dimensional manifold [Roweis and Saul, 2000]. The goal of dimensionality reduction is to remove the noise and redundant information, and at the same time to preserve the desired intrinsic information of the input data. Collecting the labeled data is usually costly, while the unlabeled data are abundant and can be easily obtained. Therefore, semi-supervised dimensionality reduction has attracted great interest in recent years.

If we do not have more information than similarities between data points, a nice way to represent the data is in the form of a graph [Zhang *et al.*, 2014; Liu *et al.*, 2010], which aims to capture the intrinsic geometric structure of data manifold. There have been many graph-based methods for dimensionality reduction. To provide a unified perspective of various algorithms, [Yan *et al.*, 2007] proposed a general framework known as graph embedding, in which the algorithms such as LLE [Roweis and Saul, 2000], LE [Belkin and Niyogi, 2001] and LPP [He *et al.*, 2005] share the common formulation with different graph design. To better cope with the data sampled from nonlinear manifold, [Nie *et al.*, 2010] proposed the flexible manifold embedding (FME) framework for semi-supervised and unsupervised dimensionality reduction. There are many other semi-supervised graph-based methods that were developed with different prior assumptions [He *et al.*, 2008; Gao *et al.*, 2015; Chatpatanasiri and Kijsirikul, 2010] or by label propagation [Nie *et al.*, 2009]. By adding a graph regularization term, some supervised dimensionality reduction methods can also be extended to the semi-supervised case [Cai *et al.*, 2007; Song *et al.*, 2008; Huang *et al.*, 2012].

All the graph-based methods mentioned above need to construct a graph beforehand. Therefore, graph construction is a crucial step for these methods, since their performance highly relies on how well the graph models the intrinsic structure of data manifold. In general, one can construct an adjacency graph by k-nearest neighbor or ε-ball neighborhood criteria. The edge weights are then assigned by Gaussian kernel or local linear reconstruction. However, due to the noise and redundant information, such a pre-defined graph has no clear structure and may not be appropriate for the subsequent dimensionality reduction task.

To overcome the drawbacks, it is natural for us to consider how to learn an adaptive graph which is the optimal one for dimensionality reduction. The adaptive graph should better be sparse and have clear structure that the number of connected components in the graph is exactly the number of data clusters/classes. Such a structured graph would be benifical to many tasks since it contains more accurate information of the data. Motivated by these ideas, we propose a novel approach called linear manifold regularization with adaptive graph (LMRAG) for semi-supervised dimensionality reduction. It is worthwhile to highlight the main contributions of the paper as follows:

1. LMRAG performs dimensionality reduction and graph construction simultaneously, by incorporating the adaptive neighbor learning into the objective function of linear Laplacian regularized least squares (LapRLS/L). Both the optimal graph and the projection matrix can then be obtained.
2. To learn an adaptive graph that has clear structure, a structure constraint is imposed to the graph Laplacian. To the best of our knowledge, it is the first time to introduce an adaptive and structured graph for semi-supervised

dimensionality reduction.

3. A simple yet effective algorithm is developed for our new model. Extensive experiments on several widely used datasets demonstrate the effectiveness of the proposed method.

2 Background

We first introduce some notations used throughout the paper. For a matrix $W \in \mathbb{R}^{m\times n}$, the (i,j)-th entry and the i-th column are denoted by w_{ij} and w_i, respectively. The trace and Frobunius norm of W are denoted by $Tr(W)$ and $\|W\|_F$, respectively. The p-norm of vector v is denoted by $\|v\|_p$, and $I_k \in \mathbb{R}^{k\times k}$ is an identity matrix. $\mathbf{1} \in \mathbb{R}^{n\times 1}$ is a vector with all entries being 1.

The data matrix is denoted by $X \in \mathbb{R}^{d\times n}$ ($n = l+u$), where the first l samples $\{x_i\}_{i=1}^{l}$ are labeled and the last u samples $\{x_i\}_{i=l+1}^{n}$ are unlabeled. c is the number of data classes. The label matrix $Y \in \mathbb{R}^{c\times n}$ is defined as $y_{ji} = 1$ if x_i has label $j \in \{1, 2, \ldots, c\}$ and $y_{ji} = 0$, otherwise. Let $G = \{X, S\}$ be an undirected and weighted graph, in which X is viewed as the vertex set and $S \in \mathbb{R}^{n\times n}$ is the similarity matrix. The entry s_{ij} measures the similarity between x_i and x_j. The graph Laplacian is then defined as $L = D - S$, where the diagonal matrix D has entry $d_{ii} = \sum_{j=1}^{n} s_{ij}$ $(i = 1, \ldots, n)$.

2.1 Linear Manifold Regularization

Manifold regularization [Belkin *et al.*, 2006; Sindhwani *et al.*, 2005a; 2005b] is a widely used geometric framework that brings together three distinct concepts from the theory of regularization in reproducing kernel Hilbert spaces (RKHS), manifold learning and spectral methods. It has successfully extended linear regression and support vector machine (SVM), respectively, to the semi-supervised learning methods Laplacian regularized least squares (LapRLS) and Laplacian SVM. We take LapRLS/L as an example to briefly introduce the linear manifold regularization. The formulation of LapRLS/L is as follows:

$$\min_{W,b} \gamma_A \|W\|_F^2 + \gamma_I Tr(W^T X L X^T W) + \frac{1}{l}\sum_{i=1}^{l} \|W^T x_i + b - y_i\|^2, \quad (1)$$

where $W \in \mathbb{R}^{d\times c}$ is the projection matrix and $b \in \mathbb{R}^{c\times 1}$ is the bias term. The third term is the label fitness term. γ_A, γ_I are two regularization parameters that control the RKHS norm and the intrinsic norm, respectively.

2.2 Adaptive Neighbor Learning

We consider the probabilistic neighbors to learn the similarity matrix. The probability of two data points to be neighbor can be regarded as their similarity [Nie *et al.*, 2014]. It is natural to presume that a smaller distance should be assigned a larger probability, and vice versa. For simplicity, we adopt the Euclidean distance. Therefore, we can adaptively determine the probabilities by solving the following problem:

$$\min_{s_i^T \mathbf{1}=1, 0\le s_{ij}\le 1} \sum_{i,j=1}^{n} (\|x_i - x_j\|_2^2 s_{ij} + \gamma s_{ij}^2), \quad (2)$$

where $\gamma > 0$ is the regularization parameter, and $s_i \in \mathbb{R}^{n\times 1}$ is a vector with the j-th entry as s_{ij}. The regularization term s_{ij}^2 is used to avoid the trivial solution that the nearest neighbor has the probability of 1 while the others are all 0. We do not consider x_i itself is the neighbor of x_i. For the diagonal entries of S, we simply have $\{s_{ii} = 0\}_{i=1}^{n}$. In Eq.(2), we can measure the distance in the projected space by replacing x_i, x_j with $W^T x_i$, $W^T x_j$, respectively. Moreover, we enforce S to be symmetric by $(S^T + S)/2$.

3 The Proposed Method

3.1 Formulation

It is not difficult to verify that

$$Tr(W^T X L X^T W) = \frac{1}{2}\sum_{i,j=1}^{n} \|W^T x_i - W^T x_j\|_2^2 s_{ij}. \quad (3)$$

Based on Eq.(1) to Eq.(3), by incorporating the adaptive neighbor learning into the objective function of LapRLS/L, the proposed LMRAG is formulated as follows:

$$\begin{aligned} &\min_{W,b,S} \sum_{i,j=1}^{n} \|W^T x_i - W^T x_j\|_2^2 s_{ij} + \gamma \|S\|_F^2 + \beta \|W\|_F^2 \\ &+ \alpha Tr(W^T X + b\mathbf{1}^T - Y) U (W^T X + b\mathbf{1}^T - Y)^T, \\ &s.t. \quad S \ge 0, S^T \mathbf{1} = \mathbf{1} \end{aligned} \quad (4)$$

where α, β and γ are three trade-off parameters. The fourth term is the label fitness term. U is a diagonal matrix with the first l and the last u diagonal entries being 1 and 0, respectively.

By solving Eq.(4), we can learn an adaptive graph while in most cases all the data points are in one connected component. According to [Mohar *et al.*, 1991], the multiplicity c of eigenvalue 0 of the graph Laplacian matrix is equal to the number of connected components in the graph. Therefore, to make the adaptive graph structured, we can add a structure constraint by restricting the rank of L to be $(n - c)$.

However, it is challenge to directly solve the problem of Eq.(4) with the rank constraint. Suppose $\sigma_i(L)$ is the i-th smallest eigenvalue of L, we can transform the rank constraint to the sum of the first c smallest eigenvalues. Note that $\sigma_i(L) \ge 0$, since L is positive semi-definite. The objective function of LMRAG then becomes:

$$\begin{aligned} &\min_{W,b,S} \sum_{i,j=1}^{n} \|W^T x_i - W^T x_j\|_2^2 s_{ij} + \gamma \|S\|_F^2 + \beta \|W\|_F^2 \\ &+ \alpha Tr(W^T X + b\mathbf{1}^T - Y) U (W^T X + b\mathbf{1}^T - Y)^T \\ &+ 2\lambda \sum_{i=1}^{c} \sigma_i(L), \qquad s.t. \quad S \ge 0, S^T \mathbf{1} = \mathbf{1} \end{aligned} \quad (5)$$

As we can see, for a large enough λ, solving Eq.(5) will make $\sum_{i=1}^{c} \sigma_i(L)$ get infinitely close to zero, then the rank constraint is approximately satisfied. Such a relaxation is beneficial to the subsequent optimization, while Eq.(4) with the rank constraint is hard to tackle.

Further, we have the following equation:

$$\sum_{i=1}^{c} \sigma_i(L) = \min_{F\in\mathbb{R}^{c\times n}, FF^T = I_c} Tr(FLF^T), \quad (6)$$

where the optimal F is formed by eigenvectors of L corresponding to the first c smallest eigenvalues as rows. $f_i \in \mathbb{R}^{c\times 1}$ can be seen as a kind of embedding of x_i. The term on the right hand side of Eq.(6) is actually the objective function of spectral clustering [Von Luxburg, 2007]. Therefore, our final objective function is formulated as follows:

$$\begin{aligned}&\min_{W,b,S,F}\sum_{i,j=1}^{n}\|W^Tx_i-W^Tx_j\|_2^2s_{ij}+\gamma\|S\|_F^2+\beta\|W\|_F^2\\&+\alpha Tr(W^TX+b\mathbf{1}^T-Y)U(W^TX+b\mathbf{1}^T-Y)^T\\&+2\lambda Tr(FLF^T),\quad s.t.\ S\geq 0, S^T\mathbf{1}=\mathbf{1}, FF^T=I_c\end{aligned}\tag{7}$$

3.2 Optimization

We divide the problem in Eq.(7) into three subproblems, and propose an alternative and iterative algorithm to optimize them. The whole procedure is summarized in Algorithm 1.

Step 1: Update F with W, b and S fixed. The problem in Eq.(7) becomes:

$$\min_{FF^T=I_c} Tr(FLF^T).\tag{8}$$

The optimal solution F is formed by eigenvectors of L corresponding to the first c smallest eigenvalues.

Step 2: Update W, b with F and S fixed. The problem in Eq.(7) becomes:

$$\begin{aligned}&\min_{W,b}\ \alpha Tr(W^TX+b\mathbf{1}^T-Y)U(W^TX+b\mathbf{1}^T-Y)^T\\&+\sum_{i,j=1}^{n}\|W^Tx_i-W^Tx_j\|_2^2s_{ij}+\beta\|W\|_F^2.\end{aligned}\tag{9}$$

To obtain the optimal solution, by setting the derivatives of the objective function with respect to W and b equal to zero, respectively, we have:

$$\begin{aligned}W&=\alpha(2XLX^T+\alpha XH_{cu}X^T+\beta I_d)^{-1}XH_{cu}Y^T,\\b&=\frac{1}{l}(Y-W^TX)U\mathbf{1},\end{aligned}\tag{10}$$

where $H_{cu}=U-\frac{1}{l}U\mathbf{1}\mathbf{1}^TU$ is the centering matrix for the labeled data.

Step 3: Update S with W, b and F fixed. The problem in Eq.(7) becomes:

$$\begin{aligned}&\min_{S}\sum_{i,j=1}^{n}\|W^Tx_i-W^Tx_j\|_2^2s_{ij}+\gamma\|S\|_F^2\\&+2\lambda Tr(FLF^T),\quad s.t.\ S\geq 0, S^T\mathbf{1}=\mathbf{1}\end{aligned}\tag{11}$$

We have $2Tr(FLF^T)=\sum_{i,j=1}^{n}\|f_i-f_j\|_2^2s_{ij}$ which is similar to Eq.(3). Note that it is independent to conduct adaptive neighbor learning for each data point. Thus we can solve the following problem for the i-th sample:

$$\begin{aligned}&\min_{s_i}\sum_{j=1}^{n}(\|W^Tx_i-W^Tx_j\|_2^2s_{ij}+\gamma s_{ij}^2)\\&+\lambda\sum_{j=1}^{n}\|f_i-f_j\|_2^2s_{ij},\quad s.t.\ s_i\geq 0, s_i^T\mathbf{1}=1\end{aligned}\tag{12}$$

Algorithm 1 The Proposed Method LMRAG

Input: Data matrix $X\in\mathbb{R}^{d\times n}$, where $\{x_i\}_{i=1}^{l}$ are labeled and $\{x_i\}_{i=l+1}^{n}$ are unlabeled, label matrix $Y\in\mathbb{R}^{c\times n}$, trade-off parameters α,β, and the neighbor number k.

1: Initialize S, λ, γ according to the initialization section.
2: **while** not converge **do**
3: Update F, which is formed by eigenvectors of L corresponding to the first c smallest eigenvalues.
4: Update W,b by Eq.(10).
5: Update S by solving Eq.(13) for each sample.
6: **end while**

Output: Projection matrix W.

Denote $d_{ij}=\|W^Tx_i-W^Tx_j\|_2^2+\lambda\|f_i-f_j\|_2^2$, and denote $d_i\in\mathbb{R}^{n\times 1}$ as a constant vector with the j-th entry as d_{ij}, Eq.(12) can be rewritten as follows:

$$\min_{s_i\geq 0, s_i^T\mathbf{1}=1}\|s_i-(-\frac{1}{2\gamma})d_i\|_2^2.\tag{13}$$

The problem in Eq.(13) naturally has a sparse solution and can be solved by an efficient iterative algorithm [Huang *et al.*, 2015]. We can also just update the k nearest similarities for each sample to ensure a sparse solution.

3.3 Initialization

We can learn an initial graph by solving the problem of Eq.(2), and the algorithm proposed in [Huang *et al.*, 2015] can be adopted again. Alternatively, based on k-nearest neighbor (KNN) assumption, we apply another strategy to tackle the problem, and at the same time to determine the parameter γ. The Lagrangian function of Eq.(2) for the i-th sample can be written as follows:

$$L(s_i,\eta,\xi)=\frac{1}{2}\|s_i+\frac{1}{2\gamma_i}z_i\|_2^2-\eta(s_i^T\mathbf{1}-1)-\xi_i^Ts_i,\tag{14}$$

where $z_{ij}=\|x_i-x_j\|_2^2$, η and $\xi\in\mathbb{R}^{n\times 1}$ are the Lagrangian multipliers. $z_i\in\mathbb{R}^{n\times 1}$ is a constant vector with the j-th entry as z_{ij}, and the overall γ can be set to the average of $\{\gamma_i\}_{i=1}^{n}$. Based on the KKT condition, the optimal s_i has $s_{ij}=(-\frac{z_{ij}}{2\gamma_i}+\eta)_+$, where $(z)_+=\max(z,0)$.

We consider that each sample has k nearest neighbors, *i.e.*, s_i has k nonzero entries. Let us rank z_i in ascending order, we have

$$\begin{cases}s_{ik}=-\dfrac{z_{ik}}{2\gamma_i}+\eta>0\\ s_{i,k+1}=-\dfrac{z_{i,k+1}}{2\gamma_i}+\eta\leq 0\\ s_i^T\mathbf{1}=\displaystyle\sum_{j=1}^{k}(-\frac{z_{ij}}{2\gamma_i}+\eta)=1\end{cases}\Rightarrow\begin{cases}\gamma_i=\dfrac{k}{2}z_{i,k+1}-\dfrac{1}{2}\displaystyle\sum_{j=1}^{k}z_{ij}\\ \eta=\dfrac{1}{k}+\dfrac{1}{2k\gamma_i}\displaystyle\sum_{j=1}^{k}z_{ij}\end{cases}$$

In above derivations, we get a value range for γ_i and we set it to the maximum. Consequently, the initial S can be computed by

$$s_{ij}=\begin{cases}\frac{z_{i,k+1}-z_{ij}}{kz_{i,k+1}-\sum_{m=1}^{k}z_{im}}, & j\leq k\\ 0, & j>k\end{cases}\tag{15}$$

As for parameter λ, in practice, we can use a dynamic strategy to determine it and to accelerate the iterative process. Specifically, we can initialize $\lambda = \gamma$. Denote the number of connected components in S as ncc, then in each iteration, we double λ if $c > ncc$, halve λ if $c < ncc$, and we stop the iteration, otherwise.

3.4 Computational Complexity

The complexity of step 1 is $O(n^3)$. Considering that L is sparse, the ARPACK eigensolver [Lehoucq *et al.*, 1998] can be adopted to reduce the cost to $(O(p^3) + [O(np) + O(nk)] \times O(p - c)) \times T$, where p is a value several times larger than c, and T is the number of times of restarted Arnoldi.

Step 2 mainly takes $O(n^2d + nd^2 + d^3)$. We can use the Nyström method [Fowlkes *et al.*, 2004] to reduce the cost of the inverse operation performed on a symmetric matrix. Woodbury formula can also be used when $d > n$.

Comparing to step 1 and step 2, the complexity of step 3 can be ignored since the algorithm proposed in [Huang *et al.*, 2015] is based on Newton method that has quadratic convergence rate, and in practice we can just update the local similarities.

4 Discussions

There have been extensive study on the problem of dimensionality reduction. Besides the traditional KNN graph, there also exist many graph construction methods [Liu *et al.*, 2010; Zhang *et al.*, 2014]. However, most graph-based methods conduct graph construction and dimensionality reduction in two separate steps, and a very limited number of works have devoted to learning an optimized graph for dimensionality reduction.

Graph optimized locality preserving projection (GoLPP) [Zhang *et al.*, 2010] is the first attempt to perform graph optimization during a specific dimensionality reduction task according to the authors. The idea of GoLPP is to regularize the objective function of asymmetrical LPP [He *et al.*, 2005] by an entropy term. However, GoLPP suffers the nonuniqueness of the solutions, since GoLPP is formulated in the trace ratio form while solved in the ratio trace form [Wang *et al.*, 2007; Jia *et al.*, 2009]. Due to the entropy regularizer, the graph learned by GoLPP is dense even though a sparse initial graph is given.

To address the problems of GoLPP, graph optimization for dimensionality reduction with sparsity constraints (GODRSC) was then proposed [Zhang *et al.*, 2012], based on the orthogonalization of sparsity preserving projections (SPP) [Qiao *et al.*, 2010]. GODRSC obtains the sparsity of graph by replacing the entropy regularizer in GoLPP with an ℓ_1-norm minimization, and avoids nonunique solution by directly solving the trace ratio formulation.

GoLPP and GODRSC are both proposed for unsupervised dimensionality reduction. Therefore, LMRAG is of great value as an effective extension of the existing graph optimized dimensionality reduction methods in semi-supervised case. In fact, LMRAG can be easily extended to the unsupervised case, by removing the label fitness term in the formulation and adding an orthogonal constraint to the projection matrix to avoid trivial solution.

Table 1: Description of Datasets

dataset	Type	# samples	# Dim	# Classes
Corel	feature	2074	144	18
COIL-20	object	1440	1024	20
JAFFE	face	213	1024	10
CMU PIE	face	3332	1024	68
UMIST	face	575	2576	20
YALE-B	face	2414	1024	38
YALE	face	165	1024	15

Recently, [Meng *et al.*, 2015] proposed the adaptive semi-supervised dimensionality reduction (ASSDR), trying to optimize the graph by a heuristic iteration scheme. Two matrices of size $n \times n$ need to be stored in each iteration, which is quite memory consuming. ASSDR may rely on *kmeans* in the second step while *kmeans* itself is sensitive to initialization. Compared to ASSDR, LMRAG has several advantages: (1) LMRAG has a specific objective function, while ASSDR does not have one since it is based on a heuristic scheme. (2) LMRAG adaptively learns the graph, while ASSDR still uses the pre-defined way in each iteration. (3) The adaptive graph learned by LMRAG is sparse and structured, and the initial graph computed by Eq.(15) is also scale invariant.

5 Experiments

5.1 Datasets

We use several widely used benchmark datasets JAFFE[1] , CMU PIE [Sim *et al.*, 2003], UMIST[2], YALE, YALE-B[3], Corel [Chen *et al.*, 2011] and COIL-20[4] to evaluate the proposed LMRAG in our experiments. We provide a brief description of these datasets below.

JAFFE contains 213 images of 7 facial expressions posed by 10 Japanese female models. We used the frontal pose subset (C27) of CMU PIE, in which the images were acquired under variable illuminations and with different expressions. The images in UMIST cover a wide range of poses from profile to frontal views. YALE contains 15 individuals and each one has 11 grayscale images under variable illuminations. YALE-B is an extended version of YALE. Corel has 2074 images, which are represented by color, texture, and shape. COIL-20 is an object dataset that the images were captured from varying angles.

These datasets were first scaled to [0,1] by feature. We cropped UMIST to the size of 56×46. Except for Corel, PCA is then conducted on them with 98% information reserved. The detailed statistics can be seen in Table 1.

5.2 Comparison Algorithms

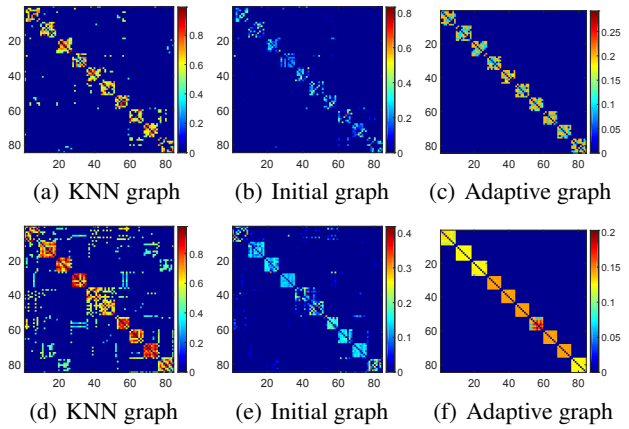
We compare LMRAG with five existing methods: semi-supervised discriminant analysis (SDA) [Cai *et al.*, 2007], trace ratio based flexible SDA (TR-FSDA) [Huang *et al.*, 2012], stable semi-supervised discriminant learning (SSDL)

[1]http://www.kasrl.org/jaffe.html

[2]http://www.cs.nyu.edu/ roweis/data.html

[3]http://www.cad.zju.edu.cn/home/dengcai/Data/data.html

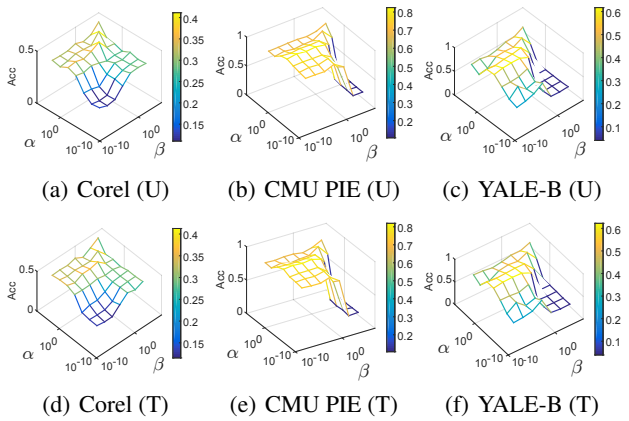
[4]http://www.cs.columbia.edu/CAVE/software/softlib/coil-20.php



(a) KNN graph (b) Initial graph (c) Adaptive graph

(d) KNN graph (e) Initial graph (f) Adaptive graph

Figure 1: Illustrations of the KNN graph, the initial graph and the adaptive graph on JAFFE. The neighbor number k is 5 in (a)(b)(c) and increases to 10 in (d)(e)(f). The data points are reorganized such that the samples with the same label are placed continuously.



(a) Corel (U) (b) CMU PIE (U) (c) YALE-B (U)

(d) Corel (T) (e) CMU PIE (T) (f) YALE-B (T)

Figure 2: The effect of parameters α and β to accuracy (Acc). U denotes the unlabeled training data and T means the test data.

[Gao *et al.*, 2015], FME [Nie *et al.*, 2010] and LapRLS/L [Sindhwani *et al.*, 2005b].

SDA is a representative method that imposes the graph Laplacian regularization into the objective function of linear discriminant analysis (LDA) [Belhumeur *et al.*, 1997]. Based on SDA and FME, TR-FSDA was proposed as the first semi-supervised dimensionality reduction method using trace ratio criterion. SSDL considers both the similarity and diversity of data to design the graph, which is then incorporated into the objective function of LDA. We also evaluate the projection ability of LapRLS/L to verify the effectiveness of incorporating LapRLS/L and the adaptive neighbor learning.

Since ASSDR [Meng *et al.*, 2015] is based on pairwise constraints rather than directly uses the label information, to be fair, we do not consider it as a comparison method.

5.3 Experimental Setting

The parameters α and β in LMRAG, SDA, TR-FSDA and SSDL[5], μ and γ in FME, γ_A and γ_I in LapRLS/L need to be tuned, respectively. We searched their values in the range of $\{10^{-6}, 10^{-4}, 10^{-2}, 10^{0}, 10^{2}, 10^{4}, 10^{6}\}$. For fair comparison, the reduced dimensionality was fixed as c in SDA, TR-FSDA and SSDL. We randomly chose 40% samples per class as the training data, and used the remaining 60% as the test data. Among the training data, we randomly selected $p = \{1, 2, 3\}$ samples per class as the labeled data, and used the remaining as the unlabeled data.

To evaluate the projection ability, the nearest neighbor classifier was performed on the projected data for final classification. We uniformly set the neighbor number k to 5 and chose the band width σ of Gaussian kernel in a self-tuning way [Chen *et al.*, 2011] while evaluating the classification performance. We report the best mean accuracy and standard deviation (std) over 20 random splits on each dataset.

[5]We applied the Tikhonov regularization to handle the singular problem, thus an additional parameter β is introduced to SSDL.

5.4 Experimental Results

In Figure 1, we tested on JAFFE to give intuitive and practical illustrations of the initial graphs and the adaptive graphs learned by LMRAG with the different neighbor number k. For comparison, the traditional KNN graphs were also illustrated. As can be seen, there are many strong inter-class connections in the KNN graph when k is just a small value 5, and the situation becomes much worse as k increases to 10. We see that the initial graphs of LMRAG are sparse and the sparsity marginally changes with k increasing. With the good initialization and structure constraint, the final adaptive graphs of LMRAG are indeed sparse and structured.

Since parameter γ can be initialized adaptively, and in practice λ can be tuned by a dynamic strategy, we only studied the effect of parameters α and β to the final classification performance on three datasets Corel, CMU PIE and YALE-B. The parameter p was set to 3 during the tests. Figure 2 displays the 3D mesh plots, from which we have the following observations:

1. On each dataset, the performance on the unlabeled training data is basically consistent with the performance on the test data.
2. Comparing to the performance on Corel and YALE-B, the performance on CMU PIE is pretty robust to parameters α and β in a wide range, perhaps because there are more training data in CMU PIE to make up for the accuracy loss caused by inappropriate parameter settting.
3. As parameter α goes up in a range, which means the label fitness term plays a more and more important role in Eq.(7), the performance on all datasets tends to become better. This point is consistent with the first observation from Table 2 listed below. The accuracy may drop when α gets too large, since the model can not make the best use of the unlabeld training data.

Table 2 shows the classification performance in the projected space. Several observations can be made as follows:

1. As the number of labeled samples goes up, the performance of all the methods tends to be better, which demonstrates the usefulness of the labeled data.

Table 2: Performance Comparison (% $\pm$ std)

Dataset	Method	1 labeled sample		2 labeled sample		3 labeled sample	
		Unlabeled	Test	Unlabeled	Test	Unlabeled	Test
Corel	SDA	25.44 ± 3.42	25.42 ± 2.81	34.86 ± 3.67	34.58 ± 2.71	39.61 ± 4.68	38.39 ± 1.43
	TR-FSDA	25.44 ± 4.46	25.70 ± 2.97	33.83 ± 3.16	33.46 ± 3.27	38.07 ± 4.00	38.55 ± 3.56
	SSDL	26.75 ± 1.93	27.20 ± 2.84	34.89 ± 2.55	34.08 ± 2.26	38.32 ± 1.79	37.62 ± 2.74
	FME	23.65 ± 3.13	24.44 ± 2.56	30.15 ± 3.16	31.83 ± 4.46	33.79 ± 1.06	32.60 ± 1.26
	LapRLS/L	26.90 ± 1.60	26.60 ± 3.65	33.83 ± 2.34	34.31 ± 0.55	40.49 ± 1.42	39.94 ± 3.52
	LMRAG	$\mathbf{27.86 \pm 3.49}$	$\mathbf{27.73 \pm 3.00}$	$\mathbf{36.12 \pm 2.56}$	$\mathbf{36.46 \pm 1.06}$	$\mathbf{41.13 \pm 1.28}$	$\mathbf{41.27 \pm 2.08}$
COIL-20	SDA	69.75 ± 3.46	68.42 ± 2.35	77.85 ± 2.87	76.70 ± 3.29	82.23 ± 2.50	81.79 ± 2.74
	TR-FSDA	69.54 ± 2.36	68.53 ± 2.01	76.52 ± 3.11	76.69 ± 3.62	83.00 ± 1.18	82.77 ± 3.23
	SSDL	65.29 ± 2.54	64.56 ± 2.53	75.11 ± 2.04	75.56 ± 2.08	79.69 ± 3.37	80.12 ± 1.53
	FME	69.36 ± 4.12	69.35 ± 2.28	78.48 ± 1.95	76.98 ± 2.38	84.38 ± 1.74	84.35 ± 2.70
	LapRLS/L	68.68 ± 4.29	66.38 ± 1.69	75.85 ± 1.29	75.98 ± 2.67	79.69 ± 1.33	79.30 ± 1.61
	LMRAG	$\mathbf{70.75 \pm 1.80}$	$\mathbf{70.12 \pm 2.57}$	$\mathbf{79.70 \pm 2.73}$	$\mathbf{77.84 \pm 3.09}$	$\mathbf{84.58 \pm 1.86}$	$\mathbf{85.00 \pm 1.45}$
JAFFE	SDA	91.62 ± 1.76	88.06 ± 4.05	95.94 ± 2.84	97.36 ± 1.41	98.52 ± 1.55	99.22 ± 1.88
	TR-FSDA	87.84 ± 3.02	86.05 ± 7.25	96.88 ± 3.66	96.28 ± 2.76	98.15 ± 3.21	99.22 ± 1.34
	SSDL	83.24 ± 3.65	84.65 ± 3.97	94.69 ± 2.61	94.26 ± 2.72	98.89 ± 1.01	98.29 ± 1.77
	FME	80.27 ± 8.23	82.48 ± 5.20	92.19 ± 5.18	90.23 ± 3.54	94.81 ± 4.22	94.57 ± 1.98
	LapRLS/L	86.49 ± 6.12	86.51 ± 4.98	95.63 ± 5.11	94.57 ± 3.84	$\mathbf{99.26 \pm 2.69}$	98.29 ± 1.01
	LMRAG	$\mathbf{97.84 \pm 2.80}$	$\mathbf{98.45 \pm 1.55}$	$\mathbf{99.38 \pm 2.09}$	$\mathbf{98.45 \pm 1.45}$	$\mathbf{99.26 \pm 1.66}$	$\mathbf{99.69 \pm 0.43}$
CMU PIE	SDA	31.53 ± 3.71	32.29 ± 1.68	68.40 ± 2.31	68.38 ± 1.78	77.47 ± 2.32	77.57 ± 2.63
	TR-FSDA	18.98 ± 0.92	22.56 ± 1.37	67.55 ± 2.85	67.51 ± 1.35	79.27 ± 1.79	78.13 ± 1.30
	SSDL	53.78 ± 1.98	53.17 ± 2.35	70.28 ± 2.83	70.69 ± 2.10	77.49 ± 1.14	78.14 ± 1.11
	FME	53.49 ± 1.47	52.26 ± 1.24	69.92 ± 2.17	69.06 ± 1.36	78.06 ± 2.39	77.19 ± 1.92
	LapRLS/L	53.31 ± 2.19	52.80 ± 2.68	69.15 ± 2.09	68.63 ± 1.77	77.35 ± 2.33	76.57 ± 2.20
	LMRAG	$\mathbf{61.30 \pm 2.29}$	$\mathbf{61.29 \pm 1.08}$	$\mathbf{72.61 \pm 2.50}$	$\mathbf{72.61 \pm 2.71}$	$\mathbf{82.42 \pm 1.06}$	$\mathbf{81.93 \pm 1.15}$
UMIST	SDA	50.67 ± 5.17	47.54 ± 2.80	77.68 ± 4.20	77.10 ± 4.01	85.06 ± 5.01	86.03 ± 2.44
	TR-FSDA	47.81 ± 4.50	50.78 ± 6.00	77.37 ± 6.15	76.93 ± 4.68	$\mathbf{87.47 \pm 4.42}$	87.62 ± 3.36
	SSDL	48.57 ± 2.65	48.93 ± 2.55	79.16 ± 3.85	78.03 ± 3.96	84.71 ± 4.01	84.93 ± 1.93
	FME	48.57 ± 3.70	48.00 ± 3.61	76.00 ± 4.69	74.06 ± 4.96	83.41 ± 5.42	82.42 ± 2.76
	LapRLS/L	48.29 ± 4.44	46.32 ± 4.68	66.21 ± 1.55	67.88 ± 4.21	78.71 ± 3.56	75.54 ± 4.28
	LMRAG	$\mathbf{58.38 \pm 3.39}$	$\mathbf{57.33 \pm 3.86}$	$\mathbf{80.84 \pm 2.70}$	$\mathbf{79.30 \pm 2.29}$	86.47 ± 5.03	$\mathbf{88.87 \pm 3.37}$
YALE-B	SDA	21.47 ± 2.24	22.87 ± 2.49	49.51 ± 1.32	48.36 ± 2.01	58.31 ± 2.40	59.35 ± 1.94
	TR-FSDA	13.70 ± 2.03	16.27 ± 2.43	45.52 ± 2.31	45.49 ± 2.49	57.29 ± 1.77	58.09 ± 2.26
	SSDL	29.81 ± 0.98	29.87 ± 1.72	47.07 ± 2.43	47.47 ± 2.82	59.54 ± 1.14	57.66 ± 2.36
	FME	31.23 ± 1.62	32.87 ± 1.74	49.48 ± 1.37	49.79 ± 3.23	58.65 ± 1.11	59.77 ± 2.19
	LapRLS/L	32.85 ± 2.79	33.97 ± 2.09	49.07 ± 3.54	49.05 ± 3.55	60.02 ± 3.86	58.73 ± 3.51
	LMRAG	$\mathbf{46.17 \pm 3.65}$	$\mathbf{44.86 \pm 3.45}$	$\mathbf{57.41 \pm 3.13}$	$\mathbf{57.14 \pm 3.16}$	$\mathbf{61.76 \pm 1.89}$	$\mathbf{62.84 \pm 2.24}$
YALE	SDA	43.11 ± 7.47	40.38 ± 4.24	54.00 ± 5.48	56.95 ± 2.17	68.00 ± 7.30	68.76 ± 2.97
	TR-FSDA	43.56 ± 7.37	40.95 ± 5.95	60.67 ± 6.41	58.29 ± 5.28	66.67 ± 8.16	67.62 ± 4.40
	SSDL	42.22 ± 7.03	41.14 ± 6.58	56.00 ± 7.01	57.71 ± 5.83	72.00 ± 9.05	68.00 ± 5.35
	FME	40.89 ± 5.31	37.71 ± 4.88	49.33 ± 7.23	53.71 ± 5.11	62.67 ± 5.58	58.67 ± 3.73
	LapRLS/L	38.67 ± 4.33	39.05 ± 4.86	58.67 ± 7.67	56.95 ± 4.54	74.67 ± 7.60	64.57 ± 5.97
	LMRAG	$\mathbf{45.78 \pm 5.75}$	$\mathbf{44.95 \pm 4.01}$	$\mathbf{66.67 \pm 4.35}$	$\mathbf{61.33 \pm 4.38}$	$\mathbf{76.00 \pm 5.58}$	$\mathbf{69.90 \pm 3.96}$

2. With respect to the mean accuracy, LMRAG outperforms the other five methods in 40 out of 42 cases, showing the effectiveness of learning an adaptive graph.

3. On four face datasets, when p equals to 1, the performance of LMRAG has a great improvement over others. Specifically, for unlabeled training data, LMRAG exceeds the second best results 6.22%, 7.52%, 7.71% and 13.32% on JAFFE, CMU PIE, UMIST and YALE-B, respectively. For test data, LMRAG exceeds 10.39%, 8.12%, 6.55% and 10.89%, respectively.

4. The superior of LMRAG over LapRLS/L demonstrates the effectiveness of incorporating the adaptive neighbor learning into the objective function of LapRLS/L.

6 Conclusion

In this paper, we have proposed a novel approach denoted by LMRAG which incorporates the graph construction into the semi-supervised dimensionality reduction. The projection matrix and the optimal graph for the specific task are then both obtained. The proposed LMRAG is meaningful as an effective extension and supplement of the existing graph optimized dimensionality reduction methods. Extensive experiments have demonstrated the superiority of LMRAG, comparing to other state-of-the-art methods.

Acknowledgments

This work was supported in part by the National Science Foundation of China under Grants 61522207 and 61473231.

References

[Belhumeur *et al.*, 1997] Peter N. Belhumeur, João P Hespanha, and David J. Kriegman. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *TPAMI*, 19(7):711–720, 1997.

[Belkin and Niyogi, 2001] Mikhail Belkin and Partha Niyogi. Laplacian eigenmaps and spectral techniques for embedding and clustering. In *NIPS*, volume 14, pages 585–591, 2001.

[Belkin *et al.*, 2006] Mikhail Belkin, Partha Niyogi, and Vikas Sindhwani. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *JMLR*, 7(Nov):2399–2434, 2006.

[Cai *et al.*, 2007] Deng Cai, Xiaofei He, and Jiawei Han. Semi-supervised discriminant analysis. In *ICCV*, pages 1–7. IEEE, 2007.

[Chatpatanasiri and Kijsirikul, 2010] Ratthachat Chatpatanasiri and Boonserm Kijsirikul. A unified semi-supervised dimensionality reduction framework for manifold learning. *Neurocomputing*, 73(10):1631–1640, 2010.

[Chen *et al.*, 2011] Wen-Yen Chen, Yangqiu Song, Hongjie Bai, Chih-Jen Lin, and Edward Y Chang. Parallel spectral clustering in distributed systems. *TPAMI*, 33(3):568–586, 2011.

[Fowlkes *et al.*, 2004] Charless Fowlkes, Serge Belongie, Fan Chung, and Jitendra Malik. Spectral grouping using the nystrom method. *TPAMI*, 26(2):214–225, 2004.

[Gao *et al.*, 2015] Quanxue Gao, Yunfang Huang, Xinbo Gao, Weiguo Shen, and Hailin Zhang. A novel semi-supervised learning for face recognition. *Neurocomputing*, 152:69–76, 2015.

[He *et al.*, 2005] Xiaofei He, Shuicheng Yan, Yuxiao Hu, Partha Niyogi, and Hong-Jiang Zhang. Face recognition using laplacianfaces. *TPAMI*, 27(3):328–340, 2005.

[He *et al.*, 2008] Xiaofei He, Deng Cai, and Jiawei Han. Learning a maximum margin subspace for image retrieval. *TKDE*, 20(2):189–201, 2008.

[Huang *et al.*, 2012] Yi Huang, Dong Xu, and Feiping Nie. Semi-supervised dimension reduction using trace ratio criterion. *TNNLS*, 23(3):519–526, 2012.

[Huang *et al.*, 2015] Jin Huang, Feiping Nie, and Heng Huang. A new simplex sparse learning model to measure data similarity for clustering. In *IJCAI*, pages 3569–3575. AAAI Press, 2015.

[Jia *et al.*, 2009] Yangqing Jia, Feiping Nie, and Changshui Zhang. Trace ratio problem revisited. *TNN*, 20(4):729–735, 2009.

[Lehoucq *et al.*, 1998] Richard B Lehoucq, Danny C Sorensen, and Chao Yang. *ARPACK users' guide: solution of large-scale eigenvalue problems with implicitly restarted Arnoldi methods*, volume 6. SIAM, 1998.

[Liu *et al.*, 2010] Wei Liu, Junfeng He, and Shih-Fu Chang. Large graph construction for scalable semi-supervised learning. In *ICML*, pages 679–686, 2010.

[Meng *et al.*, 2015] Meng Meng, Jia Wei, Jiabing Wang, Qianli Ma, and Xuan Wang. Adaptive semi-supervised dimensionality reduction based on pairwise constraints weighting and graph optimizing. *IJMLC*, pages 1–13, 2015.

[Mohar *et al.*, 1991] Bojan Mohar, Y Alavi, G Chartrand, and OR Oellermann. The laplacian spectrum of graphs. *Graph theory, combinatorics, and applications*, 2(871-898):12, 1991.

[Nie *et al.*, 2009] Feiping Nie, Shiming Xiang, Yangqing Jia, and Changshui Zhang. Semi-supervised orthogonal discriminant analysis via label propagation. *PR*, 42(11):2615–2627, 2009.

[Nie *et al.*, 2010] Feiping Nie, Dong Xu, Ivor Wai-Hung Tsang, and Changshui Zhang. Flexible manifold embedding: A framework for semi-supervised and unsupervised dimension reduction. *TIP*, 19(7):1921–1932, 2010.

[Nie *et al.*, 2014] Feiping Nie, Xiaoqian Wang, and Heng Huang. Clustering and projected clustering with adaptive neighbors. In *SIGKDD*, pages 977–986. ACM, 2014.

[Qiao *et al.*, 2010] Lishan Qiao, Songcan Chen, and Xiaoyang Tan. Sparsity preserving projections with applications to face recognition. *PR*, 43(1):331–341, 2010.

[Roweis and Saul, 2000] Sam T Roweis and Lawrence K Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326, 2000.

[Sim *et al.*, 2003] T Sim, S Baker, and M Bsat. The cmu pose, illumination, and expression database. *TPAMI*, 25(12):1615–1618, 2003.

[Sindhwani *et al.*, 2005a] Vikas Sindhwani, Partha Niyogi, and Mikhail Belkin. Beyond the point cloud: from transductive to semi-supervised learning. In *ICML*, pages 824–831. ACM, 2005.

[Sindhwani *et al.*, 2005b] Vikas Sindhwani, Partha Niyogi, Mikhail Belkin, and Sathiya Keerthi. Linear manifold regularization for large scale semi-supervised learning. In *ICML Workshop on Learning with Partially Classified Training Data*, volume 28, 2005.

[Song *et al.*, 2008] Yangqiu Song, Feiping Nie, Changshui Zhang, and Shiming Xiang. A unified framework for semi-supervised dimensionality reduction. *PR*, 41(9):2789–2799, 2008.

[Von Luxburg, 2007] Ulrike Von Luxburg. A tutorial on spectral clustering. *Statistics and computing*, 17(4):395–416, 2007.

[Wang *et al.*, 2007] Huan Wang, Shuicheng Yan, Dong Xu, Xiaoou Tang, and Thomas Huang. Trace ratio vs. ratio trace for dimensionality reduction. In *CVPR*, pages 1–8. IEEE, 2007.

[Yan *et al.*, 2007] Shuicheng Yan, Dong Xu, Benyu Zhang, Hong-Jiang Zhang, Qiang Yang, and Stephen Lin. Graph embedding and extensions: a general framework for dimensionality reduction. *TPAMI*, 29(1):40–51, 2007.

[Zhang *et al.*, 2010] Limei Zhang, Lishan Qiao, and Songcan Chen. Graph-optimized locality preserving projections. *PR*, 43(6):1993–2002, 2010.

[Zhang *et al.*, 2012] Limei Zhang, Songcan Chen, and Lishan Qiao. Graph optimization for dimensionality reduction with sparsity constraints. *PR*, 45(3):1205–1210, 2012.

[Zhang *et al.*, 2014] Yan-Ming Zhang, Kaizhu Huang, Xinwen Hou, and Cheng-Lin Liu. Learning locality preserving graph from data. *IEEE Transactions on Cybernetics*, 44(11):2088–2098, 2014.