We are IntechOpen, the world’s leading publisher of Open Access books
Built by scientists, for scientists

3,800
Open access books available

116,000
International authors and editors

120M
Downloads

154
Countries delivered to

TOP 1%
Our authors are among the most cited scientists

12.2%
Contributors from top 500 universities

WEB OF SCIENCE™
Selection of our books indexed in the Book Citation Index in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com
Chapter 3

A Generally Semisupervised Dimensionality Reduction Method with Local and Global Regression Regularizations for Recognition

Mingbo Zhao, Yuan Gao, Zhao Zhang and Bing Li

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/63273

Abstract

The insufficiency of labeled data is an important problem in image classification such as face recognition. However, unlabeled data are abundant in the real-world application. Therefore, semisupervised learning methods, which corporate a few labeled data and a large number of unlabeled data into learning, have received more and more attention in the field of face recognition. During the past years, graph-based semisupervised learning has been becoming a popular topic in the area of semisupervised learning. In this chapter, we newly present graph-based semisupervised learning method for face recognition. The presented method is based on local and global regression regularization. The local regression regularization has adopted a set of local classification functions to preserve both local discriminative and geometrical information, as well as to reduce the bias of outliers and handle imbalanced data; while the global regression regularization is to preserve the global discriminative information and to calculate the projection matrix for out-of-sample extrapolation. Extensive simulations based on synthetic and real-world datasets verify the effectiveness of the proposed method.

Keywords: Semi-supervised Learning, Dimensionality Reduction, Local and Global Regressions, Face Recognition, Transductive and Inductive Learning

1. Introduction

In the real world, there are ever-increasing vision face data generated from Internet surfing and daily social communication. These metadata can be labeled or unlabeled, and accordingly be
utilized for image retrieval, summarization, and indexing. To handle these datasets for realizing the above tasks, automatic annotation is an elementary step, which can be formulated as a pattern classification problem and accomplished by learning-based techniques. Traditionally, the supervised-learning-based methods, such as Linear discriminant analysis (LDA) and Support Vector Machine (SVM), can deliver satisfactory recognition accuracy given that the number of labeled data is adequate. But labeling a huge amount of data is expensive and time consuming. On the other hand, the unlabeled data are sufficient and can be easily obtained from real-world application. Therefore, semisupervised learning-based methods that utilize a few of labeled data and a huge amount of unlabeled data are becoming more and more popular than only relying on the supervised learning methods [27–33].

Recently, since two pioneer semisupervised methods, i.e., Gaussian Fields and Harmonic Functions (GFHF) and Learning with Local and Global Consistency (LLGC), have been proposed in 2003 and 2004, respectively, graph-based semisupervised learning methods have received considerable research interest in the area of semisupervised learning. These methods usually represent both labeled and unlabeled sets by a graph, and then utilize their graph Laplacian matrix to characterize the manifold structure. Finally, different learning tasks such as image classification, clustering, and dimensionality reduction are performed on the graph Laplacian matrix. For example, GFHF and LLGC work in a transductive way by directly propagating the class label information from the labeled set to the unlabeled set along the graph, where the labels of unlabeled data can be estimated. Other similar works include Random Walk [5] and Special Label Propagation (SLP) [8]. However, the transductive learning methods cannot predict the class labels of new-coming samples, hence suffering the out-of-sample problem.

To solve the out-of-sample problem, inductive learning methods are proposed during the past decades. Typical methods for inductive learning are Manifold Regularization (MR) [1] and Semisupervised Discriminant Analysis (SDA) [2]. The MR tries to learn a projection matrix by adding the graph Laplacian regularized term to the cost function of original supervised methods. Therefore, both unlabeled and new-coming data can be cast into a low-dimensional subspace, hence the out-of-sample problem can be naturally solved [7, 9, 10, 16]. For example, MR has extended the regularized least square and SVM to their semisupervised learning extensions, i.e., Laplacian regularized least squares (Lap-RLS) and Laplacian SVM by adding a manifold regularized term. Similarly, Cai et al. [2] have extended LDA to SDA for semisupervised dimensionality reduction.

It should be noted that the success of semisupervised learning is based on how to utilizing the unlabeled data for characterizing the distribution of labels in data space. Several methods including Locally Linear Reconstruction [11, 12, 20], Local Regression and Global Alignment [13, 14], and Local Spline Regression [18, 19] have been developed to discover the intrinsic manifold structure of data. However, when we do semisupervised classification, the data points lying far away the data manifold are noisy for learning the correct classifier and can deteriorate the classification performance. On the other hand, sampling in real-world applications is usually not uniform. As a result, the sampled data may be imbalanced or with multi-density distribution. None of the aforementioned methods focus on solving the two problems.
In this chapter, we develop an effective semisupervised dimensionality reduction method, i.e., Local and Global Regression (LGR), for face recognition with outliers and imbalanced face data. In order to both handle transductive and inductive learning problems, LGR aims to sufficiently learn the classification function by using all data. In detail, the presented method first extends the original supervised regression term to a supervised loss term and a global regression regularized term, where the loss term is to fix the inconsistency between the predicted labels and initial labels, while the global regression term is to sufficiently learn the classification function using all training data and to obtain the projection matrix for handling out-of-sample problem. Furthermore, to capture the local discriminative information, a set of weighted local classification functions are adopted for each dataset to estimate the labels of its nearby data, where the weight is to reduce the outliers bias and to deal with imbalanced data. Thus, both local and global discriminative information of dataset can be preserved by the proposed LGR method.

The main contributions of this work are as follows: (1) we propose a new effective method for semisupervised dimensionality reduction, which can handle both transductive and inductive learning problems; (2) we develop a graph Laplacian matrix, which can characterize both local geometrical and discriminative information, as well as reduce the bias of outliers and handle imbalanced data; (3) we have also established the connection between the proposed method and other state-of-the-art methods. Theoretical analysis has shown that many popular semisupervised methods such as LRGA can be viewed as the special cases of the proposed method. Extensive simulations based on synthetic and real-world datasets verify the effectiveness of the proposed method.

This chapter is organized as follows. In Section 2, the notations and motivations are first given. We then propose our LGR method for both handling transductive and inductive learning problems. Finally, we also establish the connection between the proposed method and other state-of-the-art methods. Section 3 demonstrates the extensive simulations and the final conclusions are drawn in Section 4.

2. The proposed method

2.1. Notation and motivation

In semi-supervised learning, we define $X = \{X_l, X_u\} = \{x_1, x_2, \ldots, x_{l+u}\} \in R^{D \times (l+u)}$ be the data matrix where the first $l$ and the remaining $u$ columns are the labeled and unlabeled samples, respectively; $Y_l = \{y_1, y_2, \ldots, y_l\} \in R^{c \times l}$ be the binary label matrix with each column $y_i$ representing the class assignment of $x_i$, i.e. $y_{ij} = 1$, as the class matrix, where $y_{ij} = 1$, if $x_i$ belongs to the $i$th class; $y_{ij} = 0$, otherwise, $D$ and $c$ are the numbers of features and classes, respectively. We also let $L = D - W$ be the graph Laplacian matrix associated with both labeled and unlabeled sets [17], where $W$ is the weight matrix defined as $w_{ij} = \exp(-\|x_i - x_j\|^2 / 2\sigma^2)$, if $x_i$ is within the $k$
nearest neighbor of $x_i$ or if $x_j$ is within the $k$ nearest neighbor of $x_i$; $w_{ij} = 0$, otherwise, $D$ is a diagonal matrix satisfying $D_{ii} = \sum_{j=1}^{l+u} w_{ij}$.

Most semi-supervised learning methods utilize the Gaussian function based affinity matrix. As pointed out in references [11, 12], the Gaussian function based affinity matrix is found to be oversensitive to the Gaussian variance; only a slight variation on the variance may affect the results dramatically. Thus, Gaussian function based affinity matrix is not a good method for handling image classification. The method developed should be robust to the parameters.

Second, when carrying out semi-supervised classification, the samples lying far away from the data manifold are outliers which may lead to learn an incorrect classifier and deteriorate the classification performance. Considering Figure 1(a and b) as examples, we generalize a two-cycle and two-moon datasets with outliers. Considering the distribution of two data, the ideal decision boundary should lie in the gap between two data sub-manifolds. However, since there are many outliers around the data manifold, these outliers will blur the clear distribution of the whole data and are noisy to learn a correct classifier. Therefore, it is very important to develop a method that can adaptively reduce the effects of outliers.

Third, in real-world applications, sampling is usually not uniform. Consequently, the sampled data can be imbalanced or follows multi-density distribution. Figure 1(c) shows a two-plate dataset with two classes: each class follows a Gaussian distribution but with different cores and density. Obviously, the data points (left data points) in the high-density area will take more important part than those (right data points) in the low-density area when to learn a classifier, which may cause incorrect classification results. The method developed should handle such imbalanced data with multi-density distribution.

The method developed should also solve the out-of-sample problem. To address the above problems, we, in this paper, propose a new semi-supervised learning method, which is based on local and global regression.

The method developed should also solve the out-of-sample problem. To address the above problems, we, in this paper, propose a new semi-supervised learning method, which is based on local and global regression.

Figure 1. (a) Two-cycle dataset; (b) two-moon dataset; (c) two-plate dataset.
2.2. Local and global regression

We start from the supervised least-squares regression. The least-square regression is to fix a linear model \( y_j = V^T x_j + b^T \) by regressing \( X \) on \( Y \):

\[
\min_{V, b} \sum_{j=1}^{l} \| y_j - V^T x_j - b^T \|^2_F + \alpha \| V \|^2_F,
\]

(1)

where \( V \) is the projection matrix that is to project the new-coming samples and \( b \) is the bias term. Although the label \( y_j \) of \( x_j (j \leq l) \) has already been known, since \( l \) is usually very small, the classification function \( z_j = V^T x_j + b \) may not be sufficiently trained due to the small sample size. To solve this problem, we introduce \( Z = \{ Z_l, Z_u \} = \{ z_1, z_2, \ldots, z_{l+u} \} \in \mathbb{R}^{c \times (l+u)} \) as a set of estimated labels to play the same roll by replacing \( V^T x_j + b \) with \( z_j \) and add a regression term to Eq. (1) as follows:

\[
\min_{V, b, Z} \sum_{j=1}^{l} \| z_j - y_j \|^2_F + \alpha \left( \sum_{j=1}^{l+u} \| V^T x_j + b^T - z_j \|^2_F + \eta \| V \|^2_F \right).
\]

(2)

According to Eq. (2), the classification function \( z_j = V^T x_j + b \) can be sufficiently learned by using all the predicted labels and to fix to their original labels. In other meaning, the global discriminative information can be preserved by the regression term of Eq. (2). Furthermore, to grasp the local discriminative information, we induce a local regression function for each data sample \( x_j \). We denote \( N_k(x_j) \) as the \( k \) neighborhood set of \( x_j \) with itself, \( X_j = \{ x_{j1}, x_{j2}, \ldots, x_{jk} \} \in \mathbb{R}^{D \times k} \) as the local data matrix formed by all samples in \( N_k(x_j) \), where \( \{ j1, j2, \ldots, jk \} \) is the index set of \( N_k(x_j) \) and \( j1 = j \), \( x_{j1} = x_j \). We also denote \( Z_j = \{ z_{j1}, z_{j2}, \ldots, z_{jk} \} \in \mathbb{R}^{c \times k} \) as the local low-dimensional label matrix in \( N_k(x_j) \). Then, the local regression function for all data samples can be given as follows:

\[
\min_{Z, V, b} \sum_{j=1}^{l+u} \left( \sum_{i=0}^{k-1} \| V_j^T x_{ji} + b^T_j - z_{ji} \|^2_F + \eta \| V_j \|^2_F \right).
\]

(3)

However, minimizing the above total errors over all data samples tends to force each local error \( \alpha_{ji} = \| V_j^T x_{ji} + b^T_j - z_{ji} \|^2_F \) similar to each other. Given some cases that the dataset includes some outliers, assuming all the local regression errors equally may emphasize the effects from outliers and weaken the effects from normal data. In this section, to weaken the effects from outliers, we add a weight vector \( \Gamma_j = \{ \tau_{j1}, \tau_{j2}, \ldots, \tau_{jk} \} \in \mathbb{R}^{1 \times k} \) for each local data patch \( x_j \) in order to penalize each regression error, which can be shown as
In the following section, we will discuss how to select the weight $\tau_j$. Our motivation is to let the weight of local error $\alpha_j$ be large given $x_j$ are the normal data and in the contrast to let the weight be small given $x_j$ is outlier. In detail, to obtain local projection matrix $V_j$ and bias $b_j$, we perform derivatives to Eq. (4) w.r.t. $V_j$ and $b_j$ to zeros. Then, Eq. (4) will be reduced to

$$
\min_Z \sum_{j=1}^{l+u} \left( \sum_{i=0}^{k-1} \tau_i \| V_j x_i + b_j - z_j \|_F^2 + \eta \| V_j \|_F^2 \right).
$$

where $L_j = H_j - H_j X_j \left( X_j H_j X_j^T + \eta I \right)^{-1} X_j H_j$; $S_j \in \mathbb{R}^{(l+u) \times k}$ is the selected matrix satisfying $(S_j)_{pq} = 1$, if $x_p$ is the $q$th neighbors to $x_p$; $(S_j)_{pq} = 0$, otherwise, $L_d = \sum_{j=1}^{l+u} (S_j L_j S_j^T)$ is the local graph Laplacian matrix. Similarly, by setting the derivatives of Eq. (2) w.r.t. $V$ and $b$ to zero, we have

$$
\left\{ \begin{array}{l}
\begin{aligned}
&b = \left( e Z^T - e X^T V \right) / \| e \|_F \\
&V = \left( X L_c X^T + \eta I \right)^{-1} X L_c Z^T
\end{aligned}
\end{array} \right.
$$

where $e \in \mathbb{R}^{1 \times (l+u)}$ is a unit vector and $L_c = I - e^T e / \| e \|_F^2$ is used for centering the samples by subtracting the mean of all samples. With $b$ and $V$ in Eq. (6), the global regression term in Eq. (2) can be written as

$$
\| V^T X + b^T e - Z \|_F^2 + \eta \| V \|_F^2 = \text{Tr} \left( Z L_g Z^T \right),
$$

where $L_g = L - L_c X \left( X L_c X^T + \eta I \right)^{-1} X L_c$ is the global graph Laplacian matrix. By integrating Eq. (7) with Eq. (2), we formulate our method as follows:

$$
J(Z) = \min_Z \text{Tr} \left( (Z - Y) U \left( Z - Y \right)^T \right) + \alpha_m \text{Tr} \left( Z L_m Z^T \right) + \alpha_r \text{Tr} \left( Z L_r Z^T \right),
$$

where $U \in \mathbb{R}^{(l+u) \times (l+u)}$ is the diagonal matrix with the first $l$ and the remaining $u$ diagonal elements as 1 and 0, respectively; the second term describes the local discriminative structure of data; the third term describes the global discriminative structure; and $\alpha_m$ and $\alpha_r$ are the two balancing parameters. Since both local and global regressions are regularized in our method,
we refer our method as LGR. Finally, by performing derivatives of $J(Z)$ w.r.t. $z$ to zero, we can calculate the solution of $z$ as

$$Z = YU\left(U + \alpha_m L_d + \alpha_g L_g\right)^{-1}. \quad (9)$$

Then, we can obtain the optimal projection matrix and bias term by replacing $z$ in Eq. (6).

### 2.3. Weight selection for bias reduction

In this section, we consider how to select the weights in the proposed method suggested in Section 2.2. Note, our goal of using the weights is to weaken the effects of outliers and the weight $\tau_j$ should be set to a small value if $x_{ij}$ is an outlier. Then we can make the weight $\tau_j$ inversely proportional to the distance between $x_{ij}$ and a center $\mu_j$, i.e., $\tau_j = \frac{1}{\|x_{ij} - \mu_j\|}$. Such a center is expected to represent the idea center of data in the neighborhoods of $x_{ij}$ and should be far away from outliers. Hence, the weight $\tau_j$ is usually small if $x_{ij}$ is an outlier. But this center $\mu_j$ is unknown. We next present an iterative approach to calculate $\mu_j$ and the weight $\tau_j$ simultaneously. The approach is converged and proved afterward.

1. Initialize $\mu_j^0$ as the average center of all data points in the local patch of $x_j$.
2. Update $\tau_j^t$ for each $x_{ij}$ as $\tau_j^t = \frac{1}{\|x_{ij} - \mu_j^{t-1}\|}$ and form the weight matrix $\Gamma_j^t$.
3. Update $\mu_j^t = \frac{\sum_{i=0}^{k-1} \tau_j^t x_{ij}}{\sum_{i=0}^{k-1} \tau_j^t} = X_j \Delta_j^t e / e^T \Delta_j^t e$.
4. Iterate steps 2 and 3 until $\sum_{i=0}^{k-1} \|x_{ij} - \mu_j^t\|$ no changes. Output $\tau_j^t$.

Table 1. Iterative approach for calculating the weight.

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Initialize $\mu_j^0$ as the average center of all data points in the local patch of $x_j$.</td>
</tr>
<tr>
<td>2</td>
<td>Update $\tau_j^t$ for each $x_{ij}$ as $\tau_j^t = \frac{1}{|x_{ij} - \mu_j^{t-1}|}$ and form the weight matrix $\Gamma_j^t$.</td>
</tr>
<tr>
<td>3</td>
<td>Update $\mu_j^t = \frac{\sum_{i=0}^{k-1} \tau_j^t x_{ij}}{\sum_{i=0}^{k-1} \tau_j^t} = X_j \Delta_j^t e / e^T \Delta_j^t e$.</td>
</tr>
<tr>
<td>4</td>
<td>Iterate steps 2 and 3 until $\sum_{i=0}^{k-1} |x_{ij} - \mu_j^t|$ no changes. Output $\tau_j^t$.</td>
</tr>
</tbody>
</table>

Table 1 shows the basic steps of the iterative approach. Following Table 1, the weight $\tau_j^t$ at each iteration is updated from the last $\mu_j^{t-1}$ and the newly updated center $\mu_j^t$ is calculated from current $\tau_j^t$. The whole iterations are continued until convergence, so that the weight $\tau_j^t$ can be adaptively and iteratively re-weighted to minimize $\sum_{i=0}^{k-1} \|x_{ij} - \mu_j^t\|$. In addition, as can be seen in simulation of Figure 2, the updated $\mu_j^t$ will be adaptively re-weighted to be close to the main center of most data points, while the updated $\tau_j^t$ will be weaken if $x_{ij}$ is outliers or be strengthened if $x_{ij}$ is close to the ideal center. We next discuss a theorem to guarantee the convergence of the approach of Table 1.

**Theorem 1.** The approach in Table 1 will monotonically decrease the objective function $\sum_{i=0}^{k-1} \|x_{ij} - \mu_j^t\|$ until convergence.
Proof. According to step 3 in Table 1, we know that

$$\mu'_j = \arg \min_{\mu'_j} \sum_{i=0}^{k-1} \tau_i \| x_{i,j} - \mu'_j \|^2_F,$$

(10)

where \( \tau_i = 1/\| x_{i,j} - \mu_i \|^2 \) as in step 2 of Table 1. Following Eq. (10), we have

$$\sum_{i=0}^{k-1} \left\{ \| x_{i,j} - \mu'_j \|^2 / \| x_{i,j} - \mu_i \|^2 \right\} \leq \sum_{i=0}^{k-1} \left\{ \| x_{i,j} - \mu_i \|^2 / \| x_{i,j} - \mu_i \|^2 \right\}.$$

(11)

Based on the lemma in reference [6] that \( 2\sqrt{a-b}/\sqrt{b} \leq 2\sqrt{a-b}/\sqrt{b} \) holds for any two nonzero value, we have

$$\sum_{i=0}^{k-1} 2\| x_{i,j} - \mu'_j \|^2 / \| x_{i,j} - \mu'_j \|^2 \leq \sum_{i=0}^{k-1} 2\| x_{i,j} - \mu'_j \|^2 / \| x_{i,j} - \mu'_j \|^2.$$

(12)

By summing Eqs. (11) and (12) in two sides, we have

$$\sum_{i=0}^{k-1} \| x_{i,j} - \mu'_j \|^2 \leq \sum_{i=0}^{k-1} \| x_{i,j} - \mu'_j \|^2.$$

(13)

Eq. (14) indicates that the objective function \( \sum_{i=0}^{k-1} \| x_{i,j} - \mu'_j \|^2 \) is monotonically decreased in each iteration. Since there is a lower bound in the objective function (\( \sum_{i=0}^{k-1} \| x_{i,j} - \mu'_j \|^2 \geq 0 \)), the iterative approach will certainly converge. We thus prove Theorem 1. Finally, by incorporating the weight for reducing the bias for each local regression error into Eq. (4), we can reduce the bias of outliers of data samples.

Here, in order to show the convergence of the approach, we simply show an example in Figure 2(a), where we generalize eight normal data points and two outliers in \( \mathbb{R}^2 \). Figure 2(b) shows the converged route of \( \mu \), where we start \( \mu^0 \) as the average mean of all data points and mark \( \mu^t \) in each iteration with \( t \). From Figure 2(b), we can observe that the optimal solution \( \mu^t \) will iterative close to the main center of normal data while be far away from the outliers. Figure 2(c) shows the converged curve of approach as discussed in Table 1. From Figure 2(c), we can observe that the objective \( \sum_{i=0}^{k-1} \| x_{i,j} - \mu'_j \|^2 \) will monotonically decrease until convergence. Figure 2(d) shows the converged weight of data points. From Figure 2(d), we can observe the weights of normal data points are strengthen while those of outliers can be reduced.
2.4. Normalizing graph Laplacian matrix

It can be easily proved that $L_d$ is a graph Laplacian matrix (see the Appendix). But $L_d$ may not be a normalized graph Laplacian matrix. As pointed in references [8, 23], the normalization can strengthen the local regressions in the low-density region and weaken those in the high-density region. Since the data sampling is usually uniform in practice, normalization is useful for handling the case when the density of dataset varies dramatically. In this section, we show that by choosing a special weight vector $\Gamma_j$ for each $X_j$, $L_d$ can be a normalized graph Laplacian matrix.

Specifically, let us consider a data sample $x_j$ and let $K_l$ be the index set of those neighborhoods; set $N_k(x_j)$ contains $x_j$ as a neighbor of $x_j$, i.e., if $j \in K_l$, then $x_l \in N_k(x_j)$, where $x_j$ can be denoted as $x_j$ in the neighborhood set $N_k(x_j)$, and $i = i(l,j)$ is the local index depending on $l$ and $j$.

Obviously, if $x_j$ is in the low-density area, it has sparse neighbors and $K_l$ is relatively small. As a result, its connections to other samples will be weaker than that which has large $K_l$. Here, to strengthen the connections of samples in the low-density area, we need to normalize the weights corresponding to each $K_l$. Let $\tau_{lj}$ be the weight of $x_{lj}$ and $l$ be the global index of $x_{lj}$.

We then define $\tau_{lj} = \tau^l_{lj}$ as follows:

$$\tau_{lj} = \tau^l_{lj} \leftarrow \frac{\tau^l_{lj}}{\sum_{i \in K_l} \tau^l_{lj}}. \quad (14)$$
Hence, based on this definition, we have the following theorem:

**Theorem 2.** With the normalization for each \( w_j \) as in Eq. (14), \( L_d \) is both graph Laplacian matrix and normalized graph Laplacian matrix.

**Proof.** The proof that \( L_d \) is a graph Laplacian matrix can be seen in the Appendix. In order to prove \( L_d \) is a normalized graph Laplacian matrix, we need to prove \( L_d \) can be reformulated in the form of 
\[
L_d = I - W_d
\]
and the sum of each row or column of the affinity matrix \( W_d \) is equal to 1.

Note \( L_d = \sum_{j=1}^{l+u} (S_j L_j S_j^T) \) and \( L_d = H_j - H_j X_j^T (X_j H_j X_j^T + \eta I)^{-1} X_j H_j \), where \( H_j = \Delta_j - (\Delta_j e_k e_k^T) / (e_k^T \Delta_j e_k) \), we first define the affinity matrix \( W_d \) as follows:
\[
W_d = \sum_{j=1}^{l+u} (S_j W_j^d S_j^T),
\]
where each \( W_j^d \) satisfies
\[
W_j^d = \left( \Delta_j e_k e_k^T / (e_k^T \Delta_j e_k) \right) - H_j X_j^T (X_j H_j X_j^T + \eta I)^{-1} X_j H_j.
\]

Then, \( L_d \) can be reformulated as
\[
L_d = \sum_{j=1}^{l+u} (S_j \Delta_j S_j^T) - \sum_{j=1}^{l+u} (S_j W_j^d S_j^T).
\]

Here, for each \( S_j \Delta_j S_j^T \), we have \( S_j^T e = e_k^T \Rightarrow S_j \Delta_j S_j^T e = S_j \Gamma_j^T \), where \( S_j \Gamma_j^T \in \mathbb{R}^{(l+u) \times 1} \) is a column vector by putting each \( \tau_j^l \) to its global index \( l \) corresponding to \( x_j \). We thus have
\[
\sum_{j=1}^{l+u} \left( S_j \Delta_j S_j^T \right) e = \sum_{j=1}^{l+u} \left( S_j \Gamma_j^T \right) e = e^T.
\]

The second equation holds as \( \sum_{i \in K} \tau_j^l = 1 \); hence, the sum of all \( S_j \Gamma_j^T \) in each element is equal to 1. Then, following Eq. (18), it indicates \( \sum_{j=1}^{l+u} (S_j \Delta_j S_j^T) \) is an identity matrix, i.e., \( \sum_{j=1}^{l+u} (S_j \Delta_j S_j^T) = I \). Then based on the above analysis, we can reformulate \( L_d \) in the form of 
\[
L_d = I - W_d.
\]
In addition, since \( L_d \) is a graph Laplacian matrix (as proved in the Appendix), it satisfies \( L_d e^T = 0 \), then we have
which indicates that the sum of each column or row of \( W_d \) is equal to 1. We thus prove the theorem. Theorem 2 indicates that by choosing a special weight vector \( \tau_j \) for each \( x_j \), \( L_d \) can be both graph Laplacian matrix and normalized graph Laplacian matrix.

Here, it should be noted that if \( x_l \) is an outlier, its local weights can be significantly decreased, whether taking \( x_l \) as a neighbor of itself or of other data points. Otherwise, the normalization does not change the magnitude of its original local weights. For some data points in the low-density area, normalizing the weights can increase the information convection through those points. Finally, the basic steps of the proposed LGR are given in Table 2 and the flowchart by utilizing the proposed LGR method for face recognition is given in Figure 3.

**Algorithm:**

1. Determine the weight for each local patch based on Table 1.
2. Normalize the weight as in Eq. (14).
3. Form local regression regularized term \( L_d \) as in Eq. (5) with special local weight vector.
4. Form global regression regularized term \( L_g \) as in Eq. (7).
5. Solve the regression problem as in Eq. (8):

\[
J(Z) = \min_Z \text{Tr} \left( (Z - Y)U(Z - Y)^T \right) + \alpha_m \text{Tr} \left( ZL_dZ^T \right) + \alpha_g \text{Tr} \left( ZL_gZ^T \right),
\]

and calculate estimated label matrix \( Z^* = YU(U + \alpha_m L_d + \alpha_g L_g)^{-1} \) as in Eq. (9). Output

\[
V^* = (XL_cX^T + \eta I)^{-1} XL_cZ^T.
\]

6. Calculate the projection matrix \( V^* \) by replacing \( z^* \) to Eq. (6) as \( V^* = (XL_cX^T + \eta I)^{-1} XL_cZ^T \). Output \( V^* \).

**Table 2.** The proposed LGR.
2.5. Discussion and relative work

In this section, we discuss the relationship of Learning from Local and Global Information (LLGDI) with other state-of-the-art methods including MR, Flexible Manifold Embedding (FME), and Local Regression and Global Alignment (LRGA).

2.5.1. Relationship to manifold regularization (Lap-RLS/L) [1]

The goal of MR [1] is to develop a semisupervised learning strategy by extending the original supervised methods, such as RLS and SVM to their semisupervised learning versions, i.e., Laplacian RLS and Laplacian SVM. For example, Lap-RLS/L is to fix a linear model $y_j = V^T x_j + b^T$ by regressing $X$ on $Y$ and simultaneously to preserve the manifold smoothness in the embeddings of both the labeled and the unlabeled set. The objective function of Lap-RLS/L can be given as

$$J(V,b) = \min \sum_{j=1}^{l} \left\| V^T x_j + b^T - y_j \right\|^2_F + \alpha_l \left\| \alpha_m \right\|^2_F + \alpha_m Tr \left\{ V^T X L X^T V \right\}.$$  \hspace{1cm} (20)

However, it can be observed that Lap-RLS/L cannot sufficiently train the classification function due to the utilization of labeled samples, though it uses manifold term as complementary. Hence, the proposed LGR is superior to Lap-RLS/L.

2.5.2. Relationship to FME [7, 10]

Nie et al. has proposed another unified framework, i.e., FME [7, 10], for semisupervised dimensionality reduction, in which they verify that LLGC, GFHF, and Lap-RLS/L are only special cases in the framework. The basic objective function of FME can be given as

$$J(V,Z,b) = \min \sum_{i=1}^{l} \left\| x_i - y_i \right\|^2_F + \alpha_m Tr \left\{ Z L Z^T \right\} + \alpha_e \left\| V^T X + b^T e - Z \right\|^2_F + \eta \left\| \alpha_m \right\|^2_F.$$  \hspace{1cm} (21)

It can be observed that Eq. (22) is almost the same as the objective function of LGR in Eq. (10), when we consider $L_d \rightarrow L$. However, LGR has utilized a weighted and normalized local discriminative Laplacian matrix to preserve manifold and discriminative structure in a dataset. This is a better way than only relying on neighborhood graph.
2.5.3. Relationship to LRGA \[13, 14\]

Recently, Yang et al. has proposed semisupervised transductive learning method, namely, LRGA \[13, 14\], for multimedia retrieval. They share the similar concept with the proposed method. The basic objective function of LRGA can be given as

$$J(Z) = \min_{Z,y,b_j} \sum_{j=1}^{t} \left( z_j - y_j \right)^2 + \alpha_m \sum_{j=1}^{l+u} \left( \sum_{i=1}^{l} \frac{y_j^T x_i + b_j - z_j}{2} + \frac{y_j}{2} \right)^2. \quad (22)$$

It can be noted that LRGA is a special case of LGR when $\alpha_r = 0$. Therefore, LRGA is only a transductive learning method and cannot handle the out-of-sample problem, while LGR is a transductive and inductive learning method. Another superiority of LGR over LRGA is that LGR has adopted a weighted normalized each local regression term. Thus, as shown in the simulation results, LLGDI can handle outliers and multi-density dataset remarkably.

3. Simulation results

In this section, we will evaluate the proposed LGR based on three synthetic datasets and two real-world datasets.

3.1. Synthetic datasets

In this section, we evaluate the performance of the proposed LGR and SLP for transductive learning. The SLP is an extensive method to GFHF, LLGC, and Random Walk (RW) hence, it is representative. Here, we utilize two-moon and two-cycle datasets in Figure 1(a and b) for

![Figure 4](http://dx.doi.org/10.5772/63273)

**Figure 4.** Toy examples for transductive learning: (a) and (d) the original data of two-moon and two-cycle datasets; (b) and (e) the results of LGR; (c) and (f) the results of SLP.
evaluation. Figure 4 shows the results of LGR and SLP for transductive learning. From Figure 4, we can see that LGR can achieve better simulation result than SLP, in a way that less data are misclassified in LGR than SLP. This indicates the proposed LGR is robust to the outliers.

We also evaluate the inductive performance of the proposed LGR for handling the out-of-sample problem. Figure 5 shows the gray images of decision surfaces and boundaries learned by LGR, which are formed as follows: for each pixel, we form the its gray value as the difference from each pixel to its nearest labeled data of different classes in the reduced subspace. Here, we set the reduced dimensionality as 1. Then, we form the decision boundaries by the pixels with the value 0. Following Figure 5, we can observe that the proposed LGR can learn clear decision boundary that can well separate two classes, which verifies the effectiveness of LGR for handling the out-of-sample problem.

To show the merit of normalization, we utilize two-plate dataset in Figure 1(c) for evaluation. Our goal is to show LGR can handle multi-density dataset. Figure 6 shows the gray images of decision surfaces and boundaries learned by LGR without normalization and LGR with normalization. From Figure 6, we can observe that LGR without normalization cannot find proper boundary. However, LGR with normalization can achieve better performance, as there are less missing-classified data points separated by the decision boundary, which becomes more distinctive and accurate. The improved results are believed to be due to the fact that normalization can strengthen the local regressions in the low-density region and weaken those in the high-density region. This is proved to be advantageous to be used for multi-density dataset.

Figure 5. Toy examples for inductive learning: decision surfaces and boundaries learned by LGR. (a) and (c) Two-moon dataset; (b) and (d) two-cycle dataset.
3.2. Semisupervised face recognition based on real-world benchmark datasets

For handling the face recognition problem, we use three real-world face datasets to evaluate the performance of methods, which include UMNIST: cannot find the full name [24], Extended Yale-B [25], and Massachusetts Institute of Technology Center for Biological and Computational Learning (MIT-CBCL) [26] datasets. The UMIST dataset is a multi-view face dataset, consisting of 1012 images of 20 peoples, each covering a wide range of poses from profile to frontal views. Therefore, the UMIST has widely been used for general purpose face recognition under different face poses. The size of each image is 112×92 with 256 gray levels per pixel. In our simulation, we down-sample the size of each image to 28×23 and no other preprocessing is performed. The Extended Yale-B dataset contains 16,123 images of 38 human subjects under 9 poses and 64 illumination conditions. Because of the illumination variability, the same object can appear dramatically different even when viewed in fixed pose. Hence, this is another challenge for face recognition, and Extended Yale-B dataset are extensively used for testing appearance-based face recognition methods. Similar to the UMIST dataset, the images are also cropped and resized to 32×32 pixels. This dataset now has around 64 near frontal images under different illuminations per individual. The MIT-CBCL dataset provides 3240 synthetic images rendered from 3D head models of 10 peoples. The head models are generated by fitting a morphable model to the high-resolution training images. Different from UMNIST dataset, the MIT-CBCL dataset is based on the 3D morphable model, which is rendered under varying pose and illumination conditions making the face recognition task more challenging. The size of each image is originally 200×200 with 256 gray levels per pixel. In our simulation, we down-sample the size of each image to 32×32 and no other preprocessing is performed. The detailed information of dataset and some sampled images of real-world datasets can be shown in Table 3 and Figure 7. For each dataset, we randomly select 10, 50 and 30 samples from each class as training samples for UMNIST, Extended Yale-B, and MIT-CBCL datasets. The test set is then formed by the selected or all remaining samples. The data partitioning for each dataset is also given in Table 3.

Next, we compare our method with other supervised and semisupervised dimension reduction methods. These methods include Regularized Linear discriminant analysis (RLDA), SDA
[2], Lap-RLS/L [1], least-square solution for solving SDA in Eq. (16) (in Table 1, we refer to it as LS-SDA) [28], FME [7, 10], and the proposed LGR. Note that Principal Component Analysis (PCA) is an unsupervised method while RLDA is supervised methods, and the remaining methods LGR are all semisupervised methods. The simulation settings are as follows: for SDA, Lap-RLS/L, two parameters, i.e., $\alpha_t$ and $\alpha_m$, need to be determined for balancing the trade-off between the manifold and Tikhonov terms. We use fivefold cross validation to determine the best values and the candidate set is $\{10^{-9}, 10^{-6}, 10^{-3}, 10^{0}, 10^{3}, 10^{6}, 10^{9}\}$. The above candidate set is also used for determining the best value for the Tikhonov term parameter $\alpha_t$ in RLDA and the addition regularized parameter $\alpha_r$ in FME and LGR. In order to eliminate the null space before performing dimension reduction, the training sets in all datasets are preliminarily processed with PCA operator. Since most of methods, such as RLDA, SDA, Lap-RLS/L and FME, and the proposed LGR have a limited rank of $c-1$, we simply reduce the dimensionality of all methods to $c-1$. All methods used labeled set in the output reduced subspace to train a nearest neighborhood classifier in order to evaluate the classification accuracy of test set. We also compare the performance of nearest neighborhood classifier with other state-of-the-art methods as a baseline.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Database Type</th>
<th>#Samples</th>
<th>#Dim</th>
<th>#Class</th>
<th>#Training per Class</th>
<th>#Test per Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>UMNIST</td>
<td>Face</td>
<td>1012</td>
<td>1024</td>
<td>20</td>
<td>20</td>
<td>Remains</td>
</tr>
<tr>
<td>Extended Yale-B</td>
<td>Face</td>
<td>16123</td>
<td>1024</td>
<td>38</td>
<td>50</td>
<td>Remains</td>
</tr>
<tr>
<td>MIT-CBCL</td>
<td>Face</td>
<td>3240</td>
<td>1024</td>
<td>10</td>
<td>30</td>
<td>30</td>
</tr>
</tbody>
</table>

Table 3. Dataset information and data partition for each dataset.

(a) ![Image](a)  
(b) ![Image](b)  
(c) ![Image](c)

Figure 7. Sample images of real-world datasets: (a) UMNIST dataset, (b) Extended Yale-B dataset, (c) MIT-CBCL dataset.

The average accuracies over 20 random splits with the above parameters for each dataset are shown in Table 4. From the simulation results, we can obtain the following observation: (1) given sufficient labeled samples, all the supervised and semisupervised dimension reduction methods outperform nearest neighborhood classifier due to the utilization of label information
and feature extraction; (2) the semisupervised dimension reduction methods are better than the corresponding supervised methods. For example, SDA outperforms RLDA by about 5–6% in COIL100 dataset with two labeled samples per class. For other datasets, it can outperform by 2–3%. This indicates that by incorporating the unlabeled set into the training procedure, the classification performance can be markedly improved, as the manifold structure embedded in the dataset is preserved; (3) we also observe that both SDA and the least-square solution in Table 1 can achieve the same classification results due to the reason as analyzed in Section 3; (4) the proposed LGR can deliver better accuracies than those delivered by other semisupervised dimension reduction methods such as SDA and Lap-RLS/L by about 3–4% in most datasets. The improvement can even achieve almost 8% in ETH80 dataset with two labeled samples per class. The improvement is believed to be true that LGR aims to characterize both local and global discriminative information embedded in dataset, which is better to handle classification problem; (5) we observe that LGR outperform FME by about 2% in most cases. The main reason is that LGR has utilized a weighted normalized local discriminative Laplacian matrix to preserve both manifold and discriminative structures in dataset, which is better than only relying on neighborhood graph.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Method</th>
<th>4 labeled samples per class</th>
<th>7 labeled samples per class</th>
<th>10 labeled samples per class</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Unlabeled</td>
<td>Test</td>
<td>Unlabeled</td>
<td>Test</td>
</tr>
<tr>
<td>UMNIST</td>
<td>Baseline</td>
<td>81.1±0.9</td>
<td>80.2±1.0</td>
<td>88.6±0.7</td>
</tr>
<tr>
<td></td>
<td>RLDA</td>
<td>85.2±0.6</td>
<td>85.0±0.7</td>
<td>90.7±0.5</td>
</tr>
<tr>
<td></td>
<td>SDA</td>
<td>86.4±0.7</td>
<td>86.3±0.7</td>
<td>92.1±0.6</td>
</tr>
<tr>
<td></td>
<td>LS-SDA</td>
<td>86.4±0.7</td>
<td>86.3±0.7</td>
<td>92.1±0.6</td>
</tr>
<tr>
<td></td>
<td>Lap-RLS/L</td>
<td>86.6±0.7</td>
<td>86.0±0.8</td>
<td>91.9±0.3</td>
</tr>
<tr>
<td></td>
<td>FME</td>
<td>88.2±0.6</td>
<td>87.7±0.6</td>
<td>93.1±0.3</td>
</tr>
<tr>
<td></td>
<td>LGR</td>
<td>89.2±0.4</td>
<td>88.9±0.5</td>
<td>94.2±0.2</td>
</tr>
</tbody>
</table>

Table 4. Average classification accuracy over 20 random splits on unlabeled set and test set of different datasets (means±standard derivations).

4. Conclusion

In this chapter, we propose a semisupervised method, namely LGR, for face recognition. With the above analysis, the following conclusions can be drawn: (1) the proposed LGR can achieve better results in face recognition than those delivered by other state-of-the-art methods as more discriminative information are captured based on local and global regressions, (2) the proposed LGR is robust to outliers and can handle the imbalanced data, and (3) the proposed LGR can deal with out-of-sample extrapolation to estimate the labels of new-coming face data by casting it to the global projection matrix.
Appendix

In order to prove that $L_d$ is graph Laplacian matrix, we need to prove $L_d$ is positive semidefinite matrix and the sum of each row or column of $L_d$ is equal to zero. We first have the following Lemmas:

**Lemma 1.** For each local patch $X_j$, $L_j$ can be reformulated as follows:

$$L_j = \eta G_j \left( G_j^T X_j G_j + \eta I \right)^{-1} G_j^T,$$

where $G_j = (I - \Delta_j e_k^T (e_k \Delta_j e_k^T)) \Delta_j^{-1/2} \in \mathbb{R}^{k \times k}$.

**Proof.** First, it can be easily noted that $G_j G_j^T = H_j$, which is verified as follows:

$$G_j G_j^T = \left( I - \Delta_j e_k^T (e_k \Delta_j e_k^T) \right) \Delta_j \left( I - \Delta_j e_k^T (e_k \Delta_j e_k^T) \right)^T$$

$$= \left( \Delta_j - \Delta_j e_k^T (e_k \Delta_j e_k^T) \right) \left( I - \Delta_j e_k^T (e_k \Delta_j e_k^T) \right)^T$$

$$= \Delta_j - 2 \Delta_j e_k^T (e_k \Delta_j e_k^T) + \Delta_j \Delta_j - \Delta_j e_k^T (e_k \Delta_j e_k^T)^2$$

$$= \Delta_j = H_j$$

Then, we have

$$L_j = H_j - H_j X_j^T (X_j G_j^T X_j + \eta I)^{-1} X_j H_j$$

$$= G_j G_j^T - G_j G_j^T X_j^T (X_j G_j G_j^T X_j + \eta I)^{-1} X_j G_j G_j^T$$

$$= G_j G_j^T - G_j G_j^T X_j^T G_j^T X_j G_j + \eta I)^{-1} G_j^T$$

$$= G_j G_j^T - G_j \left( G_j^T X_j^T X_j G_j + \eta I \right)^{-1} G_j^T$$

$$= \eta G_j \left( G_j^T X_j^T X_j G_j + \eta I \right)^{-1} G_j^T$$

The second equation holds as $A(A^T A + \eta I)^{-1} = (A A^T + \eta I)^{-1} A$, for any matrix $A$. Thus, Lemma 1 is proved.

**Lemma 2.** Given a positive semidefinite matrix $C$, $D C D^T$ is a positive semidefinite matrix for any matrix $D$.  


Lemma 3. Given a set of positive semidefinite matrixes \(\{C_1, C_2, \ldots, C_n\}\) then \(\sum_{j=1}^{n} C_j\) is a positive semidefinite matrix.

We neglect the proofs of Lemmas 2 and 3 as they can be seen in reference [15]. Then with Lemmas 1–3, we can easily prove Theorem 2 as follows:

**Proof of Theorem 2.** Note that following Lemma 1, we reformulate each \(L_j\) as 
\[ L_j = \eta G_j \left( G_j^T X_j X_j G_j + \eta I \right)^{-1} G_j^T. \]
It can be noted \(\left( G_j^T X_j X_j G_j + \eta I \right)^{-1}\) is a positive semidefinite matrix, then, following Lemmas 2 and 3, we have each \(\eta S_j G_j \left( G_j^T X_j X_j G_j + \eta I \right)^{-1} G_j^T S_j^T\) is a positive semidefinite matrix and \(L_d\), i.e.,
\[ L_d = \sum_{j=1}^{l+u} \left( S_j L_j S_j^T \right) = \sum_{j=1}^{l+u} \left( \eta S_j G_j \left( G_j^T X_j X_j G_j + \eta I \right)^{-1} G_j^T S_j^T \right). \]

is also a positive semidefinite matrix. In addition, for each \(\eta S_j G_j \left( G_j^T X_j X_j G_j + \eta I \right)^{-1} G_j^T S_j^T\), we have 
\[ S_j^T e^T = e_k^T \quad \text{and} \quad G_j^T e_k^T = \left( e_k^T - \left( e_k^T \Delta_j e_k \right) \right) \left( e_k^T \Delta_j e_k \right)^{-1/2} = 0 \]
\[ \Rightarrow \eta S_j G_j \left( G_j^T X_j X_j G_j + \eta I \right)^{-1} G_j^T S_j^T e^T = 0 \]
\[ \Rightarrow L_d e^T = \sum_{j=1}^{l+u} \eta S_j G_j \left( G_j^T X_j X_j G_j + \eta I \right)^{-1} G_j^T S_j^T e^T = 0 \]

which indicates that the sum of each row or column of \(L_d\) is equal to zero. We thus prove \(L_d\) is graph Laplacian matrix.

**Author details**

Mingbo Zhao\(^{1*}\), Yuan Gao\(^{1*}\), Zhao Zhang\(^{2}\) and Bing Li\(^{3}\)

*Address all correspondence to: mbzhao4@gmail.com

*Address all correspondence to: ethan.y.gao@my.cityu.edu.hk

1 Department of Electronic Engineering, City University of Hong Kong, Kowloon, Hong Kong S. A. R.

2 School of Computer Science and Technology, Soochow University, Suzhou, P. R. China

3 School of Economics, Wuhan University of Technology, Wuhan, P. R. China
References


