Flexible Shift-Invariant Locality and Globality Preserving Projections

Feiping Nie, Xiao Cai, and Heng Huang*

Department of Computer Science and Engineering, University of Texas at Arlington,
Arlington, 76019, Texas, USA
feipingnie@gmail.com, xiao.cai@mavs.uta.edu, heng@uta.edu

Abstract. In data mining and machine learning, the embedding methods have commonly been used as a principled way to understand the high-dimensional data. To solve the out-of-sample problem, local preserving projection (LPP) was proposed and applied to many applications. However, LPP suffers two crucial deficiencies: 1) the LPP has no shift-invariant property which is an important property of embedding methods; 2) the rigid linear embedding is used as constraint, which often inhibits the optimal manifold structures finding. To overcome these two important problems, we propose a novel flexible shift-invariant locality and globality preserving projection method, which utilizes a newly defined graph Laplacian and the relaxed embedding constraint. The proposed objective is very challenging to solve, hence we derive a new optimization algorithm with rigorously proved global convergence. More importantly, we prove our optimization algorithm is a Newton method with fast quadratic convergence rate. Extensive experiments have been performed on six benchmark data sets. In all empirical results, our method shows promising results.

1 Introduction

In many data mining applications, it is highly desirable to map high-dimensional input data to a lower dimensional space, with a constraint that the data from similar classes will be projected to nearby locations in the new space. Thus, many data embedding methods have been developed. Depending on whether the label information is used, these methods can be classified into two categories, *i.e.*, unsupervised and supervised. A representative of unsupervised embedding methods is PCA [11], which aims at identifying a lower-dimensional space maximizing the variance among data. A representative of supervised embedding methods is LDA [4], which aims at identifying a lower dimensional space minimizing the inter-class similarity while maximizing the intra-class similarity simultaneously.

To discover the intrinsic manifold structure of the data, multiple nonlinear embedding algorithms have been recently proposed to use an eigen-decomposition for obtaining a lower-dimensional embedding of data lying on a non-linear manifold, such as Isomap [22], LLE [19], Laplacian Eigenmap [2], Local Tangent Space Alignment

* This work was partially supported by NSF IIS-1117965, IIS-1302675, IIS-1344152, DBI-1356628.

T. Calders et al. (Eds.): ECML PKDD 2014, Part II, LNCS 8725, pp. 485–500, 2014.

(LTSA) [25] and Local Spline Embedding (LSE) [24]. However, many of them, such as Isomap and Laplacian Eigenmap, suffer from the out-of-sample problem, *i.e.* how to embed new points in relation to a previously specified configuration of fixed points. To deal with this problem, He *et al.* [7] developed the Locality Preserving Projections (LPP) method, in which the linear embedding function is used for mapping new data.Nie *et al.* [18] proposed a flexible linearization technique in which LPP and spectral regression [3] are two extreme cases.

Although LPP solved the out-of-sample problem, two crucial deficiencies exist in current LPP based methods. First, LPP has no shift-invariant property which is a basic property of subspace learning methods. The learned subspace (or the projection matrix) should be invariant when all training data points are shifted by the same constant vector. Second, in LPP, the rigid linear embedding is used as constraint, which often limits the search of optimal manifold structures.

To resolve these two important problems, we propose a novel flexible shift invariant locality and globality preserving projection (FLGPP) method. We reformulate the LPP objective using a correct Laplacian matrix which makes the new method shift invariant. Meanwhile, we show that the graph embedding methods are indeed locality and globality preserving projection methods, which were only considered as keeping the local geometrical structure. We relax the rigid linear embedding by allowing the error tolerance such that the data instances can be flexibly embedded. The proposed objective is very difficult to solve. As one important contribution of this paper, we derive a new optimization algorithm with proved global convergence. More importantly, we rigorously prove that our new optimization algorithm is a Newton method with fast quadratic convergence rate. To evaluate our method, we compare the new method to the LDA and LPP methods by performing them on six benchmark data sets. In all empirical results, our new FLGPP method shows promising results.

2 Locality Preserving Projections Revisit

2.1 Review of Related Graph Based Methods

Given n training data points $X = [x_1, \cdots, x_n] \in \mathbb{R}^{d\times n}$, where d is the data dimensionality and n is the number of data points, the graph based methods first construct a graph based on the data to encode the pairwise data similarities. With the graph affinity matrix $A \in \mathbb{R}^{n\times n}$, the Laplacian matrix is defined as $L = D - A$, where D is the diagonal matrix with the i-th diagonal element $D_{ii} = \sum_i A_{ij}$. L is positive semi-definite, and satisfies $L\mathbf{1} = \mathbf{0}$, where $\mathbf{1}$ is a vector having all elements as 1s, and $\mathbf{0}$ is a vector having all elements as 0s.

Traditional spectral clustering (or graph cut) [21,15] and Laplacian embedding (or graph embedding, manifold learning) [2] is to solve the following problem:

$$\min_{F^T QF=I} Tr(F^T LF), \tag{1}$$

where Q would be D or the n by n identity matrix I. The optimal solution $F \in \mathbb{R}^{n\times m} (m < n)$ to Eq. (1) is the eigenvectors of $Q^{-1}L$ corresponding to the smallest eigenvalues.

The methods solving problem Eq. (1) only use the given training data, with no straightforward extension for out-of-sample examples. To handle the out-of-sample problem, a seminal work called Locality Preserving Projections (LPP) was proposed [7], which is to solve the following problem:

$$\min_{\substack{F^T QF=I \\ X^T W=F}} Tr(F^T LF), \tag{2}$$

where $W \in \mathbb{R}^{d\times m}(m < d)$ is the projection matrix. This linearization method imposes a rigid constraint $X^T W = F$ on the problem Eq. (1), such that the data outside the training data can also be handled using the projection W. In LPP, only $Q = D$ is considered, thus the problem Eq. (2) can be written as:

$$\min_{W^T XDX^T W=I} Tr(W^T XLX^T W). \tag{3}$$

The optimal solution W to LPP is the eigenvectors of $(XDX^T)^{-1}XLX^T$ corresponding to the smallest eigenvalues. Many algorithms following this linearization method are also proposed for subspace learning and classifications in recent years.

2.2 Shift-Invariant Property

For subspace learning algorithms, shift invariance is a basic and important property. That is to say, the learned subspace (or the projection matrix W) should be invariant when every training data point x_i is shifted by the same constant vector c_i, *i.e.*, X is shifted to $X + c\mathbf{1}^T$. For example, PCA, LDA and regularized least squares regression are all shift-invariant algorithms. We are going to demonstrate this observation.

PCA solves:

$$\min_{W^T W=I} Tr(W^T XL_t X^T W), \tag{4}$$

where $L_t = I - \frac{1}{n}\mathbf{1}\mathbf{1}^T$ is the centering matrix, which is a Laplacian matrix and satisfies $L_t\mathbf{1} = \mathbf{0}$. As a result, we have $(X + c\mathbf{1}^T)L_t(X + c\mathbf{1}^T)^T = XL_tX^T$, and thus the optimal solution W will not be changed when the training data are shifted by an arbitrary vector c.

LDA is to solve:

$$\max_{W} Tr((W^T XL_w X^T W)^{-1} W^T XL_b X^T W), \tag{5}$$

where L_w and L_b are another two Laplacian matrices satisfying $L_w\mathbf{1} = \mathbf{0}$ and $L_b\mathbf{1} = \mathbf{0}$. Obviously we have $(X + c\mathbf{1}^T)L_w(X + c\mathbf{1}^T)^T = XL_wX^T$ and $(X + c\mathbf{1}^T)L_b(X + c\mathbf{1}^T)^T = XL_bX^T$, and thus the optimal solution W is also invariant to an arbitrary shift vector c.

Ridge regression solves:

$$\min_{W,b} \left\|X^T W + 1b^T - Y\right\|_F^2 + \gamma \left\|W\right\|_F^2, \tag{6}$$

which has a closed form solution $W = (XL_tX^T + \gamma I)^{-1}XL_tY$. Thus, the optimal solution W of the ridge regression is also invariant to arbitrary shift vector c.

One can immediately observe that the original LPP algorithm does not satisfy the shift-invariant property. When the data points are shifted by a same constant vector, although the distribution of the data points is not changed, the learned subspace by LPP will be changed. This problem should be avoided for a subspace learning algorithm.

3 Shift-Invariant Locality Preserving Projections

The original LPP was derived from Eq. (1), in which the optimal solution is the eigenvectors of $Q^{-1}L$ corresponding to the smallest eigenvalues. However, the smallest eigenvalue of $Q^{-1}L$ is 0 and the corresponding eigenvector is $\mathbf{1}$, which is usually discarded in practice. Thus the actual solutions are the eigenvectors of $Q^{-1}L$ corresponding the eigenvalues starting from the second smallest one, which is the solution to the following problem:

$$\min_{\substack{F^TQF=I\\ F^TQ\mathbf{1}=\mathbf{0}}} Tr(F^TLF). \tag{7}$$

Note that there is an additional constraint $F^TQ\mathbf{1}=\mathbf{0}$ in the problem. In the linearization method, when we use the linear constraint $X^TW = F$, the additional constraint $F^TQ\mathbf{1}=\mathbf{0}$ can not be satisfied since $W^TXQ\mathbf{1}\neq\mathbf{0}$. To fix this problem, we use the linear constraint with bias $X^TW+\mathbf{1}b^T = F$, where $b\in\mathbb{R}^{m\times 1}$ is the bias vector. With the additional constraint $F^TQ\mathbf{1}=\mathbf{0}$, we have $(W^TX+b\mathbf{1}^T)Q\mathbf{1}=\mathbf{0}\Rightarrow b=-\frac{1}{\mathbf{1}^TQ\mathbf{1}}W^TXQ\mathbf{1}$. Thus the linear constraint with bias in the linearization method is:

$$(I-\frac{1}{\mathbf{1}^TQ\mathbf{1}}\mathbf{1}\mathbf{1}^TQ)X^TW=F. \tag{8}$$

By imposing the linear constraint Eq.(8) to problem (1) or (7), the shift-invariant LPP is to solve the following problem [16]:

$$\min_{\substack{F^TQF=I\\ (I-\frac{1}{\mathbf{1}^TQ\mathbf{1}}\mathbf{1}\mathbf{1}^TQ)X^TW=F}} Tr(F^TLF)\,. \tag{9}$$

Define

$$L_q = Q-\frac{1}{\mathbf{1}^TQ\mathbf{1}}Q\mathbf{1}\mathbf{1}^TQ\,, \tag{10}$$

then the problem (14) can be re-written as

$$\min_{W^TXL_qX^TW=I} Tr(W^TXLX^TW). \tag{11}$$

Note that L and L_q are Laplacian matrix satisfying $L\mathbf{1}=\mathbf{0}$ and $L_q\mathbf{1}=\mathbf{0}$, so we have $(X+c\mathbf{1}^T)L(X+c\mathbf{1}^T)^T = XLX^T$ and $(X+c\mathbf{1}^T)L_q(X+c\mathbf{1}^T)^T = XL_qX^T$. Therefore, the optimal solution W of the problem (11) is invariant to arbitrary shift vector c.

From the above analysis we know that, the constraint $W^TXQX^TW=I$ (Q is a diagonal matrix such as D or I) will make the learned subspace does not satisfy the basic shift invariance property. The correct constraint should be $W^TXL_qX^TW = I$. There are many works following LPP used the constraint $W^TXDX^TW = I$, so this issue should be pointed out. Although this issue could be alleviated if we centralize the data such that the mean of the training data is zero, the users who are not aware of this issue may not always perform this preprocessing when they apply this kind of algorithms. Therefore, it is worth to emphasizing that the correct constraint $W^TXL_qX^TW = I$ instead of the $W^TXQX^TW = I$ should be used in subspace learning algorithm design.

4 Flexible Locality and Globality Preserving Embedding

4.1 Local and Global Viewpoints of The Graph Based Methods

It was known that the graph based data mining methods capture the local geometrical structure in training data. We will show that the graph based methods Laplacian embedding (solving Eq. (7)) and shift-invariant LPP (solving Eq. (11)) can capture both of local and global geometrical structure in training data.

Under the constraints in the problem (7), and according to Eq. (10), we know $Tr(F^TL_qF)$ is a constant. So problem (7) is equivalent to the following problem:

$$\min_{\substack{F^TQF=I \\ F^TQ1=0}} \frac{Tr(F^TLF)}{Tr(F^TL_qF)}. \tag{12}$$

Note that the following two equations hold:

$$\begin{aligned} Tr(F^TLF) &= \sum_{i=1}^{n}\sum_{j=1}^{n} A_{ij}\left\|f_i - f_j\right\|^2, \\ Tr(F^TL_qF) &= \sum_{i=1}^{n} Q_{ii}\left\|f_i - \bar{f}\right\|^2, \end{aligned} \tag{13}$$

where $\bar{f} = \sum\limits_{i=1}^{n} Q_{ii}f_i / \sum\limits_{i=1}^{n} Q_{ii}$ is the weighted mean of $f_i|_1^n$. When $Q = I, Tr(F^TL_qF)$ is the variance of the n embedded data points $f_i|_1^n$. When $Q = D$, $Tr(F^TL_qF)$ is the weighed variance of the n embedded data points $f_i|_1^n$ with the weight D_{ii} for the i-th embedded data point f_i.

Thus, from Eq. (13), we can conclude that solving the problem (12) is to minimize the Euclidean distances between local data pairs in the embedded space and also to maximize the (weighted) variance of the total data points in the embedded space at the same time, which provides us a new understanding on the Laplacian embedding methods.

Similarly, problem (11) is equivalent to the following problem

$$\min_{W^TXL_qX^TW=I} \frac{Tr(W^TXLX^TW)}{Tr(W^TXL_qX^TW)}. \tag{14}$$

Thus, solving the problem (11) is to minimize the Euclidean distances between local data pairs in the projected subspace and also to maximize the (weighted) variance of the total data points in the projected subspace at the same time. That is to say, although the algorithm LPP is called "locality preserving", it can preserve both of the locality and globality structure in the training data.

If we use the orthogonal constraint instead of the constraint $W^T X L_q X^T W = I$, the problem (14) becomes the trace ratio LPP problem [16], which can be efficiently solved by an iterative algorithm with quadratic convergence rate [23,10]:

$$\min_{W^T W = I} \frac{Tr(W^T X L X^T W)}{Tr(W^T X L_q X^T W)}. \tag{15}$$

4.2 Locality and Globality Preserving Projections with Flexible Constraint

Traditional linearization method imposes a constraint $X^T W = F$ to learn the projection matrix W. Because F in the original problems (*e.g.* Eq. (1)) usually is nonlinear, imposing the constraint that F must be exactly equal to the linear model $X^T W$ is too rigid in practice. In this paper, we propose to use a flexible constraint $\left\|X^T W - F\right\|_F^2 \leq \delta$ instead of the rigid constraint $X^T W = F$ in the linearization method. With this flexible linearization constraint and motivations inspired by Eq. (12), we propose the Flexible Locality and Globality Preserving Projections (FLGPP), which is to solve :

$$\min_{\substack{F, W^T W = I \\ ||X^T W - F||_F^2 \leq \delta}} \frac{Tr(F^T L F)}{Tr(F^T L_q F)}. \tag{16}$$

The problem (16) is equivalent to

$$\min_{F, W^T W = I} \frac{Tr(F^T L F)}{Tr(F^T L_q F)} + \lambda \left\|X^T W - F\right\|_F^2, \tag{17}$$

where $\lambda > 0$ is the Lagrangian multiplier coefficient. We propose to solve a similar problem to Eq. (17) for the FLGPP as follows:

$$\min_{F, W^T W = I} \frac{Tr(F^T L F) + \gamma \left\|X^T W - F\right\|_F^2}{Tr(F^T L_q F)}. \tag{18}$$

This new objective is very difficult to optimize, because there are two variables W and F to be solved. Moreover, the non-convex objective function is a ratio of two terms, meanwhile there is a non-convex constraint in the problem, which makes the optimization procedure more challenging. In next section, as one important contribution of this paper, we will propose an effective algorithm to solve the proposed objective, and also prove the algorithm converges to the global optimal solution with quadratic convergence rate, even though the problem is not convex.

5 New Optimization Algorithm

5.1 Proposed Algorithm

Denote $N = (L - \lambda L_q + \gamma I)^{-1}$ and define a function $g(\lambda)$ as follows:

$$g(\lambda) = \min_{F, W^T W = I} Tr F^T (L - \lambda L_q) F + \gamma \left\| X^T W - F \right\|_F^2 \tag{19}$$

Eq. (19) can be written as:

$$\begin{aligned} g(\lambda) = &\min_{F, W^T W = I} Tr(F^T N^{-1} F) \\ &+ \gamma Tr(W^T X X^T W) - 2\gamma Tr(W^T X F) \end{aligned} \tag{20}$$

From Eq. (20) we know N should be positive definite to guarantee the objective function is convex w.r.t. F, otherwise the objective function in Eq. (19) is not bounded. Suppose N is positive definite, by setting the derivative of Eq. (20) w.r.t. F to zero, we have

$$F = \gamma N X^T W \tag{21}$$

Substituting F into Eq. (20), we have

$$g(\lambda) = \min_{W^T W = I} Tr W^T X (I - \gamma N) X^T W \tag{22}$$

The optimal solution W consists of the m eigenvectors of $X(I - \gamma N)X^T$ corresponding to the smallest eigenvalues.

If we have an initial value λ_0 satisfying the following two conditions: $N_0 = (L - \lambda_0 L_q + \gamma I)^{-1}$ is positive definite (*i.e.*, the smallest eigenvalues of N_0 is larger than 0) and $g(\lambda_0) \leq 0$ (*i.e.*, the sum of the m smallest eigenvalues of $X(I - \gamma N_0)X^T$ is not larger than 0), we will have the algorithm to solve the proposed objective. The detailed algorithm to solve the problem (18) is described in Algorithm 1.

In the following subsections, we will prove our algorithm converges to the global optimal solution and provide the approach to find a λ_0 to satisfy the above two conditions.

5.2 Convergence Analysis of Our Algorithm

Denote

$$J(F, W) = \frac{Tr(F^T L F) + \gamma \left\| X^T W - F \right\|_F^2}{Tr(F^T L_q F)} \tag{23}$$

Assume $\lambda^* = J(F^*, W^*)$ is the global optimal value of the objective function in Eq. (18). Denote

$$h(F, W; \lambda) = Tr F^T (L - \lambda L_q) F + \gamma \left\| X^T W - F \right\|_F^2 \tag{24}$$

then $g(\lambda) = \min_{F, W^T W = I} h(F, W; \lambda)$.

Similar to the standard trace ratio problem [17], we have the following results.

Algorithm 1. Algorithm to solve the problem (18)

```
Input: $X$, Positive semi-definite matrices $L$ and $L_q$, $\gamma$, $m$.
Initialize $\lambda_0$ such that $N_0 = (L - \lambda_0 L_q + \gamma I)^{-1}$ is positive definite and $g(\lambda_0) \leq 0$.
Let $t = 1$.
repeat
    1. Calculate $N_{t-1} = (L - \lambda_{t-1} L_q + \gamma I)^{-1}$.
    2. Calculate $W_t$, in which the columns are the $m$ eigenvectors of $X(I - \gamma N_{t-1})X^T$ corresponding to the smallest eigenvalues.
    3. Calculate $F_t = \gamma N_{t-1} X^T W_t$.
    4. Calculate $\lambda_t = \frac{Tr(W_t^T X(I-\gamma N_{t-1})X^T W_t)}{\gamma Tr(W_t^T X N_{t-1} L_q N_{t-1} X^T W_t)} + \lambda_{t-1}$.
    5. Let $t = t + 1$.
until Converge
Output: $F$, $W$.
```

Lemma 1. *The below three equations hold:*

$$g(\lambda) = 0 \Rightarrow \lambda = \lambda^* \tag{25}$$

$$g(\lambda) > 0 \Rightarrow \lambda < \lambda^* \tag{26}$$

$$g(\lambda) < 0 \Rightarrow \lambda > \lambda^* \tag{27}$$

Proof: Since $\lambda^* = J(F^*, W^*)$ is the global optimal value, $\forall F, W^T W = I$, we have $J(F, W) \geq \lambda^*$. So $h(F^*, W^*; \lambda^*) = 0$ and $h(F, W; \lambda^*) \leq 0$. Thus $\min\limits_{F, W^T W = I} h(F, W; \lambda^*) = 0$, that is, $g(\lambda^*) = 0$. Similarly we can get Eq. (25).

If $\lambda \geq \lambda^*$, then

$$\begin{aligned} g(\lambda) &= \min_{F, W^T W = I} h(F, W; \lambda) \leq h(F^*, W^*; \lambda) \\ &= g(\lambda^*) + (\lambda^* - \lambda) Tr(F^{*T} L_p F^*) \\ &= (\lambda^* - \lambda) Tr(F^{*T} L_p F^*) \\ &\leq 0, \end{aligned} \tag{28}$$

which concludes Eq. (26).

If $\lambda \leq \lambda^*$, then

$$\begin{aligned} & g(\lambda) \\ = & \min_{F, W^T W = I} h(F, W; \lambda) \\ = & \min_{F, W^T W = I} h(F, W; \lambda^*) + (\lambda^* - \lambda) Tr(F^T L_p F) \\ \geq & \min_{F, W^T W = I} h(F, W; \lambda^*) + \min_F (\lambda^* - \lambda) Tr(F^T L_p F) \\ = & g(\lambda^*) + (\lambda^* - \lambda) \min_F Tr(F^T L_p F) \\ = & 0, \end{aligned} \tag{29}$$

which concludes Eq. (27). □

Theorem 1. *In each iteration of Algorithm 1, the value of the objective function in Eq. (18) will not increase.*

Proof: According to Step 3 in Algorithm 1, $F_t = \gamma N_{t-1} X^T W_t$, and notice $N_{t-1} = (L - \lambda_{t-1} L_q + \gamma I)^{-1}$ according to Step 1, hence we have

$$\begin{aligned} J(F_t, W_t) &= \frac{Tr(F_t^T L F_t) + \gamma \left\| X^T W_t - F_t \right\|_F^2}{Tr(F_t^T L_q F_t)} \\ &= \frac{Tr W_t^T X (I - \gamma N_{t-1}) X^T W_t}{\gamma Tr W_t^T X N_{t-1} L_q N_{t-1} X^T W_t} + \lambda_{t-1} \\ &= \lambda_t \,. \end{aligned} \tag{30}$$

Thus, $\lambda_t = J(F_t, W_t) \geq J(F^*, W^*) = \lambda^*$. According to Eq. (26) in Lemma 1, $g(\lambda_t) \leq 0$. On the other hand, according to the condition of λ_0, we have $g(\lambda_0) \leq 0$. Therefore, for $t \geq 0$, $g(\lambda_t) \leq 0$.

According to Steps 2 and 3, $\{F_{t+1}, W_{t+1}\}$ are the optimal solutions to $g(\lambda_t)$, so $g(\lambda_t) = h(F_{t+1}, W_{t+1}; \lambda_t)$. Therefore, for $t \geq 0$, we have

$$\begin{aligned} & g(\lambda_t) \leq 0 \\ \Rightarrow\; & h(F_{t+1}, W_{t+1}; \lambda_t) \leq 0 \\ \Rightarrow\; & \frac{Tr(F_{t+1}^T L F_{t+1}) + \gamma \left\| X^T W_{t+1} - F_{t+1} \right\|_F^2}{Tr(F_{t+1}^T L_q F_{t+1})} \leq \lambda \\ \Rightarrow\; & J(F_{t+1}, W_{t+1}) \leq J(F_t, W_t) \,, \end{aligned} \tag{31}$$

which completes the proof.

□

Note that $J(F_t, W_t)$ has lower bound, thus the Algorithm 1 will converge.

Theorem 2. *The Algorithm 1 converges to the global optimal solution.*

Proof: According to Step 4 in Algorithm 1,

$$\lambda_{t+1} = \frac{Tr W_{t+1}^T X (I - \gamma N_t) X^T W_{t+1}}{\gamma Tr W_{t+1}^T X N_t L_q N_t X^T W_{t+1}} + \lambda_t \,. \tag{32}$$

Note that $\lambda_{t+1} = \lambda_t$ in the convergence. Therefore

$$\begin{aligned} & \lambda_{t+1} = \frac{Tr W_{t+1}^T X (I - \gamma N_t) X^T W_{t+1}}{\gamma Tr W_{t+1}^T X N_t L_q N_t X^T W_{t+1}} + \lambda_{t+1} \\ \Rightarrow\; & \frac{Tr W_{t+1}^T X (I - \gamma N_t) X^T W_{t+1}}{\gamma Tr W_{t+1}^T X N_t L_q N_t X^T W_{t+1}} = 0 \\ \Rightarrow\; & Tr W_{t+1}^T X (I - \gamma N_t) X^T W_{t+1} = 0 \\ \Rightarrow\; & g(\lambda_t) = 0. \end{aligned} \tag{33}$$

According to Eq. (25) in Lemma 1, $\lambda_t = \lambda^*$. Therefore, the converged solution of Algorithm 1 is the global optimal solution. □

To study the convergence rate of our algorithm, we prove the following theorem.

Theorem 3. *The Algorithm 1 is a Newton's method to find the root of* $g(\lambda) = 0$.

Proof: Denote the i-th smallest eigenvalue of $X(I - \gamma N_t)X^T$ by $\beta_i(\lambda_t)$ and the corresponding eigenvector by $w_i(\lambda_t)$. According to the definition of eigenvalues and eigenvectors, we have:

$$\begin{aligned}
&(X(I - \gamma N_t)X^T - \beta_i(\lambda_t)I)w_i(\lambda_t) = 0 \qquad (34)\\
\Rightarrow\ &\frac{\partial(X(I - \gamma N_t)X^T - \beta_i(\lambda_t)I)w_i(\lambda_t)}{\partial\lambda_t} = 0\\
\Rightarrow\ &(-\gamma X N_t L_q N_t X^T - \beta_i'(\lambda_t)I)w_i(\lambda_t) +\\
&(X(I - \gamma N_t)X^T - \beta_i(\lambda_t)I)w_i'(\lambda_t) = 0\\
\Rightarrow\ &w_i^T(\lambda_t)(-\gamma X N_t L_q N_t X^T - \beta_i'(\lambda_t)I)w_i(\lambda_t) +\\
&w_i^T(\lambda_t)(X(I - \gamma N_t)X^T - \beta_i(\lambda_t)I)w_i'(\lambda_t) = 0\\
\Rightarrow\ &w_i^T(\lambda_t)(-\gamma X N_t L_q N_t X^T - \beta_i'(\lambda_t)I)w_i(\lambda_t) = 0\\
\Rightarrow\ &\beta_i'(\lambda_t) = -\gamma w_i^T(\lambda_t) X N_t L_q N_t X^T w_i(\lambda_t)
\end{aligned}$$

From Eq. (22) we know, $g(\lambda) = \min_{W^TW=I} trW^TX(I - \gamma N)X^TW$, so $g(\lambda_t) = \sum_{i=1}^{m} \beta_i(\lambda_t)$. Then we have:

$$\begin{aligned}
g'(\lambda_t) &= \sum_{i=1}^{m} \beta_i'(\lambda_t)\\
&= \sum_{i=1}^{m} -\gamma w_i^T(\lambda_t) X N_t L_q N_t X^T w_i(\lambda_t)\\
&= -\gamma Tr(W_{t+1}^T X N_t L_q N_t X^T W_{t+1})\,. \qquad (35)
\end{aligned}$$

According to Step 4 in Algorithm 1, we have:

$$\begin{aligned}
\lambda_{t+1} &= \frac{Tr(W_{t+1}^T X(I - \gamma N_t)X^T W_{t+1})}{\gamma Tr(W_{t+1}^T X N_t L_q N_t X^T W_{t+1})} + \lambda_t\\
&= \lambda_t - \frac{g(\lambda_t)}{g'(\lambda_t)}\,. \qquad (36)
\end{aligned}$$

Thus the iterative procedure of Algorithm 1 is essentially a Newton's method to find the root of $g(\lambda) = 0$.

□

It is well known the rate of convergence of Newton's method is quadratic convergence under mild conditions, which is very fast to converge in practice. In our experiments, we find that the Algorithm 1 indeed converges very fast, and always converges within 5-20 iterations.

5.3 Approach to Find An Initial λ_0

Lemma 1 can be used to find a feasible λ_0 that satisfies the following two conditions: $N_0 = (L - \lambda_0 L_q + \gamma I)^{-1}$ is positive definite and $g(\lambda_0) \leq 0$.

Algorithm 2. Find a feasible value λ_0

```
Initialize $F$ and $W$ such that $W^T W = I$. Let:
    $\lambda_{\min} = 0$ and $\lambda_{\max} = \frac{Tr(F^T LF)+\gamma\|X^T W-F\|_F^2}{Tr(F^T L_q F)}$.
repeat
  Let $\lambda_0 = \frac{\lambda_{\min}+\lambda_{\max}}{2}$, $N_0 = (L - \lambda_0 L_q + \gamma I)^{-1}$.
  if the smallest eigenvalue of $N_0$ is not larger than 0 then
    $\lambda_{\max} \leftarrow \lambda_0$.
  end if
  if the sum of the $m$ smallest eigenvalues of $X(I - \gamma N_0)X^T$ is larger than 0 then
    $\lambda_{\min} \leftarrow \lambda_0$.
  end if
until $N_0$ is positive definite and the sum of the $m$ smallest eigenvalues of $X(I - \gamma N_0)X^T$ is
not larger than 0
```

We apply bisection method to find such a λ_0. First, we evaluate the lower bound $\lambda_{\min}$ and upper bound $\lambda_{\max}$ of such a λ_0. According to Lemma 1, $g(\lambda_0) \leq 0$ indicates $\lambda_0 \geq \lambda^*$. If L and L_q are positive semi-definite, $\lambda^* \geq 0$, so we can set the initial lower bound $\lambda_{\min} = 0$. [1] Randomly initialize F and W such that $W^T W = I$, we have $J(F, W) \geq \lambda^*$, so we can set the initial upper bound $\lambda_{\max} = J(F, W)$. With the initial lower and upper bounds $\lambda_{\min}$ and $\lambda_{\max}$, we can use the bisection method to find a feasible λ_0 satisfying the two conditions. If N_0 is not positive, then the current λ_0 is too large, we update the upper bound $\lambda_{\max}$ with the current λ_0. If $g(\lambda_0) > 0$, then $\lambda_0 < \lambda^*$, which indicates the current λ_0 is too smaller, we update the lower bound $\lambda_{\min}$ with the current λ_0. The detailed approach is described in Algorithm 2.

It is worth noting that similar method can also be used to find an initial λ_0 for solving a different problem in [9], such that the algorithm in [9] is applicable with any parameter combination. We have updated the code for [9] in the author's website.

5.4 Shift Invariance of The Algorithm

It can be easily verified that $(I - \gamma N)\mathbf{1} = \mathbf{0}$, so we have $(X + c\mathbf{1}^T)(I - \gamma N)(X + c\mathbf{1}^T)^T = X(I - \gamma N)X^T$. Thus, according to the Algorithm 1, the optimal solution W to the problem (18) is invariant to arbitrary shift vector c.

6 Experiment

We evaluate the performance of the proposed flexible shift-invariant locality and globality preserving projection (FLGPP) on six benchmark data sets with the comparison to

[1] If the symmetric matrix L is not positive and L_q is positive, we can set $\lambda_{\min}$ to the smallest eigenvalue σ of $L_q^{-1}L$ since it can be verified $\lambda^* \geq \sigma$. We can also evaluate the smallest eigenvalue of L and the largest eigenvalue of L_q using the Gershgorin circle theorem to avoid computing the eigenvalue.

four related supervised embedding approaches, including multi-class Linear discriminant analysis (LDA), locality preserving projection (LPP), shift-invariant locality preserving projection (SILPP) in §3 as well as trace ratio locality preserving projection (TLPP) in §4.1.

Table 1. The summary of six benchmark datasets used in the experiments

data name	# classes(k)	image size	# data point(n)	# training per class
AT&T [20]	40	28×23	400	4
UMIST [6]	20	112×92	575	6
BINALPHA [1]	36	20×16	1404	6
COIL20 [14]	20	32×32	1440	12
YALEB [5]	31	24×21	1984	8
AR [12]	120	32×24	840	3

6.1 Data Descriptions

We use six image benchmark data sets in our experiments, because these data typically have high dimensionality.

AT&T [8] data set has 40 distinct subjects and each subject has 10 images. We downsampled each image (standard procedure to reduce the misalignment effect) to the size of 28×23. The training number per class is 4.

UMIST faces are for multiview face recognition. This data set contains 20 persons and totally there are 575 images. All these images of UMIST database are cropped and resized into 112×92 images. The training number per class is 6.

Binary Alpha data set contains binary digits of 0 through 9 and capital A through Z with size 20×16. There are 39 examples of each class. We randomly select 6 images per class as the training data.

Columbia University Image Library (COIL-20) data set [13] consists of color images of 20 objects where the images of the objects were taken at pose intervals of 5 degree, form the front view with 0 degree. Thus, there are 72 poses per objects. The images are converted to gray-scale image and they are normalized to the size of 32×32 pixels in our experiment. We randomly pick up 12 images for each object to do the training.

Yale database B data set [5] contains single light source images of 38 subjects (10 subjects in original database and 28 subjects in extended one) under 576 viewing conditions (9 poses $\times$ 64 illumination conditions). We fixed the pose. Thus, for each subject, we obtained 64 images under different lighting conditions. The facial areas were cropped into the final images for matching [5]. The size of each cropped image in our experiments is 24×21 pixels, with 256 gray levels per pixel. Because there is a set of images which are corrupted during the image acquisition [5], we have 31 subjects. We randomly select 64 illumination conditions for all 31 subjects to create the experimental dataset with 1984 images and randomly pick up 8 images per subject to do the training.

AR face database contains 120 people with different facial expressions, lighting conditions and occlusions. Each person has 26 different images, and the image resolution is 50×40. We random select 7 images per person and downsample the each image to the size of 32×24 to obtain the experimental dataset with 840 images. Then, we randomly select 3 per class as the training dataset.

We summarize the six data sets that used in our experiments in Table 1, and some image samples of the data sets are shown in Figure 1.

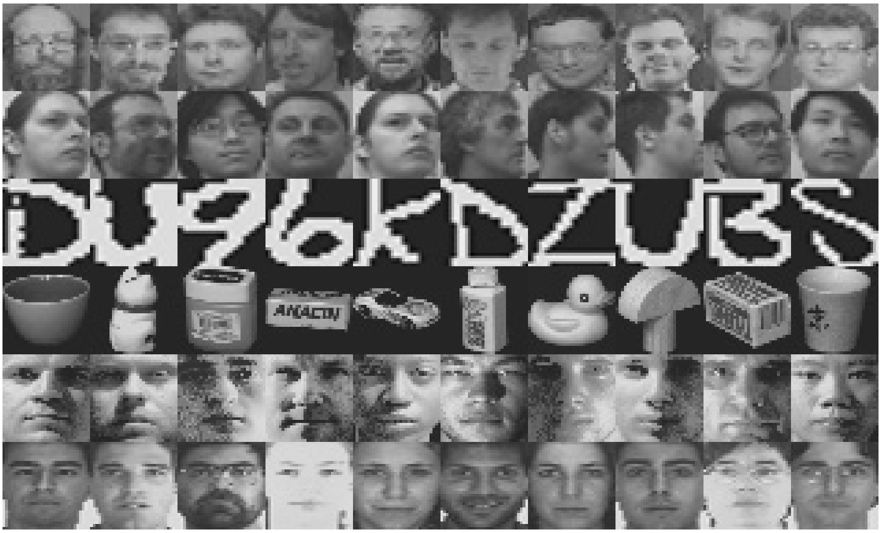
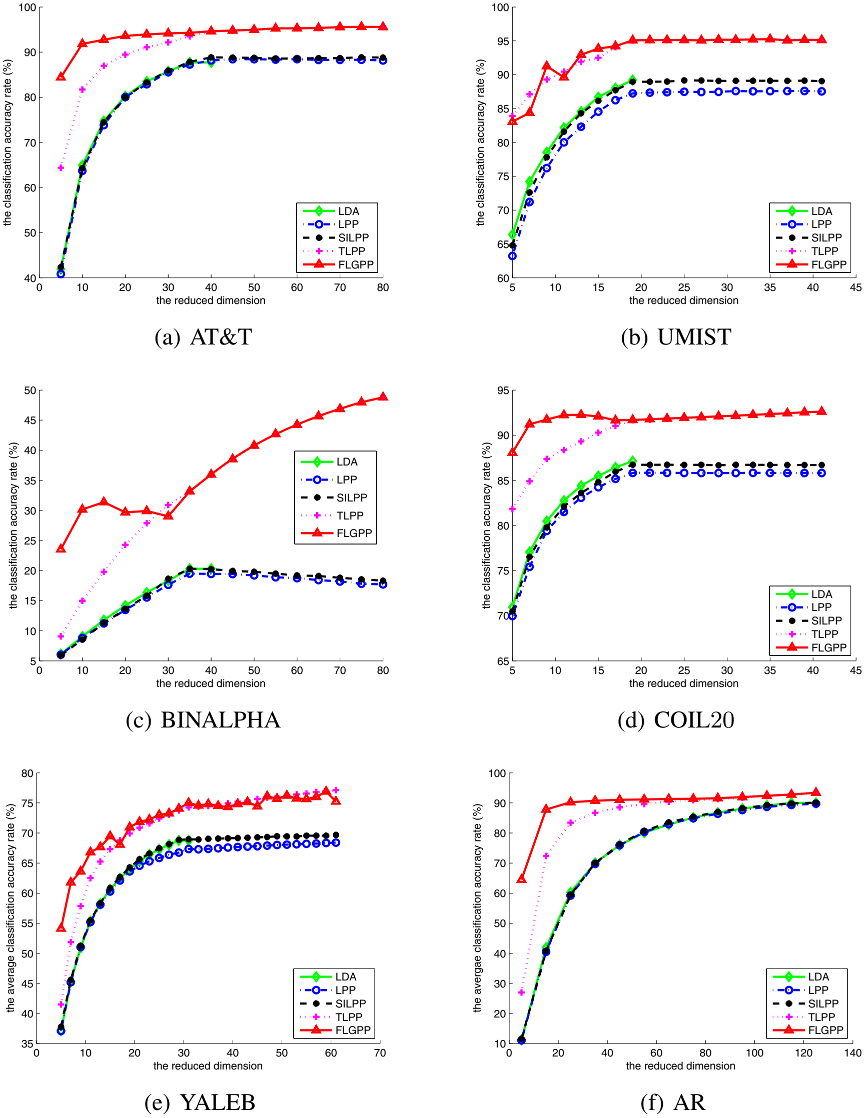


Fig. 1. Examples of the six data sets used in our experiments. From the first row to the sixth row: AT&T, UMIST, BINALPHA, COIL20, YALEB, AR.

6.2 Experiment Setup

In the training step, we firstly build the graph using the strategy described in next paragraph. Based on the same graph structure, five different embedding methods are conducted for a pre-defined reduced dimension. After we get the projection matrices for different methods, in the testing step, we use the simple k-NN (k=1) classifier (a simple classifier can avoid introducing any bias) to classify the testing data in the embedded space. In each experiment, we randomly select several data point per class for training and the rest are used as for testing. The average classification accuracy rates and standard deviations are reported over 50 random splits.

Regarding the graph construction, since we are discussing supervised embedding methods, we utilize the label information of the training data to build the graph. To be specific, $w_{i,j} = 1$, if i-th training data point and j-th training data point belong to the same class; $w_{i,j} = -1$, otherwise. We also remove the self-loop, *i.e.* let $w_{i,j} = 0$, if $i = j$. The regularization parameter in FLGPP is set to 0.1 in all the experiments. We record the average classification accuracy rate V.S. the different reduced dimensions for all the methods. For multi-class LDA, we only record its performance up to $C - 1$, where C is the number of classes.



(a) AT&T

(b) UMIST

(c) BINALPHA

(d) COIL20

(e) YALEB

(f) AR

Fig. 2. The average (50 trials with random data split) classification accuracy of k-NN method on the embedded data by five different embedding approaches

Table 2. The average classification accuracy rate ± standard deviation on six benchmark datasets among all the reduced dimension from 5 to $C - 1$

data name	MLDA	LPP	SILPP	TLPP	FLGPP
AT&T	87.70 ± 2.06	87.24 ± 1.96	87.83 ± 2.49	93.48 ± 1.72	**94.28** ± 1.91
UMIST	88.00 ± 2.63	86.24 ± 2.91	87.72 ± 2.70	94.07 ± 2.08	**94.22** ± 2.07
BINALPHA	18.33 ± 1.52	17.63 ± 1.55	18.63 ± 1.44	30.89 ± 1.92	**31.38** ± 3.24
COIL20	86.43 ± 1.53	85.16 ± 1.69	85.95 ± 1.49	91.03 ± 1.27	**92.26** ± 1.07
YALEB	68.76 ± 5.25	66.74 ± 6.22	68.87 ± 5.67	73.75 ± 2.05	**74.05** ± 2.56
AR	89.15 ± 1.18	88.68 ± 1.35	89.23 ± 1.31	92.28 ± 1.22	**92.39** ± 1.14

6.3 Experiment Results

Fig. 2 shows the average classification accuracy rate evaluated by 1-NN v.s. the number of the reduced dimension on six datasets over 50 random data split. From Fig. 2 we clearly observe that the performance of our proposed FLGPP method consistently outperforms that of the other embedding approaches, especially when the reduced dimension is low. When the reduced dimension becomes larger, the performance of FLGPP and TLPP become similar. But they still beat the other three methods largely. Table 2 demonstrates the mean ± standard deviation of the best classification accuracy rate among all the reduced dimensions from 5 to $C - 1$ for different algorithms.

7 Conclusion

In this paper, we proposed a novel flexible shift-invariant locality and globality preserving projection (FLGPP) method. A refined graph Laplacian was formulated and used to preserve the shift-invariant property. Meanwhile, the relaxed linear embedding was introduced to allow the error tolerance, such that the flexible embedding results can reach the more optimal manifold structures. Because the proposed new objective is very difficult to solve, we derived a new optimization algorithm with rigorously proved global convergence. Moreover, we proved the new algorithm is a Newton method with the quadratic convergence rate. We evaluated our FLGPP method on six benchmark data sets. In all empirical results, our new method is consistently better than the related methods.

References

1. Belhumeur, P., Hespanha, J., Kriegman, D.: Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. IEEE Transactions on Pattern Analysis and Machine Intelligence 19(7), 711–720 (1997)
2. Belkin, M., Niyogi, P.: Laplacian eigenmaps for dimensionality reduction and data representation. Neural Computation 15(6), 1373–1396 (2003)
3. Cai, D.: Spectral Regression: A Regression Framework for Efficient Regularized Subspace Learning. PhD thesis, Department of Computer Science, University of Illinois at Urbana-Champaign (May 2009)

4. Fisher, R.A.: The use of multiple measurements in taxonomic problems. Annals Eugen. 7, 179–188 (1936)
5. Georghiades, A.S., Belhumeur, P.N., Kriegman, D.J.: From few to many: Illumination cone models for face recognition under variable lighting and pose. IEEE Trans. Pattern Anal. Mach. Intell. 23(6), 643–660 (2001)
6. Graham, D.B., Allinson, N.M.: Characterizing virtual eigensignatures for general purpose face recognition. in face recognition: From theory to applications. NATO ASI Series F, Computer and Systems Sciences 163, 446–456 (1998)
7. He, X., Niyogi, P.: Locality preserving projections. In: NIPS (2003)
8. http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html
9. Huang, Y., Xu, D., Nie, F.: Semi-supervised dimension reduction using trace ratio criterion. IEEE Trans. Neural Netw. Learning Syst. 23(3), 519–526 (2012)
10. Jia, Y., Nie, F., Zhang, C.: Trace ratio problem revisited. IEEE Transactions on Neural Networks 20(4), 729–735 (2009)
11. Jolliffe, I.T.: Principal Component Analysis, 2nd edn. Springer, New York (2002)
12. Martinez, A.: The ar face database. CVC Technical Report, 24 (1998)
13. Nayar, S., Nene, S., Murase, H.: Real-time 100 object recognition system. In: Proceedings of the 1996 IEEE International Conference on Robotics and Automation, vol. 3, pp. 2321–2325. IEEE (1996)
14. Nene, S.A., Nayar, S.K., Murase, H.: Columbia object image library (COIL-20), Technical Report CUCS-005-96. Columbia University (1996)
15. Ng, A.Y., Jordan, M.I., Weiss, Y.: On spectral clustering: Analysis and an algorithm. In: NIPS, pp. 849–856 (2001)
16. Nie, F., Xiang, S., Song, Y., Zhang, C.: Orthogonal locality minimizing globality maximizing projections for feature extraction. Optical Engineering 48, 017202 (2009)
17. Nie, F., Xiang, S., Zhang, C.: Neighborhood minmax projections. In: IJCAI, pp. 993–998 (2007)
18. Nie, F., Xu, D., Tsang, I.W.-H., Zhang, C.: Flexible manifold embedding: A framework for semi-supervised and unsupervised dimension reduction. IEEE Transactions on Image Processing 19(7), 1921–1932 (2010)
19. Roweis, S.T., Al, E.: Nonlinear dimensionality reduction by locally linear embedding. Science 290, 2323–2326 (2000)
20. Samaria, F.S., Harter, A.C.: Parameterisation of a stochastic model for human face identification. In: 2nd IEEE Workshop on Applications of Computer Vision, pp. 138–142 (1994)
21. Shi, J., Malik, J.: Normalized cuts and image segmentation. IEEE Transactions on PAMI 22(8), 888–905 (2000)
22. Tenenbaum, J.B., de Silva, V., Langford, J.C.: A global geometric framework for nonlinear dimensionality reduction. Science 290(5500), 2319–2323 (2000)
23. Wang, H., Yan, S., Xu, D., Tang, X., Huang, T.S.: Trace ratio vs. ratio trace for dimensionality reduction. In: CVPR (2007)
24. Xiang, S., Nie, F., Zhang, C., Zhang, C.: Nonlinear dimensionality reduction with local spline embedding. IEEE Transactions on Knowledge and Data Engineering 21(9), 1285–1298 (2009)
25. Zhang, Z., Zha, H.: Principal manifolds and nonlinear dimension reduction via local tangent space alignment. SIAM Journal of Scientific Computing 26, 313–338 (2005)