Heteroscedastic Probabilistic Linear Discriminant Analysis with Semi-Supervised Extension

Yu Zhang and Dit-Yan Yeung

Hong Kong University of Science and Technology
{zhangyu,dyyeung}@cse.ust.hk

Abstract. *Linear discriminant analysis* (LDA) is a commonly used method for dimensionality reduction. Despite its successes, it has limitations under some situations, including the small sample size problem, the homoscedasticity assumption that different classes have the same Gaussian distribution, and its inability to produce probabilistic output and handle missing data. In this paper, we propose a semi-supervised and heteroscedastic extension of probabilistic LDA, called S^{2}HPLDA, which aims at overcoming all these limitations under a common principled framework. Moreover, we apply automatic relevance determination to determine the required dimensionality of the low-dimensional space for dimensionality reduction. We empirically compare our method with several related probabilistic subspace methods on some face and object databases. Very promising results are obtained from the experiments showing the effectiveness of our proposed method.

1 Introduction

The need for dimensionality reduction is pervasive in many applications of pattern recognition and machine learning due to the high dimensionality of the data involved. Dimensionality reduction techniques seek to project high-dimensional data either linearly or nonlinearly into a lower-dimensional space according to some criterion so as to facilitate subsequent processing, such as classification. Classical linear dimensionality reduction methods include principal component analysis (PCA) [1] and *linear discriminant analysis* (LDA) [2], with the former being an unsupervised technique while the latter a supervised one that exploits the label information in the labeled data. For classification applications, LDA generally outperforms PCA because label information is usually useful for finding a projection to improve class separability in the lower-dimensional space.

Although LDA is widely used in many applications, the method in its original form does have limitations under some situations. One of them is a well-known limitation often referred to as the small sample size (SSS) problem [3], which arises in applications when the sample size is much smaller than the feature dimensionality and hence the within-class scatter matrix is singular. A number of methods have been proposed to address this problem, e.g., PseudoLDA [4], PCA+LDA [5], LDA/QR [6], NullLDA [3], DCV [7], DualLDA [8] and 2DLDA [9]. The main idea underlying these methods is to seek a space or subspace in which the within-class scatter matrix is nonsingular and then perform LDA or its variants there without suffering from the singularity problem. More recently, another approach has been pursued by some researchers [10–12] to alleviate the SSS problem via *semi-supervised learning* [13], by utilizing unlabeled data in performing dimensionality reduction in addition to labeled data. Another limitation of LDA arises from the fact that the solution it gives is optimal only when the classes are homoscedastic with the same Gaussian distribution. However, this requirement is too rigid in practice and hence it does not hold in many real-world applications. To overcome this limitation, mixture discriminant analysis [14] and a maximum likelihood approach [15] have been proposed. Recently, Loog and Duin [16] proposed a heteroscedastic extension to LDA based on the Chernoff criterion, with a kernel extension proposed later in [17]. The third limitation of LDA comes from its non-probabilistic nature. As such, it cannot produce probabilistic output and handle missing data in a principled manner. While producing probabilistic output can help the subsequent decision-making process in incorporating uncertainty under a probabilistic framework, the missing data problem is so commonly encountered in applications that being able to deal with it is very essential to the success of pattern recognition tools for practical applications. Some probabilistic LDA models have been proposed, e.g., [18–20]. A by-product of most probabilistic LDA models except the one in [18] is that it imposes no restriction on the maximum number of reduced dimensions, but the original LDA model can only project data into at most $C - 1$ dimensions where C is the number of classes. Nevertheless, previous research in probabilistic LDA [19, 20] did not pay much attention to the issue of how to determine the reduced dimensionality needed.

While various attempts were made previously to address the above limitations individually, mostly one or at most two at a time, we are more aggressive here in trying to address all of them within a common principled framework. Specifically, in this paper, we will go through a two-step process in our presentation. First, we propose a *heteroscedastic probabilistic LDA* (HPLDA) model which relaxes the homoscedasticity assumption in LDA. However, in HPLDA, the parameters for each class can only be estimated using labeled data from that class. This may lead to poor performance when labeled data are scarce. Motivated by previous attempts that applied semi-supervised learning to alleviate the SSS problem, we then extend HPLDA to *semi-supervised heteroscedastic probabilistic LDA* (S^{2}HPLDA) by making use of (usually large quantities of) unlabeled data in the learning process. In S^{2}HPLDA, each class can have a different class covariance matrix and unlabeled data are modeled by a Gaussian mixture model in which each mixture component corresponds to one class. We also use automatic relevance determination (ARD) [21] to determine the required dimensionality of the lower-dimensional space which can be different for different classes and hence is fairly flexible.

The remainder of this paper is organized as follows. In Section 2, we first briefly review some previous work on probabilistic LDA. We then present HPLDA in Section 3 and S^{2}HPLDA in Section 4. Section 5 reports some experimental results based on face and object databases to demonstrate the effectiveness of our proposed method. Finally, Section 6 concludes the paper.

2 Related Work

To the best of our knowledge, three variants of probabilistic LDA [18–20] were proposed before.

In [18], each class is modeled by a Gaussian distribution with a common covariance matrix shared by all classes and the mean vectors of different classes are modeled by another Gaussian distribution whose covariance matrix is similar to the between-class scatter matrix in LDA. The solution of this probabilistic LDA model is so similar to that of LDA that it, unfortunately, also inherits some limitations of LDA. For example, it needs probabilistic PCA (PPCA) to perform (unsupervised) dimensionality reduction first to alleviate the SSS problem and it can only project data to $(C-1)$ dimensions.

Yu et al. [19] proposed a supervised extension of probabilistic PCA (PPCA) [22] called SPPCA. This approach can be viewed as first concatenating each data point with its class indicator vector and then applying PPCA to this extended form. From the analysis of [23], the maximum likelihood solution of this approach is identical to that of LDA. Yu et al. [19] also proposed a semi-supervised extension of SPPCA, called S^{2}PPCA, which can utilize unlabeled data as well.

The model in [20] is slightly different from others. It directly models the between-class and within-class variances. So each data point can be described as the aggregation of three parts: the common mean which is the mean of the whole dataset, the between-class variance which describes the characteristics of different classes, and the within-class variance which describes the characteristics of each data point. Prince and Elder [20] also gave some extensions of this model useful for face recognition.

3 HPLDA: Heteroscedastic Probabilistic Linear Discriminant Analysis

Suppose the whole dataset contains l labeled data points $\{(\mathbf{x}_i, y_i)\}_{i=1}^{l}$ from C classes $\boldsymbol{\Pi}_k$ $(k = 1, \ldots, C)$, where $\mathbf{x}_i \in \mathbb{R}^D$ with its label $y_i \in \{1, \ldots, C\}$ and class $\boldsymbol{\Pi}_k$ contains n_k examples. Moreover, all data points $\{\mathbf{x}_i\}_{i=1}^{l}$ are independent and identically distributed.

HPLDA is a latent variable model. It can be defined as follows:

$$\begin{aligned} \mathbf{x}_i &= \mathbf{W}_{y_i}\mathbf{t}_i + \boldsymbol{\mu}_{y_i} + \boldsymbol{\varepsilon}_i \\ \mathbf{t}_i &\sim \mathcal{N}(0, \mathbf{I}_d) \\ \boldsymbol{\varepsilon}_i &\sim \mathcal{N}(0, \tau_{y_i}^{-1}\mathbf{I}_D), \end{aligned} \tag{1}$$

where τ_i specifies the noise level of the ith class, $\mathbf{t}_i \in \mathbb{R}^d$ with $d < D$, $\mathbf{I}_D$ is the $D \times D$ identity matrix and $\mathcal{N}(\mathbf{m}, \boldsymbol{\Sigma})$ denotes a multivariate Gaussian distribution with mean $\mathbf{m}$ and covariance matrix $\boldsymbol{\Sigma}$. So for each class $\boldsymbol{\Pi}_k$, we have a different $\mathbf{W}_k$. This is different from the models proposed in [18–20] in which different classes share the same matrix $\mathbf{W}$. The graphical model for HPLDA is shown in Figure 1. From (1), we can get

$$P(\mathbf{x}_i|\mathbf{t}_i) = \mathcal{N}(\mathbf{W}_{y_i}\mathbf{t}_i + \boldsymbol{\mu}_{y_i}, \tau_{y_i}^{-1}\mathbf{I}_D)$$

and

$$P(\mathbf{x}_i) = \mathcal{N}(\boldsymbol{\mu}_{y_i}, \boldsymbol{\Phi}_{y_i}),$$

where $\boldsymbol{\Phi}_k = \mathbf{W}_k\mathbf{W}_k^T + \tau_k^{-1}\mathbf{I}_D$. So the log-likelihood L of the data set can be calculated as

$$L = -\frac{1}{2}\sum_{k=1}^{C}\sum_{y_i=k}\Big[(\mathbf{x}_i - \boldsymbol{\mu}_k)^T\boldsymbol{\Phi}_k^{-1}(\mathbf{x}_i - \boldsymbol{\mu}_k) + D\ln 2\pi + \ln|\boldsymbol{\Phi}_{\boldsymbol{k}}|\Big], \qquad (2)$$

where $|\mathbf{A}|$ denotes the determinant of a square matrix $\mathbf{A}$. We set the derivative of L with respect to $\boldsymbol{\mu}_k$ to 0 to obtain the maximum likelihood estimate of $\boldsymbol{\mu}_k$ as

$$\boldsymbol{\mu}_k = \bar{\mathbf{m}}_k \equiv \frac{1}{n_k}\sum_{y_i=k}\mathbf{x}_i. \qquad (3)$$

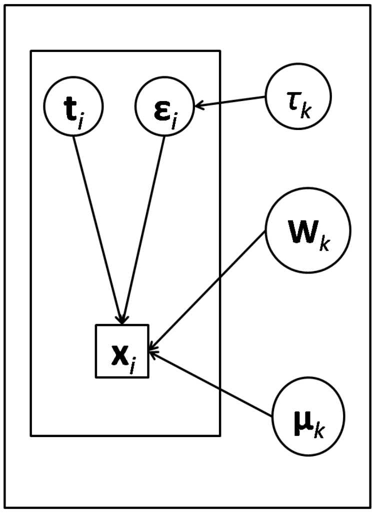

Fig. 1. Graphical model for HPLDA.

Plugging Eq. (3) into (2), the log-likelihood can be simplified as

$$L = -\frac{1}{2}\sum_{k=1}^{C} n_k\Big[\text{tr}(\boldsymbol{\Phi}_k^{-1}\mathbf{S}_k) + D\text{ln}2\pi + \text{ln}|\boldsymbol{\Phi}_k|\Big], \qquad (4)$$

where $\mathbf{S}_k = \frac{1}{n_k}\sum_{y_i=k}(\mathbf{x}_i - \bar{\mathbf{m}}_k)(\mathbf{x}_i - \bar{\mathbf{m}}_k)^T$ is the estimated covariance matrix for the kth class. Since $\mathbf{W}_k$ for different classes are independent, we can estimate each $\mathbf{W}_k$ from the following expression:

$$L_k = -\frac{1}{2}n_k\Big[\text{tr}(\boldsymbol{\Phi}_k^{-1}\mathbf{S}_k) + D\text{ln}2\pi + \text{ln}|\boldsymbol{\Phi}_k|\Big], \qquad (5)$$

which is similar to the log-likelihood in PPCA. So, following the analysis in [22], we can obtain the maximum likelihood estimate of $\mathbf{W}_k$ as the eigenvectors of $\mathbf{S}_k$ corresponding to the largest eigenvalues and τ_k^{-1} is equal to the mean of the discarded eigenvalues.

3.1 Discussion

If all $\mathbf{W}_k$ and τ_k in (1) are the same, denoted by $\mathbf{W}$ and τ, then, from Eq. (4), the log-likelihood can be expressed as

$$L = -\frac{l}{2}\Big[\mathrm{tr}(\boldsymbol{\Phi}^{-1}\mathbf{S}_{\mathbf{w}}) + D\ln 2\pi + \ln|\boldsymbol{\Phi}|\Big], \tag{6}$$

where $\mathbf{S}_{\mathbf{w}} = \frac{1}{l}\sum_{k=1}^{C}\sum_{y_i=k}(\mathbf{x}_i - \bar{\mathbf{m}}_k)(\mathbf{x}_i - \bar{\mathbf{m}}_k)^T$ is the within-class scatter matrix in LDA and $\boldsymbol{\Phi} = \mathbf{W}\mathbf{W}^T + \tau^{-1}\mathbf{I}_D$. So, also following the analysis in [22], $\mathbf{W}$ consists of the top eigenvectors of $\mathbf{S}_{\mathbf{w}}$ and τ^{-1} is equal to the mean of the discarded eigenvalues. Then if the data points are whitened by the total scatter matrix, i.e., the total scatter matrix of the dataset is the identity matrix, the estimated $\mathbf{W}$ is just the solution in traditional LDA.

There are some limitations in our model (1) though. From the above analysis, we can see that $\mathbf{W}_k$ is estimated using the data points from the kth class only. However, in many applications, labeled data are scarce due to the labeling effort required. So, as a result, $\mathbf{W}_k$ may not be estimated very accurately. On the other hand, unlabeled data are often available in large quantities at very low cost. It would be desirable if we can also make use of the unlabeled data in the estimation of $\mathbf{W}_k$. Moreover, the dimensionality of $\mathbf{W}_k$ plays an important role in the performance of our model and it should preferably be determined automatically. In the next section, we will discuss how to solve these two problems together.

4 S^{2}HPLDA: Semi-Supervised Heteroscedastic Probabilistic Linear Discriminant Analysis

As in HPLDA, there are l labeled data points $\{(\mathbf{x}_i, y_i)\}_{i=1}^{l}$ from C classes. In addition, there are u unlabeled data points $\{\mathbf{x}_{l+1}, \ldots, \mathbf{x}_{l+u}\}$, with $n = l + u$. Each class $\boldsymbol{\Pi}_k$ contains n_k labeled examples. For the labeled data points, we still use (1) to model them. For the unlabeled data points, we model them using a mixture model in which each mixture component follows (1) with prior probability $p(\boldsymbol{\Pi}_k) = \pi_k$. Thus the new model can be defined as:

$$\begin{aligned}
\mathbf{x}_i &= \mathbf{W}_{y_i}\mathbf{t}_i + \boldsymbol{\mu}_{y_i} + \boldsymbol{\varepsilon}_i, \ \text{for } i \le l \\
\mathbf{t}_i &\sim \mathcal{N}(0, \mathbf{I}_d) \\
\boldsymbol{\varepsilon}_i &\sim \mathcal{N}(0, \tau_{y_i}^{-1}\mathbf{I}_D) \\
p(\mathbf{x}_i) &= \sum_{k=1}^{C}\pi_k p(\mathbf{x}_i|\boldsymbol{\Pi}_k), \ \text{for } i > l,
\end{aligned} \tag{7}$$

where $\mathbf{t}_i \in \mathbb{R}^d$. Moreover, we use the ARD method [21] to determinate the dimensionality of $\mathbf{W}_k$ by introducing a data-dependent prior distribution

$$p(\mathbf{W}_{k,j}) \sim \mathcal{N}(0, \nu_{kj}^{-1}\mathbf{X}\mathbf{L}\mathbf{X}^T),$$

where $\mathbf{W}_{k,j}$ is the jth column of $\mathbf{W}_k$, $\mathbf{X} \in \mathbb{R}^{D\times n}$ is the total data matrix including both labeled and unlabeled data, and $\mathbf{L}$, whose construction will be described later, is the graph Laplacian matrix defined on $\mathbf{X}$. The graphical model is shown in Figure 2. Using the data-dependent prior on $\mathbf{W}_{k,j}$, we are essentially adopting the manifold assumption, which has been widely used in dimensionality reduction [24] and semi-supervised learning [25]. More specifically, if two points are close with respect to the intrinsic geometry of the underlying manifold, they should remain close in the embedding space after dimensionality reduction. The parameter ν_{kj} can be viewed as an indicator of the importance of the corresponding dimension of $\mathbf{W}_k$ to determine whether that dimension should be kept.

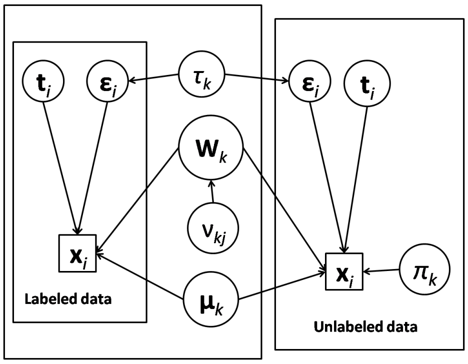

Fig. 2. Graphical model for S^{2}HPLDA.

We now describe the construction of $\mathbf{L}$. Given the dataset $\mathcal{D} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, we first construct a K nearest neighbor graph $G = (V, E)$, with the vertex set $V = \{1, \ldots, n\}$ corresponding to the labeled and unlabeled data points and the edge set $E \subseteq V \times V$ representing the relationships between data points. Each edge is assigned a weight r_{ij} which reflects the similarity between points $\mathbf{x}_i$ and $\mathbf{x}_j$:

$$r_{ij} = \begin{cases} \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{\sigma_i\sigma_j}\right) & \text{if } \mathbf{x}_i \in N_K(\mathbf{x}_j) \text{ or } \mathbf{x}_j \in N_K(\mathbf{x}_i) \\ 0 & \text{otherwise} \end{cases}$$

where $N_K(\mathbf{x}_i)$ denotes the neighborhood set of the K-nearest neighbors of $\mathbf{x}_i$, σ_i the distance between $\mathbf{x}_i$ and its Kth nearest neighbor, and σ_j the distance between $\mathbf{x}_j$ and its Kth nearest neighbor. This way of constructing the nearest neighbor graph is called *local scaling* [26]. Then $\mathbf{G}$ is the similarity graph with its (i, j)th element being r_{ij}, $\mathbf{D}$ is a diagonal matrix whose entries are the column sums of $\mathbf{G}$, and $\mathbf{L} = \mathbf{D} - \mathbf{G}$.

Model (7) has parameters $\{\boldsymbol{\mu}_k\}$, $\{\tau_k\}$, $\{\pi_k\}$, $\{\nu_{kj}\}$, $\{\mathbf{W}_k\}$. We use the expectation maximization (EM) algorithm [27] to estimate them from data. Here we introduce $\mathbf{z}_i$ as a hidden indicator vector for each unlabeled data point $\mathbf{x}_i$, with z_{ik} being 1 if $\mathbf{x}_i$ belongs to the kth class. Since the number of parameters in this model is quite large, we apply two-fold EM [28] here to speed up convergence. In the E-step of the outer-fold EM, $\{\mathbf{z}_i\}$ are the hidden variables. We estimate $p(z_{ik} = 1)$ as:

$$p(z_{ik} = 1) = p(\boldsymbol{\Pi}_k | \mathbf{x}_i) = \frac{\pi_k p(\mathbf{x}_i | \boldsymbol{\Pi}_k)}{\sum_{j=1}^{C} \pi_j p(\mathbf{x}_i | \boldsymbol{\Pi}_j)}$$

where

$$\begin{aligned} p(\mathbf{x}_i | \boldsymbol{\Pi}_k) &= \int p(\mathbf{x}_i | \mathbf{t}_i, \mathbf{W}_k, \boldsymbol{\mu}_k, \tau_k) p(\mathbf{t}_i) d\mathbf{t}_i \\ &= \mathcal{N}(\mathbf{x}_i | \boldsymbol{\mu}_k, \mathbf{W}_k \mathbf{W}_k^T + \tau_k^{-1} \mathbf{I}_D). \end{aligned}$$

In the M-step of the outer-fold EM, we aim to estimate $\{\pi_k\}$ and $\{\boldsymbol{\mu}_k\}$. The complete-data log-likelihood is defined as

$$L_C = \sum_{i=1}^{l} \ln p(\mathbf{x}_i | \boldsymbol{\Pi}_{y_i}) + \sum_{i=l+1}^{n} \sum_{k=1}^{C} z_{ik} \Big\{ \ln \big[\pi_k p(\mathbf{x}_i | \boldsymbol{\Pi}_k) \big] \Big\}.$$

So the expectation of the complete-data log-likelihood in the M-step of the outer-fold EM can be calculated as

$$\begin{aligned} &\langle L_C \rangle = \\ &\sum_{k=1}^{C} \sum_{y_i = k} \left\{ -\frac{1}{2} (\mathbf{x}_i - \boldsymbol{\mu}_k)^T \boldsymbol{\Phi}_k^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_k) - \frac{1}{2} \ln |\boldsymbol{\Phi}_k| - \frac{D}{2} \ln 2\pi \right\} + \\ &\sum_{k=1}^{C} \sum_{i=l+1}^{n} \langle z_{ik} \rangle \left\{ -\frac{1}{2} (\mathbf{x}_i - \boldsymbol{\mu}_k)^T \boldsymbol{\Phi}_k^{-1} (\mathbf{x}_i - \boldsymbol{\mu}_k) - \frac{1}{2} \ln |\boldsymbol{\Phi}_k| + \ln \pi_k - \frac{D}{2} \ln 2\pi \right\}, \end{aligned}$$

where $\boldsymbol{\Phi}_k = \mathbf{W}_k \mathbf{W}_k^T + \tau_k^{-1} \mathbf{I}_D$. We maximize the expectation of the complete-data log-likelihood with respect to $\{\pi_i\}$ and $\{\boldsymbol{\mu}_i\}$. The update rules are given by

$$\begin{aligned} \tilde{\pi}_k &= \frac{\sum_{i=l+1}^{n} \langle z_{ik} \rangle}{\sum_{i=l+1}^{n} \sum_{k=1}^{C} \langle z_{ik} \rangle} = \frac{1}{u} \sum_{i=l+1}^{n} \langle z_{ik} \rangle \\ \tilde{\boldsymbol{\mu}}_k &= \frac{\sum_{y_i = k} \mathbf{x}_i + \sum_{i=l+1}^{n} \langle z_{ik} \rangle \mathbf{x}_i}{n_k + \sum_{i=l+1}^{n} \langle z_{ik} \rangle}, \end{aligned}$$

where $\langle \cdot \rangle$ denotes the expectation of a variable.

In the E-step of the inner-fold EM, $\{\mathbf{t}_i\}$ are the hidden variables. We estimate

$$\begin{aligned} p(\mathbf{t}_i | \mathbf{x}_i, \boldsymbol{\Pi}_k) &= \mathcal{N}(\mathbf{t}_i | \boldsymbol{\Sigma}_k^{-1} \mathbf{W}_k^T (\mathbf{x}_i - \tilde{\boldsymbol{\mu}}_k), \tau_k^{-1} \boldsymbol{\Sigma}_k^{-1}) \\ \langle \mathbf{t}_i | \boldsymbol{\Pi}_k \rangle &= \boldsymbol{\Sigma}_k^{-1} \mathbf{W}_k^T (\mathbf{x}_i - \tilde{\boldsymbol{\mu}}_k) \\ \langle \mathbf{t}_i \mathbf{t}_i^T | \boldsymbol{\Pi}_k \rangle &= \tau_k^{-1} \boldsymbol{\Sigma}_k^{-1} + \langle \mathbf{t}_i | \boldsymbol{\Pi}_k \rangle \langle \mathbf{t}_i | \boldsymbol{\Pi}_k \rangle^T, \end{aligned}$$

with $\boldsymbol{\Sigma}_k = \tau_k^{-1}\mathbf{I}_d + \mathbf{W}_k^T\mathbf{W}_k$.

In the M-step of the inner-fold EM, we aim to estimate $\{\mathbf{W}_k\}$, $\{\nu_{kj}\}$ and $\{\tau_k\}$. The complete-data log-likelihood can be calculated as

$$\tilde{L}_C = \sum_{i=1}^{l} \ln p(\mathbf{x}_i, \mathbf{t}_i|\boldsymbol{\Pi}_{y_i}) + \sum_{k=1}^{C}\sum_{j=1}^{d} p(\mathbf{W}_{k,j}) + \sum_{i=l+1}^{n}\sum_{k=1}^{C}\langle z_{ik}\rangle \ln\{\pi_k p(\mathbf{x}_i, \mathbf{t}_i|\boldsymbol{\Pi}_k)\}.$$

The expectation of the complete-data log-likelihood can be calculated as

$$\begin{aligned}
&\langle\tilde{L}_C\rangle = \\
&\sum_{i=1}^{C}\sum_{y_j=i}\left\{\frac{D}{2}\ln\tau_i - \frac{1}{2}\text{tr}(\langle\mathbf{t}_j\mathbf{t}_j^T|C_i\rangle) - \frac{\tau_i}{2}\|\mathbf{x}_j - \tilde{\boldsymbol{\mu}}_i\|^2 + \right.\\
&\qquad\qquad \left.\tau_i\langle\mathbf{t}_j|C_i\rangle^T\mathbf{W}_i^T(\mathbf{x}_j - \tilde{\boldsymbol{\mu}}_i) - \frac{\tau_i}{2}\text{tr}(\mathbf{W}_i^T\mathbf{W}_i\langle\mathbf{t}_j\mathbf{t}_j^T|C_i\rangle)\right\} + \\
&\sum_{i=1}^{C}\sum_{j=l+1}^{n}\langle z_{ji}\rangle\left\{\frac{D}{2}\ln\tau_i - \frac{1}{2}\text{tr}(\langle\mathbf{t}_j\mathbf{t}_j^T|C_i\rangle) - \frac{\tau_i}{2}\|\mathbf{x}_j - \tilde{\boldsymbol{\mu}}_i\|^2 + \right.\\
&\qquad\qquad \left.\tau_i\langle\mathbf{t}_j|C_i\rangle^T\mathbf{W}_i^T(\mathbf{x}_j - \tilde{\boldsymbol{\mu}}_i) - \frac{\tau_i}{2}\text{tr}(\mathbf{W}_i^T\mathbf{W}_i\langle\mathbf{t}_j\mathbf{t}_j^T|C_i\rangle) + \ln\tilde{\boldsymbol{\pi}}_i\right\} + \\
&\sum_{i=1}^{C}\sum_{j=1}^{d}\left\{\frac{D}{2}\ln\nu_{ij} - \frac{1}{2}\nu_{ij}\mathbf{W}_{i,j}^T\mathbf{L}_\star^{-1}\mathbf{W}_{i,j}\right\},
\end{aligned}$$

where $\mathbf{L}_\star = \mathbf{XLX}^T$. Maximization of the expected complete-data log-likelihood with respect to $\mathbf{W}_k$, τ_k and ν_{kj} gives the following update rules:

$$\begin{aligned}
\tilde{\mathbf{W}}_k &= (\tau_k\mathbf{S}_k\mathbf{W}_k - \mathbf{L}_\star^{-1}\mathbf{W}_k\boldsymbol{\Lambda}_k\boldsymbol{\Sigma}_k)(\tilde{n}_k\mathbf{I}_d + \tau_k\boldsymbol{\Sigma}_k^{-1}\mathbf{W}_k^T\mathbf{S}_k\mathbf{W}_k)^{-1}\\
\tilde{\tau}_k &= \frac{D\tilde{n}_k}{\text{tr}\left\{(\mathbf{I}_D - \mathbf{W}_k\boldsymbol{\Sigma}_k^{-1}\mathbf{W}_k^T)^2\mathbf{S}_k + \tilde{n}_k\tau_k^{-1}\mathbf{W}_k\boldsymbol{\Sigma}_k^{-1}\mathbf{W}_k^T\right\}}\\
\tilde{\nu}_{kj} &= \frac{D}{\tilde{\mathbf{W}}_{k,j}^T\mathbf{L}_\star^{-1}\tilde{\mathbf{W}}_{k,j}},
\end{aligned}$$

where $\mathbf{L}_\star = \mathbf{XLX}^T$, $\boldsymbol{\Lambda}_k = \text{diag}(\nu_{k1}, \dots, \nu_{kM})$ is a diagonal matrix with the (j,j)th element being ν_{kj}, $\tilde{n}_k = n_k + u\tilde{\pi}_k$, and $\mathbf{S}_k = \sum_{y_i=k}(\mathbf{x}_i - \tilde{\boldsymbol{\mu}}_k)(\mathbf{x}_i - \tilde{\boldsymbol{\mu}}_k)^T + \sum_{i=l+1}^{n}\langle z_{ik}\rangle(\mathbf{x}_i - \tilde{\boldsymbol{\mu}}_k)(\mathbf{x}_i - \tilde{\boldsymbol{\mu}}_k)^T$.

After estimating the parameters, we can use ν_{kj} to determinate the dimensionality of $\mathbf{W}_k$. We can set a threshold η and discard the $\mathbf{W}_{k,j}$ whose corresponding ν_{kj} is larger than η. In our experiments, we set η to be 10000.

For a test data point $\mathbf{x}_{\text{test}}$, we classify it to class $\boldsymbol{\Pi}_k$ where $k = \arg\max_j p(\boldsymbol{\Pi}_j|\mathbf{x}_{\text{test}})$.

4.1 Discussion

Our S^{2}HPLDA model has advantages over existing probabilistic subspace methods. In our method, each class is modeled by a Gaussian distribution with a possibly different covariance matrix, giving our model higher expressive power than existing methods.

Moreover, our model, being a semi-supervised method, can utilize unlabeled data but most other probabilistic LDA models cannot, except S^{2}PPCA.

There exist several variants of LDA [10–12] which also utilize unlabeled data to alleviate the SSS problem. Cai et al. [10] and Zhang and Yeung [11] used unlabeled data to define a regularization term to incorporate the manifold and cluster assumptions, which are two widely adopted assumptions in semi-supervised learning. Zhang and Yeung [12] used unlabeled data to maximize the criterion of LDA and estimate the labels simultaneously, in a way similar to the idea behind transductive SVM (TSVM) [29, 30]. Unlike these methods, our method works in a different way. We use a Gaussian mixture model to model the unlabeled data with each component corresponding to one class. From previous research in semi-supervised learning, unlabeled data are more suitable for generative models since unlabeled data can help to estimate the data density [13] and our method also follows this strategy.

According to [31], integrating out all parameters is better than performing point estimation in terms of the generalization performance. In our future research, we plan to propose a fully Bayesian extension of S^{2}HPLDA by placing priors on the parameters of S^{2}HPLDA. For example, we can add a Dirichlet prior to $(\pi_1, \ldots, \pi_C)$, a Gaussian prior to $\boldsymbol{\mu}_k$, and Gamma priors to τ_k and ν_{kj}:

$$\begin{aligned}(\pi_1, \ldots, \pi_C) &\sim \text{Dir}(\alpha_0, \ldots, \alpha_0)\\ \boldsymbol{\mu}_k &\sim \mathcal{N}(\boldsymbol{\mu}_0, \beta_0 \mathbf{I}_D)\\ \tau_k &\sim \text{Gamma}(a_0, b_0)\\ \nu_{kj} &\sim \text{Gamma}(c_0, d_0).\end{aligned}$$

Since direct inference is intractable, we may resort to the variational approximation approach [32].

5 Experiments

In this section, we report experimental results based on two face databases and one object database to evaluate the performance of our method and compare it with some related probabilistic subspace methods.

5.1 Experimental Setup

Subspace methods are widely used in face recognition and object recognition applications. Previous research found that face and object images usually lie in a low-dimensional subspace of the ambient image space. Eigenface [33] (based on PCA) and Fisherface [5] (based on LDA) are two representative subspace methods. Many variants have also been proposed in recent years. These subspace methods use different dimensionality reduction techniques to obtain a low-dimensional subspace and then perform classification in the subspace using some classifier. Some researchers also proposed probabilistic versions of these subspace methods, with PPCA [22] and SPPCA [19] being two popular ones. From the analysis in [22], the maximum likelihood solution to PPCA is identical to that to PCA. Since the models proposed in [19] and [23] are identical, then from the analysis in [23], the maximum likelihood solution to SPPCA is also the same as that to LDA. Moreover, PPCA and SPPCA can deal with missing data using the EM algorithm, but PCA and LDA cannot. In our experiments, we study our method empirically and compare it with several probabilistic subspace methods, including PLDA [20], SPPCA [19] and S^{2}PPCA [19]. Note that PLDA and SPPCA are supervised, but S^{2}PPCA and our method S^{2}HPLDA are semi-supervised in nature. For SPPCA and S^{2}PPCA, we use a simple nearest-neighbor classifier to perform classification after dimensionality reduction.

5.2 Face Recognition

We use the ORL face database [5] for the first experiment. The ORL face database contains 400 face images of 40 persons, each having 10 images. These face images contain significant variations in pose and scale. Some images from the database are shown in Figure 3. We randomly select seven images for each person to form the training set and the rest for the test set. Of the seven images for each person, $p \in \{2, 3\}$ images are randomly selected and labeled while the other images remain unlabeled. We perform 10 random splits and report the average results across the 10 trials. Table 1 reports the error rates of different methods evaluated on the unlabeled training data and the test data separately. For each setting, the lowest classification error is shown in bold. Since S^{2}PPCA exploits the structure of unlabeled data, we can see that its performance is better than PLDA and SPPCA. Moreover, S^{2}HPLDA relaxes the homoscedasticity assumption and so it achieves better performance than its homoscedastic counterpart S^{2}PPCA in our settings. From Table 1, we can see that the performance of PLDA is very bad, probably because it gets trapped in an unsatisfactory local optimum when running the EM algorithm.

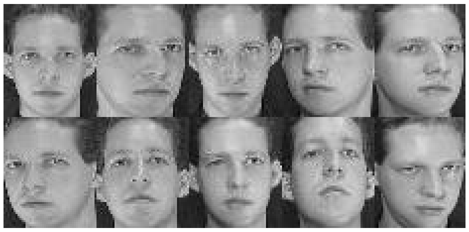

Fig. 3. Some images for one person in the ORL database

The PIE database [34] is used in our second experiment. This database contains 41,368 face images from 68 individuals and these images have large variations in pose, illumination and expression conditions. For our experiments, we select the frontal pose (C27) [1] with varying lighting and illumination conditions and there are about 49 images for each subject. Some images from the database are shown in Figure 4. The experimental setting is almost the same as that of the first experiment. The only difference is that

[1] This face database can be downloaded from `http://www.cs.uiuc.edu/homes/dengcai2/Data/FaceData.html`.

Table 1. Recognition error rates (in mean±std-dev) on ORL for two different p values. 1ST TABLE: $p = 2$; 2ND TABLE: $p = 3$.

Method	Error rate (unlabeled)	Error rate (test)
PLDA	0.7141±0.0803	0.7016±0.0640
SPPCA	0.4562±0.1219	0.4578±0.0710
S^{2}PPCA	0.2703±0.0332	0.2422±0.0366
S^{2}HPLDA	**0.1406±0.0231**	**0.1781±0.0308**

Method	Error rate (unlabeled)	Error rate (test)
PLDA	0.5156±0.0744	0.5042±0.0603
SPPCA	0.4359±0.0713	0.4604±0.0322
S^{2}PPCA	0.2625±0.0595	0.2000±0.0245
S^{2}HPLDA	**0.1375±0.0135**	**0.1562±0.0336**

we use 22 images to form the training set. Of these 22 images, we randomly select $p \in \{3, 4, 5, 6\}$ images and label them, leaving the remaining images unlabeled. Each setting is also repeated 10 times. Table 2 reports the average results over the 10 trials. From the results, we can see that our method again gives the best performance.

Fig. 4. Some images for one person in the PIE database

5.3 Object Recognition

We use the COIL database [35] for our object recognition experiment. This database contains 1,440 grayscale images with black background for 20 objects. For each object, the camera moves around it in pan at intervals of 5 degrees and takes a total of 72 different images. These objects exhibit a wide variety of complex geometric and reflectance characteristics. Some sample images for the 20 objects are shown in Figure 5. We use 22 images from each object to form the training set. Of the 22 images, $p \in \{3, 4, 5, 6\}$ images are randomly selected as labeled data and the rest as unlabeled data. We perform 10 random splits on each configuration and Table 3 reports the average results. From the results, our method also outperforms other methods under all four settings.

6 Conclusion

In this paper, we have presented a new probabilistic LDA model. This semi-supervised, heteroscedastic extension allows it to overcome some serious limitations of LDA. As

Table 2. Recognition error rates (in mean±std-dev) on PIE for four different p values. 1st Table: $p = 3$; 2nd Table: $p = 4$; 3rd Table: $p = 5$; 4th Table: $p = 6$.

Method	Error rate (unlabeled)	Error rate (test)
PLDA	0.8421±0.0142	0.8492±0.0212
SPPCA	0.4509±0.0487	0.4798±0.0590
S^{2}PPCA	0.3367±0.0088	0.3639±0.0139
S^{2}HPLDA	**0.3066±0.0131**	**0.3109±0.0397**

Method	Error rate (unlabeled)	Error rate (test)
PLDA	0.7549±0.0451	0.7469±0.0532
SPPCA	0.2741±0.0202	0.2654±0.0073
S^{2}PPCA	0.2545±0.0110	0.2520±0.0046
S^{2}HPLDA	**0.2096±0.0324**	**0.2225±0.0066**

Method	Error rate (unlabeled)	Error rate (test)
PLDA	0.7029±0.0018	0.7201±0.0154
SPPCA	0.2080±0.0153	0.2409±0.0120
S^{2}PPCA	0.2011±0.0055	0.2330±0.0046
S^{2}HPLDA	**0.1743±0.0177**	**0.1933±0.0108**

Method	Error rate (unlabeled)	Error rate (test)
PLDA	0.7096±0.0351	0.7215±0.0420
SPPCA	0.1875±0.0104	0.2119±0.0143
S^{2}PPCA	0.1590±0.0390	0.1724±0.0347
S^{2}HPLDA	**0.1220±0.0149**	**0.1450±0.0204**

Fig. 5. Some images for different objects in the COIL database

Table 3. Recognition error rates (in mean±std-dev) on COIL for four different p values. 1st Table: $p = 3$; 2nd Table: $p = 4$; 3rd Table: $p = 5$; 4th Table: $p = 6$.

Method	Error rate (unlabeled)	Error rate (test)
PLDA	0.4026±0.0112	0.4000±0.0311
SPPCA	0.7303±0.1172	0.7195±0.1393
S^{2}PPCA	0.3303±0.0428	0.3270±0.0410
S^{2}HPLDA	**0.3145**±**0.0651**	**0.3015**±**0.0474**

Method	Error rate (unlabeled)	Error rate (test)
PLDA	0.3694±0.0118	0.3850±0.0156
SPPCA	0.6958±0.0727	0.7075±0.0658
S^{2}PPCA	0.3500±0.0039	0.3195±0.0021
S^{2}HPLDA	**0.3167**±**0.0314**	**0.3005**±**0.0375**

Method	Error rate (unlabeled)	Error rate (test)
PLDA	0.3471±0.0208	0.3290±0.0792
SPPCA	0.7691±0.0769	0.7815±0.0884
S^{2}PPCA	0.3221±0.0062	0.2865±0.0346
S^{2}HPLDA	**0.2438**±**0.0265**	**0.2670**±**0.0566**

Method	Error rate (unlabeled)	Error rate (test)
PLDA	0.3156±0.0707	0.3085±0.0559
SPPCA	0.7844±0.0398	0.7840±0.0226
S^{2}PPCA	0.3391±0.0420	0.3270±0.0028
S^{2}HPLDA	**0.2250**±**0.0312**	**0.2200**±**0.0354**

said earlier in the paper, one natural extension is a fully Bayesian extension to boost the generalization performance of the probabilistic model. Another possibility is to apply the kernel trick to introduce nonlinearity into the model using techniques such as that in [36].

Acknowledgments

This research has been supported by General Research Fund 621407 from the Research Grants Council of the Hong Kong Special Administrative Region, China. We would like to thank Shipeng Yu for providing the source code of SPPCA and S^{2}PPCA.

References

1. Jolliffe, I.T.: Principal Component Analysis. 2nd edn. Springer-Verlag, New York (2002)
2. Fukunaga, K.: Introduction to Statistical Pattern Recognition. Academic Press, New York (1991)
3. Chen, L., Liao, H., Ko, M., Lin, J., Yu, G.: A new LDA-based face recognition system which can solve the small sample size problem. Pattern Recognition **33**(10) (2000) 1713–1726
4. Krzanowski, W.J., Jonathan, P., McCarthy, W.V., Thomas, M.R.: Discriminant analysis with singular covariance matrices: methods and applications to spectroscopic data. Applied Statistics **44**(1) (1995) 101–115
5. Belhumeur, P.N., Hespanha, J.P., Kriegman, D.J.: Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection. IEEE Transactions on Pattern Analysis and Machine Intelligence **19**(7) (1997) 711–720
6. Ye, J., Li, Q.: A two-stage linear discriminant analysis via QR-decomposition. IEEE Transactions on Pattern Analysis and Machine Intelligence **27**(6) (2005) 929–941
7. Cevikalp, H., Neamtu, M., Wilkes, M., Barkana, A.: Discriminative common vectors for face recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence **27**(1) (2005) 4–13
8. Wang, X., Tang, X.: Dual-space linear discriminant analysis for face recognition. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Washington, DC (2004) 564–569
9. Ye, J., Janardan, R., Li, Q.: Two-dimensional linear discriminant analysis. In: Advances in Neural Information Processing Systems 17, Vancouver, British Columbia, Canada (2005) 1529–1536
10. Cai, D., He, X., Han, J.: Semi-supervised discriminant analysis. In: Proceedings of the IEEE International Conference on Computer Vision, Rio de Janeiro, Brazil (2007)
11. Zhang, Y., Yeung, D.Y.: Semi-supervised discriminant analysis using robust path-based similarity. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Anchorage, Alaska (2008)
12. Zhang, Y., Yeung, D.Y.: Semi-supervised discriminant analysis via CCCP. In: Proceedings of the 19th European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases, Antwerp, Belgium (2008) 644–659
13. Chapelle, O., Zien, A., Schölkopf, B., eds.: Semi-Supervised Learning. MIT Press, Boston (2006)
14. Hastie, T., Tibshirani, R.: Discriminant analysis by Gaussian mixture. Journal of the Royal Statistical Society, Series B **58**(1) (1996) 155–176

15. Kumar, N., Andreou, A.G.: Heteroscedastic discriminant analysis and reduced rank HMMS for improved speech recognition. Speech Communication **26**(4) (1998) 283–297
16. Loog, M., Duin, R.P.W.: Linear dimensionality reduction via a heteroscedastic extension of LDA: the Chernoff criterion. IEEE Transactions on Pattern Analysis and Machine Intelligence **26**(6) (2004) 732–739
17. Dai, G., Yeung, D.Y., Chang, H.: Extending kernel fisher discriminant analysis with the weighted pairwise Chernoff criterion. In: Proceedings of the Ninth European Conference on Computer Vision, Graz, Austria (2006) 308–320
18. Ioffe, S.: Probabilistic linear discriminant analysis. In: Proceedings of the 9th European Conference on Computer Vision, Graz, Austria (2006) 531–542
19. Yu, S., Yu, K., Tresp, V., Kriegel, H.P., Wu, M.: Supervised probabilistic principal component analysis. In: Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Philadelphia, PA, USA (2006) 464–473
20. Prince, S.J.D., Elder, J.H.: Probabilistic linear discriminant analysis for inferences about identity. In: Proceedings of the 11th IEEE International Conference on Computer Vision, Rio de Janeiro, Brazil (2007) 1–8
21. Neal, R.M.: Bayesian Learning for Neural Network. Springer-Verlag, New York (1996)
22. Tipping, M.E., Bishop, C.M.: Probabilistic principal component analysis. Journal of the Royal Statistic Society, B **61**(3) (1999) 611–622
23. Bach, F.R., Jordan, M.I.: A probabilistic interpretation of canonical correlation analysis. Technical Report 688, Department of Statistics, University of California, Berkeley (2005)
24. He, X., Yan, S., Hu, Y., Niyogi, P., Zhang, H.J.: Face recognition using Laplacianfaces. IEEE Transactions on Pattern Analysis and Machine Intelligence **27**(3) (2005) 328–340
25. Belkin, M., Niyogi, P., Sindhwani, V.: Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. Journal of Machine Learning Research **7** (2006) 2399–2434
26. Zelnik-Manor, L., Perona, P.: Self-tuning spectral clustering. In: Advances in Neural Information Processing Systems 17, Vancouver, British Columbia, Canada (2005) 1601–1608
27. Dempster, A.P., Laird, N.M., Rubin, D.B.: Maximum likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistic Society, B **39**(1) (1977) 1–38
28. Tipping, M.E., Bishop, C.M.: Mixtures of probabilistic principal component analysers. Neural Computation **11**(2) (1999) 443–482
29. Joachims, T.: Transductive inference for text classification using support vector machines. In: Proceedings of the Sixteenth International Conference on Machine Learning, San Francisco, CA, USA (1999) 200–209
30. Bennett, K., Demiriz, A.: Semi-supervised support vector machines. In: Advances in Neural Information Processing Systems 11, Vancouver, British Columbia, Canada (1998) 368–374
31. MacKay, D.J.C.: Comparison of approximate methods for handling hyperparameters. Neural Computation **11**(5) (1999) 1035–1068
32. Bishop, C.M.: Pattern Recognition and Machine Learning. Springer, New York (2006)
33. Turk, M., Pentland, A.: Eigenfaces for recognition. Journal of Cognitive Neuroscience **3**(1) (1991) 71–86
34. Sim, T., Baker, S., Bsat, M.: The CMU pose, illumination and expression database. IEEE Transactions on Pattern Analysis and Machine Intelligence **25**(12) (2003) 1615–1618
35. Nene, S.A., Nayar, S.K., Murase, H.: Columbia object image library (COIL-20). Technical Report 005, CUCS (1996)
36. Lawrence, N.D.: Probabilistic non-linear principal component analysis with Gaussian process latent variable models. Journal of Machine Learning Research **6** (2005) 1783–1816