A Unified Framework for Generalized Linear Discriminant Analysis

Shuiwang Ji and Jieping Ye
Department of Computer Science and Engineering
Arizona State University
{shuiwang.ji, jieping.ye}@asu.edu

Abstract

Linear Discriminant Analysis (LDA) is one of the well-known methods for supervised dimensionality reduction. Over the years, many LDA-based algorithms have been developed to cope with the curse of dimensionality. In essence, most of these algorithms employ various techniques to deal with the singularity problem, which occurs when the data dimensionality is larger than the sample size. They have been applied successfully in various applications. However, there is a lack of a systematic study of the commonalities and differences of these algorithms, as well as their intrinsic relationships. In this paper, a unified framework for generalized LDA is proposed via a transfer function. The proposed framework elucidates the properties of various algorithms and their relationships. Based on the presented analysis, we propose an efficient model selection algorithm for LDA. We conduct extensive experiments using a collection of high-dimensional data, including text documents, face images, gene expression data, and gene expression pattern images, to evaluate the proposed theories and algorithms.

1. Introduction

Recent years have witnessed an increasing prevalence of datasets that contain a large number of dimensions, including microarray gene expression data, gene expression pattern images, text documents, face images, etc. The proliferation of these data has tempted the researchers to discover knowledge and extract patterns from the data using computational approaches. One of the key issues in high-dimensional data analysis is the *curse of dimensionality* [2], i.e., an enormous number of samples is required to perform accurate prediction on problems with high dimensionality. This is because in high-dimensional spaces, data become extremely sparse and apart from each other. Dimensionality reduction, which extracts a small number of features by removing the irrelevant, redundant, and noisy features can be an effective solution. The commonly used dimensionality reduction methods include supervised approaches such as Linear Discriminant Analysis (LDA) [4, 5], and unsupervised ones such as Principal Component Analysis (PCA) [10]. When the class labels are available, supervised approaches, such as LDA, are usually more effective than unsupervised ones such as PCA in classification.

Linear Discriminant Analysis (LDA) is a classical statistical approach for dimensionality reduction [5, 8]. LDA computes an optimal transformation (projection) by minimizing the within-class distance and maximizing the between-class distance simultaneously, thus achieving maximum class discrimination. The optimal transformation (projection) can be readily computed by applying an eigen-decomposition on the scatter matrices. It has been used widely in many applications involving high-dimensional data [1, 3, 11, 7, 15, 17]. However classical LDA requires the so-called *total scatter matrix* to be nonsingular. In many applications involving high-dimensional data such as microarray gene expression data analysis, gene expression pattern image analysis, text categorization, and face recognition, the total scatter matrix can be singular, since the data points are from a very high-dimensional space and the sample size does not exceed this dimension in general. This is known as the *singularity problem* [9].

In recent years, many approaches have been proposed to deal with the singularity problem, including PCA+LDA [1], Regularized LDA [7], Null space LDA [3], Orthogonal Centroid Method [14], Uncorrelated LDA [17], Orthogonal LDA [17], and LDA/GSVD [9]. These algorithms have been applied successfully in various domains, such as PCA+LDA in face recognition [1], OCM in text categorization [14], and RLDA in microarray gene expression data analysis [7]. However, there is a lack of a systematic study to explore the commonalities and differences of these algorithms, as well as their intrinsic relationship. This has been a challenging task, since different algorithms apply completely different schemes when dealing with the singularity problem.

In this paper, we propose a unified framework for generalized LDA via a transfer function $\Phi : \mathbb{R} \rightarrow \mathbb{R}$. We show

978-1-4244-2243-2/08/$25.00 ©2008 IEEE

that various LDA-based algorithms differ in their transfer functions. Details on this unified framework as well as the transfer functions for different LDA-based algorithms are given in Section 3. The proposed framework elucidates the properties of various algorithms and their relationships. More specifically, ULDA is shown to be a special case of PCA+LDA and RLDA. We show that under a mild condition which tends to hold for high-dimensional data, the ULDA transformation maps all data points from the same class to a common vector. This leads to a perfect separation between different classes, however it may also lead to overfitting. PCA+LDA and RLDA overcome the overfitting problem by applying the PCA dimensionality reduction and the regularization, respectively. A challenging practical issue is the selection of the optimal dimensionality for the intermediate PCA stage in PCA+LDA, and the optimal value of the regularization parameter in RLDA. Motivated by the relationship between PCA+LDA and other methods in the proposed framework, we develop a model selection algorithm for PCA+LDA. Experiments on a collection of high-dimensional data sets validated the proposed theories and algorithm.

2. Overview of Linear Discriminant Analysis

Given a data matrix $X = [x_1, x_2, \cdots, x_n] \in \mathbb{R}^{d\times n}$ consisting of n samples $\{x_i\}_{i=1}^n$ in $\mathbb{R}^d$, we focus on linear feature extraction that constructs a small number, ℓ, of features by applying a linear transformation $G \in \mathbb{R}^{d\times \ell}$ that maps each data point x_i of X, for $1 \le i \le n$, in the d-dimensional space to a vector x_i^L in the ℓ-dimensional space as follows: $G : x_i \in \mathbb{R}^d \to x_i^L = G^T x_i \in \mathbb{R}^\ell \, (\ell < d)$. Let X be partitioned into k classes as $X = [X_1, \cdots, X_k]$. In classical LDA, three scatter matrices, i.e., *within-class*, *between-class*, and *total* scatter matrices are defined as follows [5]:

$$S_w = \frac{1}{n}\sum_{i=1}^{k}\sum_{x\in X_i}(x - c^{(i)})(x - c^{(i)})^T, \quad (1)$$

$$S_b = \frac{1}{n}\sum_{i=1}^{k} n_i (c^{(i)} - c)(c^{(i)} - c)^T, \quad (2)$$

$$S_t = \frac{1}{n}\sum_{j=1}^{n}(x_j - c)(x_j - c)^T, \quad (3)$$

where n_i is the sample size of the i-th class X_i, $c^{(i)}$ is the centroid of the i-th class, and c is the global centroid. It follows from the definitions that trace(S_w) measures the within-class cohesion, trace(S_b) measures the between-class separation, and trace(S_t) measures the variance of the data, where the trace [6] of a square matrix is the summation of its diagonal entries. It can be verified that

$$S_t = S_b + S_w. \quad (4)$$

In the lower-dimensional space, the scatter matrices S_w, S_b, and S_t become $G^T S_w G$, $G^T S_b G$, and $G^T S_t G$, respectively. An optimal transformation G of LDA is computed by maximizing the following objective function [5]:

$$f(G) = \text{trace}\left((G^T S_w G)^{-1} G^T S_b G\right). \quad (5)$$

The optimization problem is equivalent to finding $y \in \mathbb{R}^d$ that satisfies $S_b y = \lambda S_w y$, for $\lambda \neq 0$ [5]. The solution can be obtained by applying an eigen-decomposition to the matrix $S_w^{-1} S_b$, if S_w is nonsingular. Since $S_t = S_b + S_w$, the solution is also given by the eigenvectors of $S_t^{-1} S_b$, assuming S_t is nonsingular. There exist no more than $k - 1$ eigenvectors corresponding to nonzero eigenvalues, since the rank of the matrix S_b is bounded from above by $k - 1$. Therefore, the reduced dimensionality, ℓ, of LDA is at most $k - 1$. One limitation of the classical LDA formulation is that the total scatter matrix S_t is required to be nonsingular, which may not hold for high-dimensional, low sample size data, such as microarray gene expression data, gene expression pattern images, etc. This *singularity problem* has been the driving force for the development of different generalized LDA algorithms [9].

A common way to deal with the singularity problem is to apply an intermediate dimensionality reduction stage such as PCA [10] to reduce the data dimensionality before classical LDA is applied. The algorithm is known as PCA+LDA, or subspace LDA [1]. In this two-stage PCA+LDA algorithm, the discriminant stage is preceded by a dimensionality reduction stage using PCA. The dimensionality, p, of the subspace transformed by PCA is chosen such that the "reduced" total scatter matrix in this subspace is nonsingular, so that classical LDA can be applied. The optimal value of p is commonly estimated through cross-validation.

Regularization is commonly applied to deal with the singularity of S_t. The algorithm is known as Regularized LDA, or RLDA in short [7, 9]. The key idea is to add a constant $\mu > 0$ to the diagonal elements of S_t as $S_t + \mu I_d$, where I_d is the identity matrix of size d. It is easy to verify that $S_t + \mu I_d$ is positive definite [6], hence nonsingular. Cross-validation is commonly applied to estimate the optimal value of μ. It has been shown [19] that the regularization employed in LDA can be interpreted from the regularization network perspective in the binary-class case.

In [3], the null space LDA (NLDA) was proposed, where the between-class distance is maximized in the null space of the within-class scatter matrix. The singularity problem is thus avoided implicitly. The efficiency of the algorithm can be improved by first removing the null space of the total scatter matrix. It is based on the observation that the null space of the total scatter matrix is the intersection of the null spaces of the between-class and within-class scatter matrices. The Orthogonal Centroid Method (OCM) [14] maximizes the between-class distance only by omitting the

Table 1. Transfer functions for different LDA-based algorithms.

	PCA+LDA	RLDA	U(O)LDA	OCM
$\Phi(\lambda_i)$	$\begin{cases} \lambda_i, & \text{for } 1 \leq i \leq p \\ 0, & \text{for } i > p \end{cases}$	$\begin{cases} \lambda_i + \mu, & \text{for } 1 \leq i \leq t \\ 0, & \text{for } i > t \end{cases}$	λ_i	1

within-class information. The optimal transformation of OCM is given by the top eigenvectors of the between-class scatter matrix S_b.

In [17], a family of generalized discriminant analysis algorithms based on a new objective function were presented. Uncorrelated LDA (ULDA) and Orthogonal LDA (OLDA) are two representative algorithms from this family. The features in the reduced space of ULDA are uncorrelated, while the transformation, G, of OLDA is orthogonal, i.e., $G^T G = I_\ell$. The LDA/GSVD algorithm [9] which overcomes the singularity problem via the Generalized Singular Value Decomposition (GSVD) also belongs to this family.

3. A Unified Framework for Generalized LDA

In essence, most of the LDA-based algorithms discussed in the last section employ various techniques to deal with the singularity problem. In this section, we propose a four-step unified framework for generalized LDA algorithms as follows:

1. Compute the set of eigenvalues, $\{\lambda_i\}_{i=1}^d$, of S_t in Eq. (3) and the corresponding eigenvectors $\{u_i\}_{i=1}^d$, with $\lambda_1 \geq \cdots \geq \lambda_d$. Then, S_t can be expressed as $S_t = \sum_{i=1}^d \lambda_i u_i u_i^T$.

2. Given a transfer function Φ, let $\tilde{\lambda}_i = \Phi(\lambda_i)$, for all i. Construct matrix $\tilde{S}_t = \sum_{i=1}^d \tilde{\lambda}_i u_i u_i^T$.

3. Compute the set of eigenvectors, $\{\phi_i\}_{i=1}^q$, of $\tilde{S}_t^+ S_b$ corresponding to nonzero eigenvalues, where $q = \text{rank}(S_b)$, $\tilde{S}_t^+$ denotes the pseudo-inverse of $\tilde{S}_t$ [6]. Construct matrix $G = [\phi_1, \cdots, \phi_q]$.

4. Optional orthogonalization step: Compute the QR decomposition [6] of G as $G = QR$, where $Q \in \mathbb{R}^{d \times q}$ has orthonormal columns and $R \in \mathbb{R}^{q \times q}$ is upper triangular.

The final transformation is given by matrix G from step 3, if the optional orthogonalization step is not applied, and by matrix Q from step 4 otherwise. In this framework, different transfer functions, Φ, in step 2 lead to different LDA algorithms, as summarized below:

- In PCA+LDA, the intermediate dimensionality reduction stage by PCA keeps the top p eigenvalues of S_t, thus it applies the following linear step function: $\Phi(\lambda_i) = \lambda_i$, for $1 \leq i \leq p$, and $\Phi(\lambda_i) = 0$, for $i > p$. The optional orthogonalization step is not employed in PCA+LDA.

- In Regularized LDA (RLDA), a regularization term is applied to S_t as $S_t + \mu I_d$, for some $\mu > 0$. It corresponds to the use of the following transfer function: $\Phi(\lambda_i) = \lambda_i + \mu$, for all i. The optional orthogonalization step is not employed in RLDA.

- In Uncorrelated LDA (ULDA), the optimal transformation consists of the top eigenvectors of $S_t^+ S_b$ [17]. The corresponding transfer function is thus given by $\Phi(\lambda_i) = \lambda_i$, for all i. The same transfer function is used in Orthogonal LDA (OLDA). Unlike ULDA, the orthogonalization step is applied in OLDA.

- In Orthogonal Centroid Method (OCM), the optimal transformation is given by the top eigenvectors of S_b [14]. The transfer function is thus given by $\Phi(\lambda_i) = 1$, for all i. Since the eigenvectors of S_b forms an orthonormal set, the optional orthogonalization step is not necessary in OCM.

Let

$$S_t = U \text{diag}\left(\Sigma_t, 0\right) U^T$$

be the SVD [6] of S_t, where U is orthogonal and $\Sigma_t \in \mathbb{R}^{t \times t}$ is diagonal and nonsingular with $t = \text{rank}(S_t)$. Let $U = (U_1, U_2)$ be a partition of U, such that $U_1 \in \mathbb{R}^{d \times t}$ and $U_2 \in \mathbb{R}^{d \times (d-t)}$. Since $S_t = S_b + S_w$, the null space of S_t is a subset of the null space of S_b. That is, $S_b U_2 = 0$. It follows that $(S_t + \mu I_d)^{-1} S_b$ can be expressed as

$$\begin{aligned} & U \left(\begin{pmatrix} \Sigma_t & 0 \\ 0 & 0 \end{pmatrix} + \mu I_d \right)^{-1} U^T S_b U U^T \\ = \; & U \begin{pmatrix} \Sigma_t + \mu I_t & 0 \\ 0 & \mu I_{d-t} \end{pmatrix}^{-1} \begin{pmatrix} U_1^T S_b U_1 & 0 \\ 0 & 0 \end{pmatrix} U^T \\ = \; & U_1 \left((\Sigma_t + \mu I_t)^{-1} U_1^T S_b U_1 \right) U_1^T. \end{aligned} \qquad (6)$$

Eq. (6) above shows that the regularization term is only effective for the nonzero eigenvalues in Σ_t and has no effect on the zero eigenvalues of S_t. Thus, we can apply the following transfer function for RLDA: $\Phi(\lambda_i) = \lambda_i + \mu$, for all $i = 1, \cdots, t$, and zero otherwise, where $t = \text{rank}(S_t)$.

The transfer functions for different algorithms are summarized in Table 1. In null space LDA (NLDA) [3], the data is first projected onto the null space of S_w, which is then followed by classical LDA. It is not clear which transfer function Φ corresponds to the projection onto the null space of S_w. In [18], the equivalence relationship between NLDA and OLDA was established under a mild condition

$$C1 : \text{rank}(S_t) = \text{rank}(S_b) + \text{rank}(S_w), \qquad (7)$$

which has been shown to hold for many high-dimensional data. Thus, for high-dimensional data, we can use the following transfer function for NLDA: $\Phi(\lambda_i) = \lambda_i$, for all i. Note that the proposed unified framework has the same flavor as the framework for the construction of cluster kernels in [16].

4. Analysis

The proposed framework from the last section summarizes the commonalities and differences of various LDA-based algorithms. It helps us in understanding the key features of various algorithms as well as their relationships.

We can observe from Table 1 that when the reduced dimensionality p in the PCA stage of PCA+LDA is chosen to be the rank of S_t, that is, the PCA stage keeps all the information, then the transfer functions for PCA+LDA and ULDA are identical, that is, PCA+LDA is equivalent to ULDA in this case. From Table 1, the transfer function for RLDA equals the one for ULDA when $\mu = 0$. Thus, ULDA can be considered as a special case of both PCA+LDA and RLDA.

The effectiveness of PCA+LDA and RLDA is critically dependent on the value of the regularization parameter involved, which is commonly estimated via cross-validation from a set of candidates. Selecting an optimal value for a parameter such as p in PCA+LDA and μ in RLDA is called *model selection* [8]. RLDA with μ approaching zero, as well as PCA+LDA with $p = \text{rank}(S_t)$, is essentially ULDA. Under Condition C1 in Eq. (7), the transformation matrix of ULDA has been shown to lie in the null space of S_w [18], that is, $G^T S_w = 0$. In this case, it follows from Eq. (1) that $G^T\left(x - c^{(i)}\right) = 0$, for all $x \in X_i$, and $G^T x = G^T c^{(i)}$. This shows that the ULDA transformation maps all data points from the same class to a common vector, provided that Condition C1 is satisfied. A similar result has been shown in [12] when all classes in the dataset have a common sample size. This leads to a perfect separation between different classes, however it may also lead to overfitting. RLDA overcome this limitation by choosing a nonzero regularization value μ, while PCA+LDA overcomes this limitation by setting $p < \text{rank}(S_t)$.

The above analysis shows the significance of the regularization and PCA dimensionality reduction in RLDA and PCA+LDA, especially when the data is noisy. This is confirmed in the experimental studies below. The optimal value of p is commonly estimated using cross-validation from the range $[k,\, \text{rank}(S_t)]$. We next propose an efficient model selection algorithm that can choose the optimal p in the range $[k,\, \text{rank}(S_t)]$ efficiently.

Note that the three scatter matrices in Eqs. (1)–(3) are all symmetric and positive semi-definite. Thus, we can define three matrices, called H_w, H_b, and H_t, so that

$$H_w H_w^T = S_w,\ H_b H_b^T = S_b,\ H_t H_t^T = S_t, \tag{8}$$

where $H_w, H_t \in \mathbb{R}^{d\times n}$ and $H_b \in \mathbb{R}^{d\times k}$. Let $H_t = U_1 \Sigma V_1^T$ be the skinny SVD of H_t where U_1 is defined in Section 3, Σ is diagonal, and V_1 has orthonormal columns. Then, $S_t = H_t H_t^T = U_1 \Sigma_t U_1^T$ where $\Sigma_t = \Sigma^2$. When $p = \text{rank}(S_t)$, PCA projects the original data onto the column space of U_1. It can be seen from the definitions of S_t and S_b that in this dimensionality-reduced space, the total scatter and between-class scatter matrices, denoted as $\tilde{S}_t$ and $\tilde{S}_b$, become

$$\tilde{S}_t = U_1^T S_t U_1 = \Sigma_t, \quad \tilde{S}_b = U_1^T S_b U_1. \tag{9}$$

It follows that performing classical LDA in this PCA-transformed space requires the diagonalization of the matrix $\tilde{S}_t^{-1}\tilde{S}_b = \Sigma_t^{-1} U_1^T S_b U_1$, which is given by

$$\Sigma_t^{-1/2}\Sigma_t^{-1/2} U_1^T H_b H_b^T U_1 \Sigma_t^{-1/2}\Sigma_t^{1/2},$$

since $S_b = H_b H_b^T$. Let $B = \Sigma_t^{-1/2} U_1^T H_b$, and $B = U_b \Sigma_b V_b^T$ be the SVD of B. Then,

$$\begin{aligned}
& \Sigma_t^{-1/2}\Sigma_t^{-1/2} U_1^T H_b H_b^T U_1 \Sigma_t^{-1/2}\Sigma_t^{1/2} \\
= \ & \Sigma_t^{-1/2}(\Sigma_t^{-1/2} U_1^T H_b)(\Sigma_t^{-1/2} U_1^T H_b)^T \Sigma_t^{1/2} \\
= \ & \Sigma_t^{-1/2} B B^T \Sigma_t^{1/2} \\
= \ & (\Sigma_t^{-1/2} U_b)\Sigma_b^2 (\Sigma_t^{-1/2} U_b)^{-1}.
\end{aligned} \tag{10}$$

It follows from Eq. (10) that the matrix $\Sigma_t^{-1/2} U_b$ diagonalizes the matrix $\tilde{S}_t^{-1}\tilde{S}_b$. This leads to a two-stage procedure for computing the eigenvectors of $\tilde{S}_t^{-1}\tilde{S}_b$: (1) Compute the SVD of H_t as $H_t = U_1 \Sigma V_1^T$; and (2) Compute the SVD of $\Sigma_t^{-1/2} U_1^T H_b$ as $\Sigma_t^{-1/2} U_1^T H_b = U_b \Sigma_b V_b^T$. The eigenvectors of $\tilde{S}_t^{-1}\tilde{S}_b$ correspond to nonzero eigenvalues are given by columns of $U_1 \Sigma_t^{-1/2} U_b$.

The key observations that underlie our efficient model selection algorithm are that the first stage needs to be computed only once regardless of the number of p values tried, and for the second stage, the matrix $B = \Sigma_t^{-1/2} U_1^T H_b$ for small values of p are submatrices of those for larger values. In particular, once the matrix B for the maximum p, i.e., $\text{rank}(S_t)$, is computed, subsequent $B's$ can be obtained directly by removing rows of the original B incrementally. Therefore, except for $p = \text{rank}(S_t)$, all that the algorithm needs to do is to remove the last row of current matrix B and compute the SVD of this matrix of decreasing size.

Note that the complexity of the second stage does not depend on d, the data dimensionality. Therefore, for high-dimensional data where d is larger than the sample size n, the second stage in the model selection algorithm has a relatively low computational cost.

5. Experiments

In this section, we perform experiments to evaluate the proposed theories and algorithms. We also report the results obtained by Support Vector Machines (SVM) and correlation-based LDA (corrLDA) proposed in [20]. When evaluating the classification performance in the dimensionality-reduced space, we report the accuracies obtained by both Nearest-Centroid (NC) and Nearest-Neighbor (NN) classifiers.

We use eight datasets in the experiments and their statistics are summarized in Tables 2 and 3. The datasets fall into four categories. re0 and re1 are two text document datasets that are derived from the *Reuters-21578* text categorization test collection Distribution 1.0[1]. ORL[2] and AR[3] are two widely used face image datasets. 14_Tumors and Brain tumor are gene expression datasets. Fruitfly [13] is the gene expression pattern image dataset where the first three (correspond to stage ranges 1-3, 4-6, and 7-8) and six (correspond to stage ranges 1-3, 4-6, 7-8, 9-10, 11-12, and 13-16) classes are used in the experiments. They are denoted by Fruitfly(3) and Fruitfly(6), respectively.

5.1. Performance Evaluation

The six datasets in Table 2 are high-dimensional and they all have a small number of samples in each class. We observe that the C1 condition, defined in Eq. (7), holds for all the six datasets. Hence, ULDA will project all samples in the same class to a common point on these datasets. For each of the six datasets, we randomly partition the entire dataset into training and test sets using the ratio 1:1 and the performance of five methods (RLDA, PCA+LDA, ULDA, corrLDA, and SVM) are recorded. The parameters of corrLDA and SVM are tuned by cross-validation. This entire process is repeated 30 times and Table 2 reports the mean accuracies over 30 random partitions.

From Table 2 we can observe that ULDA achieves similar classification performance with PCA+LDA and RLDA. Compared to re0 and re1, the ORL, AR, 14_Tumors, and Brain tumor datasets have a smaller number of samples in each class (less than 20 data points). We can also observe from the results that SVM achieves similar performance with RLDA and PCA+LDA when the number of classes is small. But when the number of classes is large, LDA-based algorithms tend to produce higher accuracies since they can handle multi-class problems naturally. In general, the performance difference between SVM and the best LDA-based method is small. It is interesting to observe that PCA+LDA outperforms corrLDA in our experiments. Since PCA+LDA is a special case of corrLDA, we expect that such difference in performance may be due to the lack of effective model selection strategy for corrLDA.

To evaluate the relative performance of the three methods when there are an increasing number of samples in each class, we apply the algorithms on the Fruitfly(3) and Fruitfly(6) datasets with increasing proportion of data in the training set. This process is repeated 30 times and the mean accuracies are summarized in Table 3. The results on Fruitfly(6) are also plotted in Figure 1. We can observe from Figure 1 and Table 3 that as the size of the training set increases, the classification performance of RLDA and PCA+LDA improves steadily. On the other hand, the performance of ULDA decreases as the number of samples in each class increases. This shows that when there are relatively large number of samples in each class, ULDA may suffer from the overfitting problem.

5.2. PCA+LDA Algorithm

The performance studies in Section 5.1 show that the regularization employed in PCA+LDA are effective to prevent the overfitting problem. To examine this effect in detail, we visualize the samples after the projection by PCA+LDA with different parameter settings in this section. In particular, we ran PCA+LDA with $p = (299, 180, 51, 15)$ on a training set of 300 Fruitfly(3) images and apply the projection to a test set of 2405 images. There are $k = 3$ classes in the Fruitfly(3) dataset, and all images are projected onto a 2D plane. In Figure 2, we show the projection of the training images (top row) and a subset of test images (bottom row) for PCA+LDA. We depict each image by the corresponding stage range (1, 2, and 3). We can observe from Figure 2 that when $p = 299$, which correspond to the case where no regularization is applied, all training points from the same class are mapped to a common point, which leads to the perfect separation in the training set. However, the test data points are scattered around and the classification accuracy using Nearest-Centroid (NC) classifier is about 63.37% only. Note that PCA+LDA with $= \text{rank}(S_t)$ are equivalent to ULDA. When the value of p decreases, the diameter of each class in the training set increases, while the three classes in the test set are better separated. We apply the PCA+LDA model selection algorithm and the optimal value of p obtained is 51. Under this optimal parameter value, PCA+LDA achieves its highest accuracy. When the value of p decreases, the accuracy starts to decrease. This experiment shows the effectiveness of the regularization in PCA+LDA, as well as the importance of model selection in estimating the optimal value of p.

To evaluate the PCA+LDA model selection algorithm, we randomly partition the re0 data into training and test sets using the ratio of 1:1, and the training data are fed into the proposed PCA+LDA model selection algorithm to compute the optimal p. We compare the accuracy achieved by this p

[1] http://www.daviddlewis.com/resources/testcollections/reuters21578/

[2] http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html

[3] http://cobweb.ecn.purdue.edu/RVL/ARdatabase/ARdatabase.html

Table 2. Summary of the mean accuracies of RLDA, PCA+LDA, ULDA, corrLDA, and SVM on six datasets. The datasets are randomly partitioned into training and test sets using the ratio 1:1 and the results are averaged over 30 splittings. The results for corrLDA on the last six datasets are not available due to computational problems.

Data set	re0		re1		ORL		AR		Brain tumor		14 tumors	
Statistics	n=320, d=2887, k=4		n=490, d=3759, k=5		n=400, d=10304, k=40		n=650, d=8888, k=50		n=308, d=15009, k=26		n=90, d=5920, k=5	
Classifier	NC	NN	NC	NN	NC	NN	NC	NN	NC	NN	NC	NN
RLDA	84.84	83.25	94.59	94.42	91.63	91.63	93.10	93.10	85.85	85.85	68.77	68.77
PCA+LDA	84.07	82.19	94.78	94.16	90.22	90.73	92.72	92.40	86.30	85.26	69.15	66.37
ULDA	84.77	79.67	94.61	94.61	91.63	91.63	93.10	93.10	85.85	85.85	68.77	68.77
corrLDA	81.36	77.56	91.97	91.39	–	–	–	–	–	–	–	–
SVM	83.65		94.46		95.05		88.40		85.04		62.01	

Table 3. Summary of the mean accuracies of RLDA, PCA+LDA, ULDA, corrLDA, and SVM on the Fruitfly(3) and Fruitfly(6) datasets. The datasets are randomly partitioned into training and test sets using the ratio 1:1 and the results are averaged over 30 splittings.

Data set	Fruitfly(3) (n=2705, d=384, k=3)								Fruitfly(6) (n=3000, d=384, k=6)							
PCT	3%		5%		10%		15%		3%		5%		10%		15%	
Classifier	NC	NN	NC	NN	NC	NN	NC	NN	NC	NN	NC	NN	NC	NN	NC	NN
RLDA	84.67	84.36	87.32	86.48	89.18	88.14	89.89	86.28	61.50	60.97	67.36	66.03	71.81	69.62	74.13	71.31
PCA+LDA	84.14	80.22	86.04	72.91	87.99	81.88	88.70	77.42	61.26	59.17	66.05	62.67	70.53	66.52	72.64	67.95
ULDA	82.01	82.01	80.53	80.53	69.99	69.99	50.81	51.17	56.97	56.97	54.32	54.32	40.50	40.50	39.40	39.71
corrLDA	81.95	78.97	82.29	78.58	82.95	79.70	82.44	79.03	61.33	54.16	65.86	57.39	70.11	60.92	70.83	62.26
SVM	84.85		87.02		88.80		89.81		57.26		62.91		68.83		72.04	

value with the accuracies for all possible values of p. Results show that the model selection algorithm is effective in estimating the value of p. We also evaluate the relative efficiencies of the two stages in the model selection algorithm on the re0 dataset. Results indicate that even though the second stage needs to be repeated once for each choice of the value for p in cross-validation, the time spent in this stage is still less than that of the first stage. This shows that the overhead of estimating the optimal value of p among a large set of candidates is small.

6. Conclusions and Discussions

In this paper, we propose a unified framework for generalized LDA via a transfer function. The proposed framework elucidates the properties of various algorithms and their relationships. More specifically, ULDA is shown to be a special case of PCA+LDA and RLDA. We further analyze the overfitting problem suffered by ULDA and show how RLDA and PCA+LDA overcome this by applying the regularization and PCA dimensionality reduction, respectively. We further propose an efficient model selection algorithm for PCA+LDA. Experiments are conducted to validate the presented analysis.

Experimental evidences show that, though both RLDA and PCA+LDA are regularized versions of ULDA, they employ different strategies of regularization. The theoretical relationship between these two generalizations of LDA need to be studied further. From the proposed unified framework we can see that all existing generalized LDA algorithms use simple transfer functions. Based on this unified framework, we plan to explore new LDA-based algorithms in the future by employing specific transfer functions. Some possible choices include the polynomial function $\Phi(\lambda_i) = \lambda_i^m$, for some positive integer m, and the exponential function $\Phi(\lambda_i) = e^{\beta\lambda_i}$, for some constant $\beta > 0$.

Acknowledgments

We thank Dr. Manli Zhu and Dr. Aleix M. Martinez for making their correlation-based LDA codes publicly available and their help on using them. This research is sponsored in part by the Arizona State University and by the National Science Foundation Grant IIS-0612069.

References

[1] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman. Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection. *IEEE TPAMI*, 19(7):711–720, 1997.

[2] R. E. Bellman. *Adaptive Control Processes: A Guided Tour*. Princeton University Press, 1961.

[3] L. F. Chen, H. Y. M. Liao, J. C. Lin, M. D. Kao, and G. J. Yu. A new LDA-based face recognition system which can solve the small sample size problem. *Pattern Recognition*, 33(10), 2000.

[4] R. Fisher. The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7:179–188, 1936.

[5] K. Fukunaga. *Introduction to statistical pattern recognition*. Academic Press Professional, Inc., 2nd edition, 1990.

[6] G. H. Golub and C. F. Van Loan. *Matrix Computations*. The Johns Hopkins University Press, 1996.

[7] Y. Guo, T. Hastie, and R. Tibshirani. Regularized linear discriminant analysis and its application in microarrays. *Biostatistics*, 8(1):86–100, 2007.

[8] T. Hastie, R. Tibshirani, and J. Friedman. *The elements of statistical learning : data mining, inference, and prediction*. Springer, 2001.

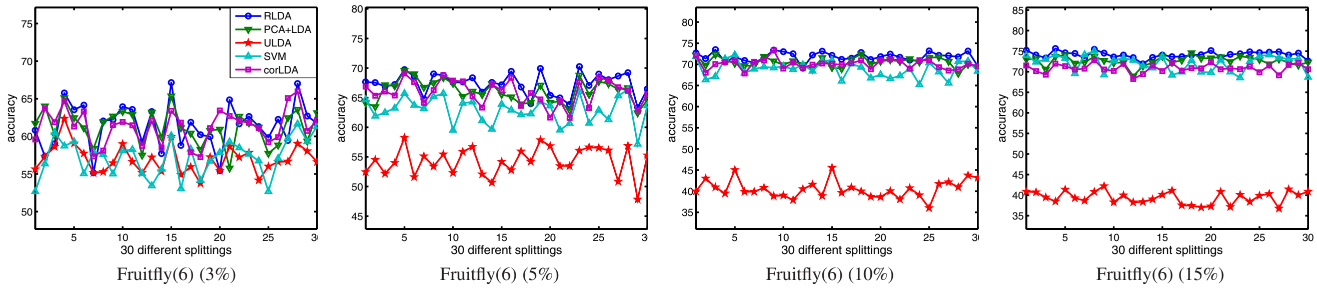

Figure 1. Comparison of the classification accuracies (in percentage) of RLDA, PCA+LDA, ULDA, corrLDA, and SVM on the Fruitfly(6) dataset as the proportion of samples in the training set increases from 3% to 15%.

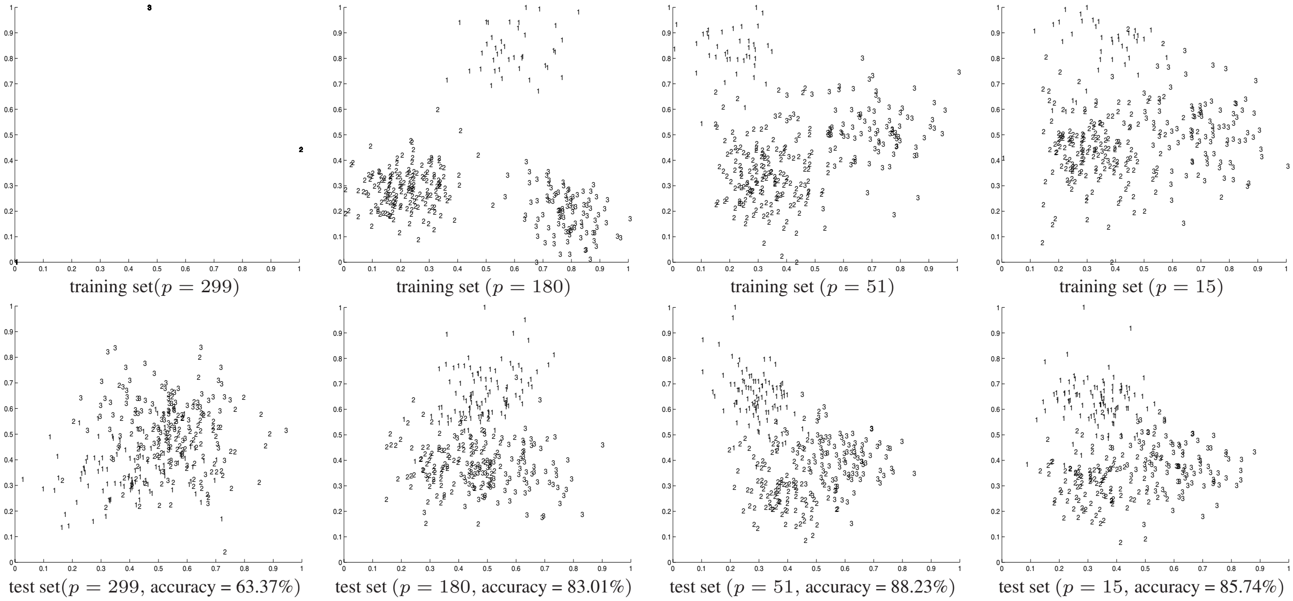

Figure 2. Visualization of the training images (top row) and a subset of test images (bottom row) after projecting onto 2D plane via PCA+LDA with different values of p (299, 180, 51, 15). The training sample size n is 300. Images from the first range (1–3), the second range (4–6), and the third range (7–8) are depicted by "1", "2", and "3", respectively. The test accuracy for each value of p is reported.

[9] P. Howland, M. Jeon, and H. Park. Structure preserving dimension reduction for clustered text data based on the generalized singular value decomposition. *SIAM J. Matrix Anal. Appl.*, 25(1):165–179, 2003.

[10] I. T. Jolliffe. *Principal Component Analysis*. Springer-Verlag, New York, 1986.

[11] A. M. Martinez and A. C. Kak. PCA versus LDA. *IEEE TPAMI*, 23(2):228–233, 2001.

[12] M. Neamtu, H. Cevikalp, M. Wilkes, and A. Barkana. Discriminative common vectors for face recognition. *IEEE TPAMI*, 27(1):4–13, 2005.

[13] P. Tomancak *et al.* Systematic determination of patterns of gene expression during *Drosophila* embryogenesis. *Genome Biology*, 3(12), 2002.

[14] H. Park, M. Jeon, and J. Rosen. Lower dimensional representation of text data based on centroids and least squares. *BIT*, 43(2):1–22, 2003.

[15] X. Wang and X. Tang. A unified framework for subspace face recognition. *IEEE TPAMI*, 26(9):1222–1228, 2004.

[16] J. Weston, C. Leslie, D. Zhou, A. Elisseeff, and W. S. Noble. Semi-supervised protein classification using cluster kernels. In *NIPS 16*. 2004.

[17] J. Ye. Characterization of a family of algorithms for generalized discriminant analysis on undersampled problems. *JMLR*, 6:483–502, 2005.

[18] J. Ye and T. Xiong. Computational and theoretical analysis of null space and orthogonal linear discriminant analysis. *JMLR*, 7:1183–1204, July 2006.

[19] P. Zhang and N. Riedel. Discriminant analysis: A unified approach. In *ICDM*, pages 514–521, 2005.

[20] M. Zhu and A. M. Martinez. Pruning noisy bases in discriminant analysis. *IEEE Transactions Neural Networks*, 19(1):148–157, 2008.