Feature Reduction via Generalized Uncorrelated Linear Discriminant Analysis

Jieping Ye, *Member*, *IEEE*, Ravi Janardan, *Senior Member*, *IEEE*,
Qi Li, *Student Member*, *IEEE*, and Haesun Park

Abstract—High-dimensional data appear in many applications of data mining, machine learning, and bioinformatics. Feature reduction is commonly applied as a preprocessing step to overcome the curse of dimensionality. Uncorrelated Linear Discriminant Analysis (ULDA) was recently proposed for feature reduction. The extracted features via ULDA were shown to be statistically uncorrelated, which is desirable for many applications. In this paper, an algorithm called ULDA/QR is proposed to simplify the previous implementation of ULDA. Then, the ULDA/GSVD algorithm is proposed, based on a novel optimization criterion, to address the singularity problem which occurs in undersampled problems, where the data dimension is larger than the sample size. The criterion used is the regularized version of the one in ULDA/QR. Surprisingly, our theoretical result shows that the solution to ULDA/GSVD is independent of the value of the regularization parameter. Experimental results on various types of data sets are reported to show the effectiveness of the proposed algorithm and to compare it with other commonly used feature reduction algorithms.

Index Terms—Feature reduction, uncorrelated linear discriminant analysis, QR-decomposition, generalized singular value decomposition.

1 Introduction

Feature reduction is important in many applications of data mining, machine learning, and bioinformatics because of the so-called curse of dimensionality [6], [10], [14]. Many methods have been proposed for feature reduction, such as Principal Component Analysis (PCA) [19] and Linear Discriminant Analysis (LDA) [10]. LDA aims to find optimal discriminant features by maximizing the ratio of the between-class distance to the within-class distance of a given data set under supervised learning conditions. It has been successfully employed in many applications including information retrieval [2], [4], face recognition [1], [25], [26], and microarray data analysis [7]. Its simplest implementation, the so-called *classical LDA*, applies an eigen-decomposition on the scatter matrices, but fails when the scatter matrices are singular, as is the case for undersampled data. This is known as the *singularity* or *undersampled* problem [20].

Uncorrelated features[1] are desirable in many applications because they contain minimum redundancy. Motivated by extracting feature vectors having uncorrelated features, *uncorrelated LDA* (ULDA) was recently proposed in [17], [18]. However, the proposed algorithm in [17] involves a sequence of generalized eigenvalue problems, which is computationally expensive for large and high-dimensional data sets. Like classical LDA, it does not address the singularity problem either. We thus call it *classical ULDA*. More details can be found in Section 3.

Classical LDA and classical ULDA were introduced from different perspectives, but it has been found that there is a close relationship between classical LDA and classical ULDA [18]. More precisely, under the assumption that the eigenvalue problem in classical LDA has no multiple eigenvalues, it was shown that classical ULDA is equivalent to classical LDA [18]. In this paper, we will show that the equivalence between these two still holds without the above assumption. Based on this equivalence, ULDA/QR is proposed to simplify the ULDA implementation in [17]. Here, ULDA/QR denotes ULDA based on QR-decomposition [11].

Classical LDA and classical ULDA do not address the singularity problem, hence it is difficult to apply them to undersampled data. Such high-dimensional, undersampled problems frequently occur in many applications including information retrieval [15], face recognition [25], and microarray analysis [7]. Several schemes have been proposed to address the singularity problem in classical LDA in the past, including pseudoinverse-based LDA [29], the subspace-based method [25], regularization [9], and the method based on the Generalized Singular Value Decomposition, called LDA/GSVD [15], [16]. Pseudoinverse-based LDA applies the pseudoinverse [11] to deal with the singularity problem. The subspace-based method applies the Karhunen-Loeve (KL) expansion, also known as Principal Component Analysis (PCA) [19], before LDA. Its limitation is that some useful information may be lost in the KL expansion. Regularized LDA overcomes the singularity problem by

1. Two variables x and y are said to be uncorrelated, if their covariance is zero, i.e., $\mathrm{cov}(x, y) = 0$.

- *J. Ye is with the Department of Computer Science and Engineering, Arizona State University, 699 South Mill Avenue, Tempe, AZ 85287. E-mail: jieping.ye@asu.edu.*
- *R. Janardan is with the Department of Computer Science and Engineering, University of Minnesota—Twin Cities, 4-192 EE/CSci. Bldg., 200 Union Street S.E., Minneapolis, MN 55455. E-mail: janardan@cs.umn.edu.*
- *Q. Li is with the Department of Computer and Information Sciences, University of Delaware, 103 Smith Hall, Newark, DE 19716. E-mail: qili@cis.udel.edu.*
- *H. Park is with the College of Computing, Georgia Institute of Technology, 801 Atlantic Drive, Atlanta, GA 30332. E-mail: hpark@cc.gatech.edu.*

Manuscript received 1 Apr. 2005; revised 30 Nov. 2005; accepted 30 May 2006; published online 18 Aug. 2006.
For information on obtaining reprints of this article, please send e-mail to: tkde@computer.org, and reference IEEECS Log Number TKDE-0126-0405.

TABLE 1
Summary of Notations Used

Notations	Descriptions	Notations	Descriptions
A	data matrix	n	number of training data points
N	dimension of the training data	ℓ	number of reduced dimensions
k	number of classes	S_b	between-class scatter matrix
S_w	within-class scatter matrix	S	total scatter matrix
G	transformation matrix	K	number of nearest neighbors in K-NN
A_i	data matrix of the i-th class	S_i	covariance matrix of the i-th class
c_i	mean of data in the i-th class	n_i	sample size of the i-th class
c	total mean of the training data	t	rank of the matrix S

increasing the magnitude of the diagonal elements of the scatter matrices (usually by adding a scaled identity matrix). The difficulty in using regularized LDA for feature reduction is the choice of the amount of perturbation. A small perturbation is desirable to preserve the original matrix structure, while a large perturbation is more effective in dealing with the singularity problem.

There is much less work on addressing the singularity problem in classical ULDA than on classical LDA. In the subspace ULDA presented in [17], a subspace-based method was applied (PCA is applied to the between-class scatter matrix). We address the singularity problem in ULDA, in the second part of this paper, by introducing a novel optimization criterion that combines the key ingredients of ULDA/QR and regularized LDA. The criterion is the perturbed version of the criterion used in ULDA/QR. Based on this criterion and the Generalized Singular Value Decomposition (GSVD) [21], we propose a novel feature reduction algorithm, called ULDA/GSVD. ULDA/GSVD solves the singularity problem directly, thus avoiding the information loss that occurs in the subspace method. Since the GSVD computation can be expensive for large and high-dimensional data sets, an efficient algorithm for ULDA/GSVD is also proposed. The difference between ULDA/GSVD and the traditional regularized LDA is that the optimal discriminant feature vectors via ULDA/GSVD are independent of the value of regularization parameter. This is quite a surprising result and the proof and the details are given in Section 5.

With the K-Nearest-Neighbor (K-NN) classifier, we evaluate the effectiveness of ULDA/GSVD and compare it with several other commonly used feature reduction algorithms, including Orthogonal Centroid Method (OCM) [22], PCA [19], and subspace ULDA [17], on various types of data sets, including text documents, chemical analysis of wine, face images, and microarray gene expression data. The experimental results show that the ULDA/GSVD algorithm is competitive with the other feature reduction algorithms (i.e., PCA, OCM, and subspace ULDA) and Support Vector Machines (SVM) [27]. Results also show that ULDA/GSVD is stable under different K-NN classifiers.

The rest of the paper is organized as follows: Sections 2 and 3 give brief reviews on classical LDA and classical ULDA, respectively. The ULDA/QR algorithm is presented in Section 4. Section 5 proposes the ULDA/GSVD algorithm, based on a novel criterion that is the regularized version of the criterion used in ULDA/QR. We prove theoretically that the solution to ULDA/GSVD is independent of the value of regularization applied. Experimental results are presented in Section 6. We conclude in Section 7. For convenience, the important notations used in this paper are listed in Table 1.

2 Classical Linear Discriminant Analysis

Given a data matrix $A = (a_{ij}) \in \mathbb{R}^{N\times n}$, where each column corresponds to a data point and each row corresponds to a particular feature, we consider finding a linear transformation $G \in \mathbb{R}^{N\times \ell}$ $(\ell < N)$ that maps each column a_i, for $1 \le i \le n$, of A in the N-dimensional space to a vector y_i in the ℓ-dimensional space as follows:

$$G : a_i \in \mathbb{R}^N \rightarrow y_i = G^T a_i \in \mathbb{R}^\ell.$$

The resulting data matrix $Z = G^T A \in \mathbb{R}^{\ell\times n}$ contains ℓ rows, i.e., there are ℓ features for each data point in the dimension reduced (transformed) space. It is also clear that the features in the dimension reduced space are linear combinations of the features in the original high-dimensional space, where the coefficients of the linear combinations depend on the transformation matrix G. A common way to compute the transformation matrix G, for clustered data sets, is through classical LDA. It computes the optimal transformation matrix G such that the class structure is preserved. More details are given below.

Assume that there are k classes in the data set. Suppose c_i is the mean vector of the ith class and c is the total mean. Then, the *between-class scatter matrix* S_b, the *within-class scatter matrix* S_w, and the *total scatter matrix* S are defined as follows [10]: $S_w = H_w H_w^T$, $S_b = H_b H_b^T$, and $S = H_t H_t^T$, where

$$H_w = \frac{1}{\sqrt{n}}[A_1, \cdots, A_k], \tag{1}$$

$$H_b = \frac{1}{\sqrt{n}}[\sqrt{n_1}(c_1 - c), \cdots, \sqrt{n_k}(c_k - c)], \tag{2}$$

$$H_t = \frac{1}{\sqrt{n}}\left(A - ce^T\right), \tag{3}$$

A_i is the data matrix of the ith class, n_i is the sample size of the ith class, and $e \in \mathbb{R}^n$ is a vector of ones.

The *trace* of the two scatter matrices can be computed as follows:

$$\mathrm{trace}(S_w) = \frac{1}{n}\sum_{i=1}^{k}||A_i||_F^2,$$

$$\mathrm{trace}(S_b) = \frac{1}{n}\sum_{i=1}^{k} n_i||c_i - c||^2,$$

where $||\cdot||_F$ denotes the Frobenius norm [11]. Hence, $\mathrm{trace}(S_w)$ measures the between-class cohesion, and $\mathrm{trace}(S_b)$ measures the between-class separation. It follows from the definition that $S_t = S_w + S_b$. In the lower-dimensional space resulting from the linear transformation G, the within-class scatter and between-class scatter matrices become

$$S_w^L = (G^T H_w)(G^T H_w)^T = G^T S_w G,$$
$$S_b^L = (G^T H_b)(G^T H_b)^T = G^T S_b G.$$

An optimal transformation G would maximize $\mathrm{trace}(S_b^L)$ and minimize $\mathrm{trace}(S_w^L)$ simultaneously. Classical LDA aims to compute the optimal G, such that

$$G = \arg\max_G \mathrm{trace}\Big(\big(G^T S_w G\big)^{-1} G^T S_b G\Big). \tag{4}$$

Other optimization criteria, including those based on the determinant, could also be used instead [6], [10]. The solution to the optimization problem in (4) can be obtained by solving an eigenvalue problem on $S_w^{-1}S_b$ [10], provided that the within-class scatter matrix S_w is nonsingular. Since the rank of the between-class scatter matrix is bounded from above by $k-1$, there are at most $k-1$ discriminant vectors by classical LDA. A stable way to solve this eigenvalue problem is to apply SVD on the scatter matrices. Details can be found in [25].

Classical LDA is equivalent to maximum likelihood classification assuming normal distribution for each class with the common covariance matrix. Although relying on assumptions which do not hold in many applications, LDA has been proven to be effective. This is mainly due to the fact that a simple, linear model is more robust against noise, and most likely will not overfit. Generalization of LDA by fitting Gaussian mixtures to each class has been studied in [13].

Classical LDA cannot handle singular scatter matrices, which limits its applicability to low-dimensional data. Several methods, including pseudoinverse-based LDA [29], subspace LDA [25], regularized LDA [9], LDA/GSVD [15], [16], and Penalized LDA [12], were proposed in the past to deal with the singularity problem. More details can be found in [20], [28].

In pseudoinverse-based LDA, the pseudoinverse is applied to avoid the singularity problem, which is equivalent to approximating the solution using a least-squares method. In subspace LDA, an intermediate dimension reduction algorithm, such as PCA, is applied to reduce the dimension of the original data, before classical LDA is applied. A limitation of this approach is that the optimal value of the reduced dimension for the intermediate dimension reduction algorithm is difficult to determine. In regularized LDA, a positive constant μ is added to the diagonal elements of S_w, as $S_w + \mu I_N$, where I_N is an identity matrix. The matrix $S_w + \mu I_N$ is positive definite, for any $\mu > 0$, hence nonsingular. A limitation of this approach is that the optimal value of the parameter μ is difficult to determine. Cross validation is commonly applied to estimate the optimal μ.

3 Uncorrelated Linear Discriminant Analysis (ULDA)

ULDA aims to find the optimal discriminant vectors that are S-orthogonal.[2] Specifically, suppose r vectors $\phi_1, \phi_2, \cdots, \phi_r$ are obtained, then the $(r+1)$th vector ϕ_{r+1} is found to maximize the Fisher criterion function [17]:

$$f(\phi) = \frac{\phi^T S_b \phi}{\phi^T S_w \phi},$$

subject to the constraints: $\phi_{r+1}^T S \phi_i = 0$, for $i = 1, \cdots, r$.

The algorithm in [17] finds ϕ_i successively as follows: The jth discriminant vector ϕ_j of ULDA is the eigenvector corresponding to the maximum eigenvalue of the following generalized eigenvalue problem: $U_j S_b \phi_j = \lambda_j S_w \phi_j$, where

$$U_1 = I_N,$$
$$D_j = [\phi_1, \cdots, \phi_{j-1}]^T \quad (j > 1),$$
$$U_j = I_N - S D_j^T (D_j S S_w^{-1} S D_j^T)^{-1} D_j S S_w^{-1} \quad (j > 1),$$

and I_N is the identity matrix.

Assume that $\{\phi_i\}_{i=1}^d$ are the d optimal discriminant vectors for the above ULDA formulation. Then, the original data matrix A is transformed into $Z = G^T A$, where $G = [\phi_1, \cdots, \phi_d]$. The ith feature component of Z is $z_i = \phi_i^T A$, and the covariance between z_i and z_j is

$$\begin{aligned}\mathrm{Cov}(z_i, z_j) &= E(z_i - Ez_i)(z_j - Ez_j) \\ &= \phi_i^T \{E(A - EA)(A - EA)^T\}\phi_j \\ &= \phi_i^T S \phi_j.\end{aligned} \tag{5}$$

Hence, their correlation coefficient is

$$\mathrm{Cor}(Z_i, Z_j) = \frac{\phi_i^T S \phi_j}{\sqrt{\phi_i^T S \phi_i}\sqrt{\phi_j^T S \phi_j}}. \tag{6}$$

Since the discriminant vectors of ULDA are S-orthogonal, i.e., $\phi_i^T S \phi_j = 0$, for $i \neq j$, we have $\mathrm{Cor}(Z_i, Z_j) = 0$, for $i \neq j$. That is, the feature vectors transformed by ULDA are mutually uncorrelated. This is a desirable property for feature reduction. More details on the role of uncorrelated attributes can be found in [17]. The limitation of the above ULDA algorithm is the expensive computation of the d generalized eigenvalue problems, where d is number of optimal discriminant vectors of ULDA.

In the literature for LDA, Foley-Sammon Linear Discriminant Analysis (FSLDA), which was proposed by Foley and Sammon for two-class problems [8], has also received attention. It was then extended to the multiclass problems by Duchene and Leclerq [5]. Both ULDA and FSLDA use the same Fisher criterion function. The main difference is that the optimal discriminant vectors generated by ULDA are S-orthogonal to each other, while the optimal discriminant vectors by FSLDA are orthogonal to each other.

2. Two vectors x and y are S-orthogonal, if $x^T S y = 0$.

4 THE ULDA/QR ALGORITHM

In this section, we first show the equivalence relationship between classical ULDA and a variant of classical LDA, which holds regardless of the distribution of the eigenvalues of $S_w^{-1}S_b$. This result enhances the one in [18] where the equivalence between these two is based on the assumption that there are no multiple eigenvalues for $S_w^{-1}S_b$ (note that both results assume that the within-class scatter matrix S_w is nonsingular). Based on this equivalence relationship, we propose ULDA/QR to simplify the ULDA implementation in [17].

Consider a variant of classical LDA in (4) as follows:

$$G = \arg \max_{G^T S G = I_\ell} F(G), \tag{7}$$

where

$$F(G) = \mathrm{trace}\Big(\big(G^T S_w G\big)^{-1} G^T S_b G \Big). \tag{8}$$

The use of the total scatter S in discriminant analysis has been discussed in [3]. Note that the ULDA algorithm discussed in the previous section finds the discriminant vectors in G successively. However, in the new formulation above, we compute all discriminant vectors simultaneously. S-orthogonality is enforced as a constraint. Our main result in this section, summarized in Theorem 2, shows that these two formulations for ULDA are equivalent.

The main technique for solving the optimization problem in (7) is the simultaneous diagonalization of the within-class and between-class scatter matrices. It is well-known that, for a symmetric positive definite matrix S_w and a symmetric matrix S_b, there exists a nonsingular matrix X such that

$$X^T S_w X = I_N, \tag{9}$$

$$X^T S_b X = \Lambda = \mathrm{diag}(\lambda_1, \cdots, \lambda_N), \tag{10}$$

where $\lambda_1 \geq \cdots \geq \lambda_N$ [11]. The matrix X can be computed efficiently based on the QR-decomposition as follows: Let $H_w^T = QR$ be the QR-decomposition of H_w^T, where H_w is defined in (1), $Q \in \mathbb{R}^{n \times N}$ has orthonormal columns and $R \in \mathbb{R}^{N \times N}$ is upper triangular and nonsingular. Then, $S_w = H_w H_w^T = R^T R$ and $(R^{-1})^T S_w R^{-1} = I_N$. That is, R^{-1} diagonalizes the within-class scatter matrix S_w. Next, consider the matrix

$$(R^{-1})^T S_b R^{-1} = \big(H_b^T R^{-1}\big)^T \big(H_b^T R^{-1}\big) \equiv Y^T Y,$$

where $Y = H_b^T R^{-1}$.

Let $Y = U \Sigma V^T$ be the SVD of Y, where $U \in \mathbb{R}^{n \times q}$, $\Sigma = \mathrm{diag}(\sigma_1, \cdots, \sigma_q) \in \mathbb{R}^{q \times q}$, $V \in \mathbb{R}^{N \times q}$, $\sigma_1 \geq \cdots \geq \sigma_q$, and $q = \mathrm{rank}(H_b)$. It is easy to check that $X = R^{-1}V$ diagonalizes both S_w and S_b and satisfies the conditions in (9) and (10).

It can be shown that the matrix consisting of the first q columns of X computed above (with normalization) solves the optimization problem in (7), where q is the rank of the matrix S_b, as stated in the following theorem:

Theorem 1. *Let the matrix X be defined as in (9) and (10), and $q = \mathrm{rank}(S_b)$. Let $G^* = [\tilde{x}_1, \cdots, \tilde{x}_q]$, where $\tilde{x}_i = \frac{1}{\sqrt{1+\lambda_i}} x_i$, x_i is the ith column of the matrix X, and λ_is are defined in (10). Then, G^* solves the optimization problem in (7).*

Proof. It is clear that the constraint in (7) is satisfied for $G = G^*$. Next, we only need to show that the maximum of $F(G)$ is obtained at G^*. By (9) and (10), we have

$$G^T S_w G = G^T X^{-T} (X^T S_w X) X^{-1} G = \tilde{G} \tilde{G}^T,$$
$$G^T S_b G = G^T X^{-T} (X^T S_b X) X^{-1} G = \tilde{G} \Lambda \tilde{G}^T,$$

where $\tilde{G} = (X^{-1}G)^T$. Hence,

$$F(G) = \mathrm{trace}\Big(\big(\tilde{G}\tilde{G}^T\big)^{-1} \big(\tilde{G}\Lambda\tilde{G}^T\big) \Big).$$

Let $\tilde{G}^T = QR$ be the QR-decomposition of $\tilde{G}^T \in \mathbb{R}^{N \times \ell}$ (note that $\tilde{G}^T$ has full column rank), where $Q \in \mathbb{R}^{N \times \ell}$ has orthonormal columns and R is nonsingular. Using the fact that $\mathrm{trace}(AB) = \mathrm{trace}(BA)$, for any matrices A and B, we have

$$\begin{aligned} F(G) &= \mathrm{trace}\Big(\big(R^T R\big)^{-1} \big(R^T Q^T \Lambda Q R\big) \Big) \\ &= \mathrm{trace}\big(Q^T \Lambda Q\big) \leq \lambda_1 + \cdots + \lambda_q, \end{aligned}$$

where the inequality becomes an equality for

$$Q = \begin{pmatrix} I_\ell \\ 0 \end{pmatrix} \text{ or } G = X \begin{pmatrix} I_\ell \\ 0 \end{pmatrix} R,$$

when the reduced dimension $\ell = q$. Note that R is an arbitrary upper triangular and nonsingular matrix. Hence, G^* corresponds to the case when R is set to be

$$R = \mathrm{diag}\left(\frac{1}{\sqrt{1+\lambda_1}}, \cdots, \frac{1}{\sqrt{1+\lambda_q}} \right).$$

□

We are now ready to present our main result for this section:

Theorem 2. *Let $\tilde{x}_i$ be defined as in Theorem 1. Then, $\{\tilde{x}_i\}_{i=1}^q$ forms a set of optimal discriminant vectors for ULDA.*

Proof. By induction. It is trivial to check that $\tilde{x}_1 = \arg\max_\phi f(\phi)$, i.e., $\phi_1 = \tilde{x}_1$. Next, assume $\phi_i = \tilde{x}_i$, for $i = 1, \cdots, r$. We show in the following that $\phi_{r+1} = \tilde{x}_{r+1}$.

By the definition, $\phi_{r+1} = \arg\max_\phi f(\phi)$, subject to $\phi_{r+1}^T S \phi_i = 0$, for $i = 1, \cdots, r$. Let $\phi_{r+1} = \sum_{i=1}^N \gamma_i \tilde{x}_i$, since $\{\tilde{x}_i\}_{i=1}^N$ forms a base for $\mathbb{R}^N$. By the constraints $\phi_{r+1}^T S \phi_i = 0$, for $i = 1, \cdots, r$, we have $\gamma_i = 0$, for $i = 1, \cdots, r$, hence $\phi_{r+1} = \sum_{i=r+1}^N \gamma_i \tilde{x}_i$. It follows from (9) and (10) that

$$\begin{aligned} f(\phi_{r+1}) &= \frac{\Big(\sum_{i=r+1}^N \gamma_i \tilde{x}_i^T\Big) S_b \Big(\sum_{i=r+1}^N \gamma_i \tilde{x}_i\Big)}{\Big(\sum_{i=r+1}^N \gamma_i \tilde{x}_i^T\Big) S_w \Big(\sum_{i=r+1}^N \gamma_i \tilde{x}_i\Big)} \\ &= \frac{\sum_{i=r+1}^N \gamma_i^2 \lambda_i}{\sum_{i=r+1}^N \gamma_i^2} \leq \frac{\sum_{i=r+1}^N \gamma_i^2 \lambda_{r+1}}{\sum_{i=r+1}^m \gamma_i^2} \\ &= \lambda_{r+1}, \end{aligned}$$

where the inequality becomes an equality if $\gamma_i = 0$, for $i = r+2, \cdots, N$. Hence, $\tilde{x}_{r+1}$ can be chosen as the $(r+1)$th discriminant vector of ULDA, i.e., $\phi_{r+1} = \tilde{x}_{r+1}$. □

An efficient algorithm for computing $\{\tilde{x}_i\}_{i=1}^{q}$ through QR-decomposition is presented below as **Algorithm 1**.

Algorithm 1: The ULDA/QR Algorithm
Input: Data matrix A.
Output: Discriminant vectors $\tilde{x}_i$s of ULDA.
1. Construct matrices H_w and H_b as in (1) and (2).
2. Compute the QR-decomposition of H_w^T as $H_w^T = QR$, where $Q \in \mathbb{R}^{n\times N}$ and $R \in \mathbb{R}^{N\times N}$.
3. Form the matrix $Y \leftarrow H_b^T R^{-1}$.
4. Compute the SVD of Y as $Y = U\Sigma V^T$, where $U \in \mathbb{R}^{n\times q}$, $\Sigma = \mathrm{diag}(\sigma_1, \cdots, \sigma_q) \in \mathbb{R}^{q\times q}$, $V \in \mathbb{R}^{N\times q}$, $\sigma_1 \geq \cdots \geq \sigma_q$, and $q = \mathrm{rank}(H_b)$.
5. $[x_1, \cdots, x_q] \leftarrow R^{-1}V$.
6. $\lambda_i \leftarrow \sigma_i^2$, for $i = 1, \cdots, q$.
7. $\tilde{x}_i \leftarrow \frac{1}{\sqrt{1+\lambda_i}} x_i$, for $i = 1, \cdots, q$.

5 The ULDA/GSVD Algorithm

In the previous section, a variant of the classical LDA criterion was presented in (7). It was shown that the solution to the optimization problem in (7) forms optimal discriminant vectors for classical ULDA. Thus, it provides an efficient way for computing the optimal discriminant vectors for ULDA. However, the algorithm assumes the nonsingularity of S_w, which limits its applicability to low-dimensional data. In [17], a subspace-based method is presented to overcome the singularity problem, where the ULDA algorithm is preceded by PCA. However, the PCA stage may lose some useful information. In this section, we propose a new feature reduction algorithm, called ULDA/GSVD. The new criterion underlying ULDA/GSVD is motivated by the criterion in (7) and the regularized LDA. The new optimization problem for ULDA/GSVD is defined as follows:

$$G_\mu = \arg \max_{G^T S G = I_\ell} F_\mu(G), \tag{11}$$

where $F_\mu(G) = \mathrm{trace}\Big((G^T S_w G + \mu I_\ell)^{-1} G^T S_b G\Big)$. Note that matrix $G^T S_w G + \mu I_\ell$ is guaranteed to be nonsingular for $\mu > 0$.

Recall that a limitation of regularized LDA is that the optimal value of the perturbation μ is difficult to determine. A key difference between ULDA/GSVD and regularized LDA is that the optimal solution to ULDA/GSVD is independent of the regularization parameter, i.e., $G_{\mu_1} = G_{\mu_2}$, for any $\mu_1, \mu_2 > 0$. The main result of this section is summarized in the following theorem:

Theorem 3. *Let G_μ^*, for any $\mu > 0$, be the optimal solution to the optimization problem in (11). Then, the following equality holds:*

$$G_{\mu_1}^* = G_{\mu_2}^*, \text{for any } \mu_1, \mu_2 > 0. \tag{12}$$

To prove Theorem 3, we first show how to compute G_μ^*, for any $\mu > 0$. Recall that when the within-class scatter matrix is nonsingular, the optimal transformation can be computed by finding the matrix X, which simultaneously diagonalizes the scatter matrices. For this, the Generalized Singular Value Decomposition (GSVD) can be applied, even when both matrices are singular. A simple algorithm to compute GSVD can be found in [15], where the algorithm is based on [21].

The computation of G_μ^*, for any $\mu > 0$, is based on the following two lemmas:

Lemma 1. *Let S_w, S_b, and S be defined as in Section 2, and let $t = \mathrm{rank}(S)$. Then, there exists a nonsingular matrix $X \in \mathbb{R}^{N\times N}$, such that*

$$X^T S_b X = D_1 = \mathrm{diag}(\alpha_1^2, \cdots, \alpha_t^2, 0, \cdots, 0), \tag{13}$$

$$X^T S_w X = D_2 = \mathrm{diag}(\beta_1^2, \cdots, \beta_t^2, 0, \cdots, 0), \tag{14}$$

where

$$1 \geq \alpha_1 \geq \cdots \geq \alpha_q > 0 = \alpha_{q+1} = \cdots = \alpha_t,$$
$$0 \leq \beta_1 \leq \cdots \leq \beta_t \leq 1,$$
$$D_1 + D_2 = \begin{pmatrix} I_t & 0 \\ 0 & 0 \end{pmatrix},$$

and $q = \mathrm{rank}(S_b)$.

Proof. Let

$$K = \begin{bmatrix} H_b^T \\ H_w^T \end{bmatrix},$$

which is an $(n+k)\times N$ matrix. By the Generalized Singular Value Decomposition [21], there exist orthogonal matrices $U \in \mathbb{R}^{k\times k}$, $V \in \mathbb{R}^{n\times n}$, and a nonsingular matrix $X \in \mathbb{R}^{N\times N}$, such that

$$\begin{bmatrix} U & 0 \\ 0 & V \end{bmatrix}^T K X = \begin{bmatrix} \Sigma_1 & 0 \\ \Sigma_2 & 0 \end{bmatrix}, \tag{15}$$

where

$$\Sigma_1^T \Sigma_1 = \mathrm{diag}(\alpha_1^2, \cdots, \alpha_t^2), \quad \Sigma_2^T \Sigma_2 = \mathrm{diag}(\beta_1^2, \cdots, \beta_t^2),$$
$$1 \geq \alpha_1 \geq \cdots \geq \alpha_q > 0 = \alpha_{q+1} = \cdots = \alpha_t,$$
$$0 \leq \beta_1 \leq \cdots \leq \beta_t \leq 1,$$

$\alpha_i^2 + \beta_i^2 = 1$, for $i = 1, \cdots, t$, and $q = \mathrm{rank}(H_b) = \mathrm{rank}(S_b)$.

Hence, $H_b^T X = U[\,\Sigma_1 \quad 0\,]$, and $H_w^T X = V[\,\Sigma_2 \quad 0\,]$. It follows that

$$X^T S_b X = X^T H_b H_b^T X = \begin{bmatrix} \Sigma_1^T \Sigma_1 & 0 \\ 0 & 0 \end{bmatrix} = D_1,$$

$$X^T S_w X = X^T H_w H_w^T X = \begin{bmatrix} \Sigma_2^T \Sigma_2 & 0 \\ 0 & 0 \end{bmatrix} = D_2,$$

where $D_1 + D_2 = \begin{pmatrix} I_t & 0 \\ 0 & 0 \end{pmatrix}$. □

Lemma 2. *Define a trace optimization problem as follows:*

$$G = \arg \max_{G^T G = I_\ell} \mathrm{trace}\Big(\big(G^T W G\big)^{-1} G^T B G\Big), \tag{16}$$

where $W = \mathrm{diag}(w_1, \cdots, w_u) \in \mathbb{R}^{u\times u}$ is a diagonal matrix with $0 < w_1 \leq \cdots \leq w_u$, and $B = \mathrm{diag}(b_1, \cdots, b_u) \in \mathbb{R}^{u\times u}$ is also diagonal with $b_1 \geq \cdots \geq b_q > 0 = b_{q+1} = \cdots = b_u$. Then, $G^\star = \big(I_q, 0\big)^T$ solves the optimization problem in (16) with $\ell = q$.

Proof. It is clear that the constraint in the optimization in (16) is satisfied for $G^\star$ with $\ell = q$. Next, we show that $G^\star$ solves the following optimization problem:

$$G = \arg\max_{G} \ \mathrm{trace}\Big(\big(G^T W G\big)^{-1} G^T B G\Big). \tag{17}$$

It is well-known that the solution can be obtained by solving the eigenvalue problem on $W^{-1}B$ since W is nonsingular. Note that $W^{-1}B$ is diagonal and only the first q diagonal entries are nonzero. Hence, e_i, for $i = 1, \cdots, q$, is the eigenvector of $W^{-1}B$ corresponding to the ith largest eigenvalue, where $e_i = (0, \cdots, 1, 0 \cdots, 0)^T$ and the one appears at the ith entry. Therefore, $G^\star = \big(I_q, 0\big)^T$ solves the optimization in (17). □

With Lemma 1 and Lemma 2, we can compute G^*_μ, for any $\mu > 0$, as follows:

Theorem 4. *Let the matrix X be defined as in Lemma 1, and let $q = \mathrm{rank}(S_b)$. Then,*

$$G^*_\mu = X \begin{pmatrix} I_q \\ 0 \end{pmatrix}$$

solves the optimization problem in (11) with $\ell = q$.

Proof. By Lemma 1, $X^T S_b X = D_1$, $X^T S_w X = D_2$, where the two diagonal matrices D_1 and D_2 satisfy

$$D_1 + D_2 = \begin{pmatrix} I_t & 0 \\ 0 & 0 \end{pmatrix}.$$

It is easy to check that

$$\begin{aligned}(G^*_\mu)^T S G^*_\mu &= \big(I_q, 0\big) X^T (S_b + S_w) X \begin{pmatrix} I_q \\ 0 \end{pmatrix} \\ &= \big(I_q, 0\big)(D_1 + D_2) \begin{pmatrix} I_q \\ 0 \end{pmatrix} = I_q,\end{aligned}$$

i.e., the constraint in the optimization problem in (11) is satisfied. Next, we show that G^*_μ minimizes $F_\mu(G)$. Since

$$\begin{aligned} G^T S_b G &= G^T (X^{-1})^T (X^T S_b X) X^{-1} G = \tilde{G} D_1 \tilde{G}^T, \\ G^T S_w G &= G^T (X^{-1})^T (X^T S_w X) X^{-1} G = \tilde{G} D_2 \tilde{G}^T, \end{aligned}$$

where $\tilde{G} = (X^{-1}G)^T$, $F_\mu(G)$ can then be rewritten as

$$F_\mu(G) = \mathrm{trace}\Big(\big(\tilde{G} D_2 \tilde{G}^T + \mu I_\ell\big)^{-1} \tilde{G} D_1 \tilde{G}^T\Big). \tag{18}$$

Let $\tilde{G} = \begin{pmatrix} G_1^T, & G_2^T \end{pmatrix}$ be a partition of $\tilde{G}$, such that $G_1^T \in \mathbb{R}^{\ell \times t}$ and $G_2^T \in \mathbb{R}^{\ell \times (N-t)}$. By the constraint that $G^T S G = I_\ell$, we have

$$\begin{aligned} I_\ell &= G^T S G = G^T (S_w + S_b) G = G^T S_b G + G^T S_w G \\ &= \tilde{G} D_1 \tilde{G}^T + \tilde{G} D_2 \tilde{G}^T = \tilde{G}(D_1 + D_2)\tilde{G}^T = G_1^T G_1. \end{aligned}$$

Hence, $F_\mu(G)$ in (18) can be rewritten as

$$F_\mu(G) = \mathrm{trace}\Big(\big(G_1^T (D_2^t + \mu I_\ell) G_1\big)^{-1} G_1^T D_1^t G_1\Big),$$

where D_1^t and D_2^t are the tth leading submatrices of D_1 and D_2, respectively. It is clear that $F_\mu(G)$ is independent of G_2. Hence, we can simplify set $G_2 = 0$. Denote $\Sigma = \big(D_2^t + \mu I_t\big)$, which is a nonsingular and diagonal matrix. It follows that

$$F_\mu(G) = \mathrm{trace}\Big(\big(G_1^T \Sigma G_1\big)^{-1} G_1^T D_1^t G_1\Big).$$

The result then follows from Lemma 2, with $W = \Sigma$ and $B = D_1^t$. □

Theorem 4 implies that the optimal solution $G^\star_\mu$ to the optimization problem in (11) only depends on X, which is determined by H_w and H_b, hence it is independent of μ. That is, $G^*_{\mu_1} = G^*_{\mu_2}$, for any $\mu_1, \mu_2 > 0$. This completes the proof of the main result of this section, which is summarized in Theorem 3.

The computation of the optimal transformation G^* is summarized in **Algorithm 2**.

Algorithm 2: The ULDA/GSVD Algorithm

Input: Data matrix A

Output: Optimal transformation matrix G^*

1. Form H_b and H_w as in (2) and (1).
2. Compute GSVD on the matrix pair (H_b^T, H_w^T) to obtain the matrix X, as in Lemma 1.
3. $q \leftarrow \mathrm{rank}(H_b)$.
4. $G^* \leftarrow [X_1, \cdots, X_q]$.

5.1 Efficient Computation of Diagonalizing Matrix X

In Lemma 1, a nonsingular matrix X is computed by applying GSVD, which may be expensive, especially for large matrices. A key property of X which leads to the optimal solution G^* is that it diagonalizes the scatter matrices simultaneously. In this section, we present an efficient algorithm for computing the diagonalizing matrix X without the GSVD computation.

Let $H_t = U \Sigma V^T$ be the SVD of H_t, where H_t is defined in (3), $U \in \mathbb{R}^{N \times N}$ and $V \in \mathbb{R}^{n \times n}$ are orthogonal, and $\Sigma \in \mathbb{R}^{N \times n}$ is diagonal. Then,

$$S = H_t H_t^T = U \Sigma V^T V \Sigma^T U^T = U \Sigma \Sigma^T U^T.$$

That is, the eigen-decomposition of S can be obtained by computing the SVD of H_t. Let $U = (U_1, U_2)$ be the partition of U, such that $U_1 \in \mathbb{R}^{N \times t}$ and $U_2 \in \mathbb{R}^{N \times (N-t)}$, where $t = \mathrm{rank}(S)$. Let $\Sigma \Sigma^T = \mathrm{diag}\big(\Sigma_t^2, 0\big)$, where $\Sigma_t \in \mathbb{R}^{t \times t}$ is diagonal and nonsingular. Since $S = S_b + S_w$, the null space, U_2, of S_t also lies in the null space of S_b and S_w, that is, $U_2^T S_b U_2 = 0$ and $U_2^T S_w U_2 = 0$. Hence,

$$\Sigma_t^2 = U_1^T S_b U_1 + U_1^T S_w U_1 \tag{19}$$

and

$$I_t = \Sigma_t^{-1} U_1^T S_b U_1 \Sigma_t^{-1} + \Sigma_t^{-1} U_1^T S_w U_1 \Sigma_t^{-1}. \tag{20}$$

Recall from (2) that $S_b = H_b H_b^T$. Denote $B = \Sigma_t^{-1} U_1^T H_b$ and let $B = P \tilde{\Sigma} Q^T$ be the SVD of B, where P and Q are orthogonal and $\tilde{\Sigma}$ is diagonal. Then,

$$\Sigma_t^{-1} U_1^T S_b U_1 \Sigma_t^{-1} = P \tilde{\Sigma} \tilde{\Sigma}^T P^T = P \Sigma_b P^T,$$

where $\Sigma_b = \tilde{\Sigma} \tilde{\Sigma}^T = \mathrm{diag}(\lambda_1, \cdots, \lambda_t)$,

$$\lambda_1 \geq \cdots \geq \lambda_q > 0 = \lambda_{q+1} = \cdots = \lambda_t,$$

and $q = \mathrm{rank}(S_b)$. It can be verified that the matrix X below diagonalizes the three scatter matrices simultaneously:

$$X = U \begin{pmatrix} \Sigma_t^{-1} P & 0 \\ 0 & I \end{pmatrix}. \tag{21}$$

The pseudocode for the computation of X is given in **Algorithm 3**.

Algorithm 3: Efficient computation of diagonalizing matrix X

Input: data matrix A
Output: matrix X
1. Form matrices H_b and H_t as in (2) and (3).
2. Compute SVD of H_t as $H_t = U_1 \Sigma_t V_1^T$.
4. $B \leftarrow \Sigma_t U_1^T H_b$.
5. Compute SVD of B as $B = P\tilde{\Sigma}Q^T$; $q \leftarrow \mathrm{rank}(B)$.
6. $X \leftarrow U \begin{pmatrix} \Sigma_t^{-1} P & 0 \\ 0 & I \end{pmatrix}$.

5.2 Relationship between ULDA/GSVD and ULDA/QR

In this section, we show that ULDA/GSVD is equivalent to ULDA/QR when the within-class scatter matrix S_w is nonsingular. Therefore, ULDA/QR can be considered as a special case of ULDA/GSVD when S_w is nonsingular. Note that ULDA/GSVD is more general in the sense that it is applicable regardless of the singularity of S_w.

Recall that ULDA/QR involves the matrix X, which satisfies

$$X^T S_w X = I_N,$$
$$X^T S_b X = \Lambda = \mathrm{diag}(\lambda_1, \cdots, \lambda_N),$$

where $\lambda_1 \geq \cdots \geq \lambda_N$.

The final transformation matrix $G^* = [\tilde{x}_1, \cdots, \tilde{x}_q]$, where $\tilde{x}_i = \frac{1}{\sqrt{1+\lambda_i}} x_i$, x_i is the ith column of the matrix X. It follows that

$$(G^*)^T S G^* = I_q, \tag{22}$$

$$(G^*)^T S_b G^* = \mathrm{diag}\left(\frac{\lambda_1}{1+\lambda_1}, \cdots, \frac{\lambda_q}{1+\lambda_q}\right). \tag{23}$$

Since $f(x) = x/(1+x)$ is an increasing function, we have

$$\frac{\lambda_1}{1+\lambda_1} \geq \cdots \geq \frac{\lambda_q}{1+\lambda_q}.$$

Thus, the transformation matrix G^* from ULDA/QR satisfies the conditions in Lemma 1 for ULDA/GSVD. That is, ULDA/GSVD is equivalent to ULDA/QR, when the within-class scatter matrix S_w is nonsingular. Note that ULDA/QR is not applicable when S_w is singular. ULDA/GSVD can thus be considered as an extension of ULDA/QR for a singular within-class scatter matrix. In the following experimental studies, we focus on the ULDA/GSVD algorithm.

We close this section by showing the classification property of ULDA/GSVD and ULDA/QR:

Theorem 5. *Let G be the optimal transformation matrix for ULDA/GSVD. Then, for any test point h, the following equality holds:*

$$\arg\min_j \left\{ (h - c_j)^T S^+ (h - c_j) \right\} = \arg\min_j \left\{ ||G^T(h - c_j)||^2 \right\}.$$

Proof. Let X_i be the ith column of X. Note that G consists of the first q columns of X, and $q = \mathrm{rank}(S_b)$. From (13) and (14), we have

$$S^+ = X(D_1 + D_2)X^T = GG^T + \sum_{i=q+1}^{t} X_i X_i^T.$$

From (13), $X_i^T S_b X_i = 0$, for $i = q+1, \cdots, t$. Hence, $(c_j)^T X_i = c X_i$, for all $j = 1, \cdots, k$. It follows that

$$(h - c_j)^T S^+ (h - c_j) = ||G^T(h - c_j)||^2 + \sum_{i=q+1}^{t} (h - c)^T X_i X_i^T (h - c). \tag{24}$$

The main result follows, since the second term on the right-hand side of (24) is independent of j. □

When S is nonsingular, the classification in ULDA/QR uses the Mahalanobis distance measure as follows:

Corollary 1. *Assume S is nonsingular. Let G be the optimal transformation matrix for ULDA/QR. Then, for any test point h, the following equality holds:*

$$\arg\min_j \left\{ (h - c_j)^T S^{-1} (h - c_j) \right\} = \arg\min_j \left\{ ||G^T(h - c_j)||^2 \right\}.$$

Corollary 1 shows that the classification in ULDA/QR is based on the Mahalanobis distance measure, while Theorem 5 shows that the classification in ULDA/GSVD is based on the modified Mahalanobis distance measure.

6 Experiments

We evaluate the effectiveness of the ULDA/GSVD algorithm in this section. Section 6.1 describes our test data sets. Section 6.2 examines the effect of the number of reduced dimensions on the classification performance of ULDA/GSVD. In Section 6.3, we compare ULDA/GSVD with PCA, OCM, and subspace ULDA, as well as SVM, in terms of classification accuracy. The K-Nearest-Neighbor (K-NN) algorithm with different values of K is used as the classifier.

6.1 Data Sets

We used two data sets: Spambase and Wine from the UCI Machine Learning Repository.[3] We used a subset of the original Spambase data set, which consists of spam and nonspam emails. Most of the features indicate whether a particular word or character occurred frequently in the e-mail. The Wine data set is the result of a chemical analysis of wines grown in the same region in Italy but derived from three different cultivars. The features correspond to the quantities of 13 different constituents found in each of the three types of wines. For these two data sets, the data dimension (N) is much smaller than the sample size (n). We also used six other data sets: GCM, ALL, tr41, re1, PIX, and ORL, where the data dimension is much larger than the sample size. In this case, ULDA/QR is not applicable, since all scatter matrices are singular, while ULDA/GSVD is still applicable. GCM [23], [30] and ALL [31] are microarray gene expression data sets; tr41 is a document data set

3. http://www.ics.uci.edu/mlearn/MLRepository.html.

TABLE 2
Statistics for the Test Data Sets

Dataset	Size (n)			Dimension (N)	Number of classes (k)
	training	test	total		
Spambase	400	600	1000	57	2
Wine	118	60	178	13	3
GCM	144	46	190	11485	14
ALL	163	85	248	12559	6
tr41	—	—	210	7454	7
re1	—	—	490	3759	5
PIX	—	—	300	10000	30
ORL	—	—	400	10304	40

("—" means that the natural splitting of the data set into training and test set is not available. For Spambase, Wine, GCM, and ALL, the original training and test sets are given, while for tr41, re1, PIX, and ORL, the original splitting is not provided.)

derived from the TREC-5, TREC-6, and TREC-7 collections;[4] re1 is another document data set derived from *Reuters-21578* text categorization test collection Distribution 1.0;[5] and ORL[6] and PIX [7] are two face image data sets.

Table 2 summarizes the statistics of our test data sets.

6.2 Effect of the Number of Reduced Dimensions on ULDA/GSVD

In this experiment, we study the effect of the number of reduced dimensions on the classification performance of ULDA/GSVD. The number of reduced dimensions ranges from 1 to 20. The classification results on the GCM and ALL data sets are shown in Fig. 1, where the horizontal axis is the number of reduced dimensions and the vertical axis is the classification accuracy. We can observe that the accuracy tends to increase when the number of reduced dimensions increases, until $q = \mathrm{rank}(H_b)$ (13 for GCM and 5 for ALL) is reached. Similar trends have been observed from other data sets, and the results are not presented. In the following experiment, we set the reduced dimension of ULDA/GSVD to be the rank of H_b.

6.3 Comparison of Classification Accuracy

In this experiment, we applied ULDA/GSVD to the eight data sets from Table 2 and compared with OCM, PCA, and subspace ULDA in terms of classification accuracy. The results are summarized in Table 3. The number of principal components used in PCA and Subspace ULDA is determined through cross-validation, and may be different for different data sets.

For data sets, including Spambase, Wine, GCM, and ALL, the training and test sets given in the original data sets are used for computing the accuracy. For the other four data sets, including tr41, re1, PIX, and ORL, where the training and test sets are not given, we performed our study by repeated random splitting into training and test sets exactly as in [7]. The data was partitioned randomly into a training set consisting of two-thirds of the whole set and a test set consisting of one-third of the whole set. To reduce the variability, the splitting was repeated 50 times and the resulting accuracies were averaged. The standard deviation for each data set was also reported.

4. http://trec.nist.gov.
5. http://www.research.att.com/~lewis.
6. http://www.uk.research.att.com/facedatabase.html.
7. http://peipa.essex.ac.uk/ipa/pix/faces/manchester/test-hard/.

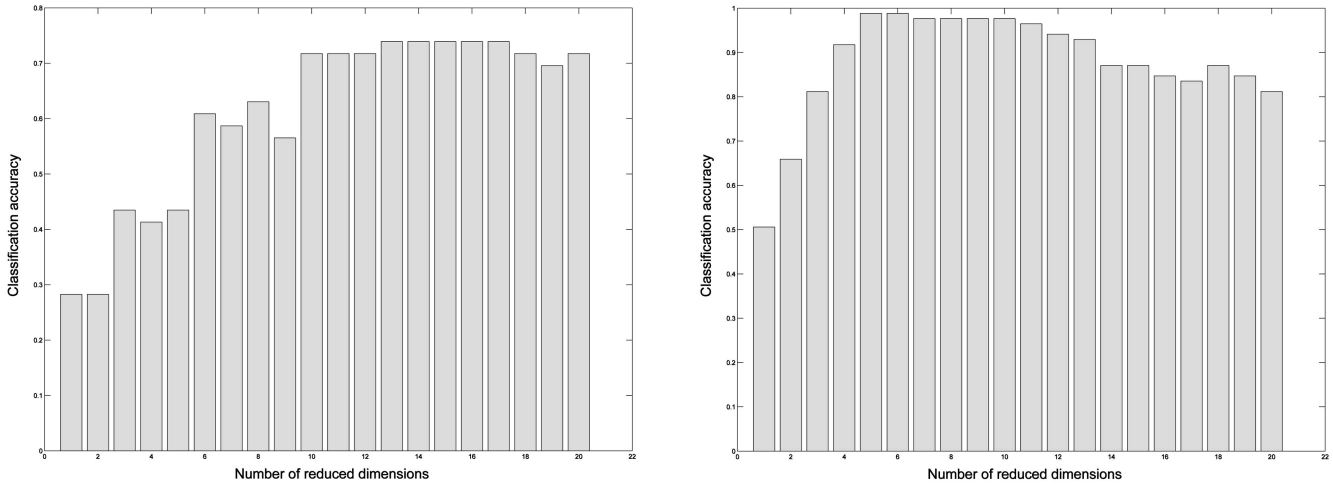
The main observations from Table 3 include: 1) ULDA/GSVD is competitive with the other three algorithms for all data sets in terms of classification. Subspace ULDA performs well for most data sets. However, subspace ULDA applies cross-validation for determining the optimal set of principal components in the PCA step, which can be expensive, especially for large data sets. Besides, the variance of the results for the other three methods is generally larger than that of ULDA/GSVD. This implies that ULDA/GSVD provides a more consistent result. 2) ULDA/GSVD is extremely stable under different K-NN classifiers for all data sets, whereas the performance of OCM and PCA degrades for many cases, as the number, K, of nearest neighbors increases. Subspace ULDA is also stable under different K-NN classifiers for most data sets. 3) PCA does not perform well in many cases. This is likely related to the fact that PCA is unsupervised and does not use the class label information, while the other three algorithms fully utilize the class label information. OCM performs well for the two document data sets and the two face image data sets, while it performs poorly for the other data sets. Both PCA and OCM perform poorly in Spambase and Wine, in comparison with ULDA/GSVD and subspace ULDA.

We have also done some preliminary studies in comparing ULDA/GSVD with linear SVM. 1NN is used to compute the accuracy for ULDA/GSVD. The main result is summarized in Fig. 2, where the x-axis denotes the eight data sets, and the y-axis denotes the classification accuracy. For tr41, re1, PIX, and ORL, the mean accuracy for 50 different runs are reported. Overall, ULDA/GSVD and linear SVM are comparable in terms of classification.

7 Conclusion

Uncorrelated features with minimum redundancy are highly desirable in feature reduction. In this paper, we present a theoretical and empirical study on uncorrelated Linear Discriminant Analysis (ULDA). We first present the theoretical result on the equivalence relationship between classical ULDA and classical LDA, which leads to a fast implementation of ULDA, ULDA/QR. Then, we propose ULDA/GSVD, based on a novel optimization criterion, that can successfully overcome the singularity problem in classical ULDA. The criterion used in ULDA/GSVD is the perturbed version of the one from ULDA/QR, while the solution to ULDA/GSVD is shown to be independent of the amount of perturbation applied, thus avoiding the limitation in regularized LDA. Experimental results on various types of data show the superiority of ULDA/GSVD over other competing algorithms including PCA, OCM, and subspace ULDA.

Experimental results show that ULDA/GSVD is extremely stable under different K-NN classifiers for all data sets. We plan to carry out detailed theoretical analysis on this in the future. The current work focuses on linear discriminant analysis, which applies a linear decision boundary. Discriminant analysis can also be studied in a nonlinear fashion—so-called kernel discriminant analysis—by using the kernel trick [24]. This is desirable if the data



(a) (b)

Fig. 1. Effect of the number of reduced dimensions on the classification performance of ULDA/GSVD for (a) the GCM and (b) ALL data sets. The optimal numbers of reduced dimensions for GCM and ALL are 13 and 5, respectively.

TABLE 3
Comparison of Classification Accuracy on Four Different Methods

Dataset	Method	1NN	3NN	5NN	10NN
Spambase	ULDA	88.17%	89.00%	89.67%	89.50%
	Subspace	85.67%	87.67%	85.67%	85.50%
	OCM	59.83%	59.00%	65.17%	69.83%
	PCA	66.50%	64.00%	67.83%	71.00%
Wine	ULDA	96.67%	96.67%	98.33%	98.33%
	Subspace	96.67%	96.67%	98.33%	95.00%
	OCM	68.33%	75.00%	68.33%	65.00%
	PCA	70.00%	75.00%	68.33%	65.00%
GCM	ULDA	73.91%	73.91%	73.91%	73.91%
	Subspace	73.91%	67.39%	65.22%	71.74%
	OCM	58.70%	56.52%	52.17%	47.83%
	PCA	60.87%	56.52%	43.48%	43.48%
ALL	ULDA	98.82%	98.82%	98.82%	98.82%
	Subspace	95.29%	95.29%	95.29%	95.29%
	OCM	95.29%	95.29%	95.29%	95.29%
	PCA	96.47%	95.29%	95.29%	95.29%
tr41	ULDA	97.74% (1.47)	98.09% (1.46)	97.63% (1.72)	97.74% (1.92)
	Subspace	95.20% (2.50)	96.54% (2.06)	96.40% (1.78)	96.74% (2.14)
	OCM	96.14% (2.26)	96.34% (2.47)	95.57% (2.07)	95.94% (2.33)
	PCA	87.37% (2.82)	84.06% (4.73)	82.94% (3.87)	81.00% (4.64)
re1	ULDA	94.97% (1.62)	94.92% (1.54)	94.72% (1.73)	94.96% (1.43)
	Subspace	94.03% (1.70)	94.52% (1.37)	94.75% (1.85)	94.86% (1.54)
	OCM	93.19% (1.88)	94.31% (1.45)	94.36% (2.07)	94.60% (1.33)
	PCA	87.90% (2.57)	88.84% (2.17)	89.71% (2.37)	90.67% (2.32)
PIX	ULDA	96.76% (1.60)	96.64% (1.60)	96.47% (1.55)	96.71% (1.85)
	Subspace	95.84% (1.89)	95.87% (2.07)	96.04% (2.03)	95.40% (2.50)
	OCM	96.84% (1.76)	94.69% (1.97)	93.56% (2.30)	87.80% (2.66)
	PCA	97.16% (1.65)	93.78% (2.07)	91.78% (2.62)	83.51% (2.47)
ORL	ULDA	93.13% (2.00)	93.38%(2.08)	93.62% (2.41)	93.07% (2.06)
	Subspace	93.70% (2.25)	93.38%(1.94)	93.58% (2.30)	92.90% (2.21)
	OCM	96.57% (1.33)	93.45%(2.35)	90.70% (2.60)	82.08% (2.94)
	PCA	95.65% (1.51)	92.23%(2.32)	87.20% (2.52)	73.33% (2.79)

has weak linear separability. We plan to extend the current work to deal with the nonlinearity in the future.

ACKNOWLEDGMENTS

The authors would like to thank the four reviewers and the associate editor for their comments, which helped improve the paper significantly. The research of J. Ye and R. Janardan was sponsored, in part, by the Army High Performance Computing Research Center under the auspices of the Department of the Army, Army Research Laboratory cooperative agreement number DAAD19-01-2-0014, the content of which does not necessarily reflect the position or the policy of the government, and no official

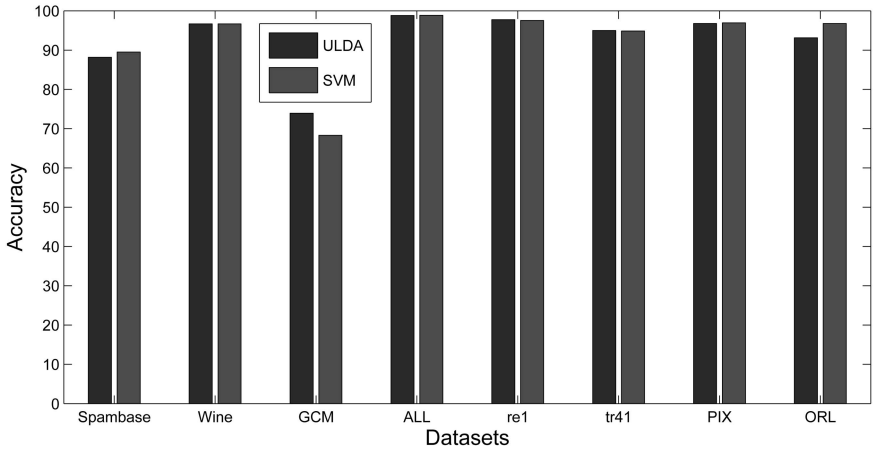


Fig. 2. Comparison of classification accuracy between ULDA/GSVD and SVM. For tr41, re1, PIX, and ORL, the mean accuracy for 50 different runs are reported.

endorsement should be inferred. Fellowships from Guidant Corporation and from the Department of Computer Science and Engineering, at the University of Minnesota, Twin Cities are gratefully acknowledged. The work of H. Park has been performed while serving as a program director at the US National Science Foundation (NSF) and was partly supported by IR/D from the NSF. Her work was also supported in part by the US National Science Foundation Grants CCR-0204109 and ACI-0305543. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the US National Science Foundation.

REFERENCES

[1] P. Belhumeour, J. Hespanha, and D. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 19, no. 7, pp. 711-720, 1997.

[2] M. Berry, S. Dumais, and G. O'Brie, "Using Linear Algebra for Intelligent Information Retrieval," *SIAM Rev.,* vol. 37, pp. 573-595, 1995.

[3] L. Chen, H. Liao, M. Ko, J. Lin, and G. Yu, "A New LDA-Based Face Recognition System Which Can Solve the Small Sample Size Problem," *Pattern Recognition,* vol. 33, pp. 1713-1726, 2000.

[4] S. Deerwester, S. Dumais, G. Furnas, T. Landauer, and R. Harshman, "Indexing by Latent Semantic Analysis," *J. Soc. for Information Science,* vol. 41, pp. 391-407, 1990.

[5] L. Duchene and S. Leclerq, "An Optimal Transformation for Discriminant and Principal Component Analysis," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 10, no. 6, pp. 978-983, 1988.

[6] R. Duda, P. Hart, and D. Stork, *Pattern Classification.* Wiley, 2000.

[7] S. Dudoit, J. Fridlyand, and T.P. Speed, "Comparison of Discrimination Methods for the Classification of Tumors Using Gene Expression Data," *J. Am. Statistical Assoc.,* vol. 97, no. 457, pp. 77-87, 2002.

[8] D. Foley and J. Sammon, "An Optimal Set of Discriminant Vectors," *IEEE Trans. Computers,* vol. 24, no. 3, pp. 281-289, 1975.

[9] J. Friedman, "Regularized Discriminant Analysis," *J. Am. Statistical Assoc.,* vol. 84, no. 405, pp. 165-175, 1989.

[10] K. Fukunaga, *Introduction to Statistical Pattern Classification.* Academic Press, 1990.

[11] G.H. Golub and C.F. Van Loan, *Matrix Computations,* third ed. The Johns Hopkins Univ. Press, 1996.

[12] T. Hastie, A. Buja, and R. Tibshirani, "Penalized Discriminant Analysis," *Annals of Statistics,* vol. 23, pp. 73-102, 1995.

[13] T. Hastie and R. Tibshirani, "Discriminant Analysis by Gaussian Mixtures," *J. Royal Statistical Soc. series B,* vol. 58, pp. 158-176, 1996.

[14] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction.* Springer, 2001.

[15] 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 Analysis and Applications,* vol. 25, no. 1, pp. 165-179, 2003.

[16] P. Howland and H. Park, "Generalizing Discriminant Analysis Using the Generalized Singular Value Decomposition," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 26, no. 8, pp. 995-1006, Aug. 2004.

[17] Z. Jin, J.Y. Yang, Z.-S. Hu, and Z. Lou, "Face Recognition Based on the Uncorrelated Discriminant Transformation," *Pattern Recognition,* vol. 34, pp. 1405-1416, 2001.

[18] Z. Jin, J.-Y. Yang, Z.-M. Tang, and Z.-S. Hu, "A Theorem on the Uncorrelated Optimal Discriminant Vectors," *Pattern Recognition,* vol. 34, no. 10, pp. 2041-2047, 2001.

[19] I.T. Jolliffe, *Principal Component Analysis.* Springer-Verlag, 1986.

[20] W. Krzanowski, P. Jonathan, W. McCarthy, and M. Thomas, "Discriminant Analysis with Singular Covariance Matrices: Methods and Applications to Spectroscopic Data," *Applied Statistics,* vol. 44, pp. 101-115, 1995.

[21] C. Paige and M. Saunders, "Towards a Generalized Singular Value Decomposition," *SIAM J. Numerical Analysis,* vol. 18, pp. 398-405, 1981.

[22] H. Park, M. Jeon, and J. Rosen, "Lower Dimensional Representation of Text Data Based on Centroids and Least Squares," *BIT Numerical Math.,* vol. 43, no. 2, pp. 1-22, 2003.

[23] S. Ramaswamy et al., "Multiclass Cancer Diagnosis Using Tumor Gene Expression Signatures," *Proc. Nat'l Academy of Science,* vol. 98, no. 26, pp. 15149-15154, 2001.

[24] B. Schökopf and A. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization and Beyond.* MIT Press, 2002.

[25] D.L. Swets and J. Weng, "Using Discriminant Eigenfeatures for Image Retrieval," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 18, no. 8, pp. 831-836, Aug. 1996.

[26] M. Turk and A. Pentland, "Face Recognition Using Eigenfaces," *Proc. Computer Vision and Pattern Recognition Conf.,* pp. 586-591, 1991.

[27] V. Vapnik, *Statistical Learning Theory.* Wiley, 1998.

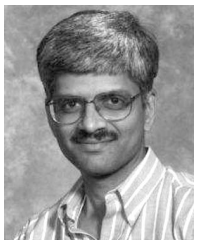
[28] J. Ye, "Characterization of a Family of Algorithms for Generalized Discriminant Analysis on Undersampled Problems," *J. Machine Learning Research,* vol. 6, pp. 483-502, 2005.

[29] J. Ye, R. Janardan, C. Park, and H. Park, "An Optimization Criterion for Generalized Discriminant Analysis on Undersampled Problems," *IEEE Trans. Pattern Analysis and Machine Intelligence,* vol. 26, no. 8, pp. 982-994, Aug. 2004.

[30] C.H. Yeang et al., "Molecular Classification of Multiple Tumor Types," *Bioinformatics,* vol. 17, no. 1, pp. 1-7, 2001.

[31] E.J. Yeoh et al., "Classification, Subtype Discovery, and Prediction of Outcome in Pediatric Lymphoblastic Leukemia by Gene Expression Profiling," *Cancer Cell,* vol. 1, no. 2, pp. 133-143, 2002.

Jieping Ye received the PhD degree in computer science from the University of Minnesota-Twin Cities in 2005. He is an assistant professor in the Department of Computer Science and Engineering at Arizona State University. He was awarded the Guidant Fellowship in 2004-2005. In 2004, his paper on generalized low rank approximations of matrices won the outstanding student paper award at the 21st International Conference on Machine Learning. His research interests include data mining, machine learning, and bioinformatics. He is a member of the IEEE and the ACM.



Ravi Janardan received the PhD degree in computer science from Purdue University in 1987. He is a professor in the Department of Computer Science and Engineering at the University of Minnesota-Twin Cities. His research interests are in the design and analysis of geometric algorithms and data structures and their application to problems in a variety of areas, including computer-aided design and manufacturing, computational biology and bioinformatics, and query retrieval in geometric databases. He has published extensively in these areas. He is a senior member of the IEEE and the IEEE Computer Society. He serves on the editorial board of the *Journal on Discrete Algorithms* and on the editorial advisory board of *Current Bioinformatics*.

Qi Li received the BS degree from the Department of Mathematics, Zhongshan University, China, in 1993, and the MS degree from the Department of Computer Science, University of Rochester, in 2002. He is currently a PhD candidate in the Department of Computer and Information Sciences, University of Delaware. His current research interests include pattern recognition, data mining, and machine learning. He is a student member of the IEEE.

Haesun Park received the BS degree in mathematics from Seoul National University, Seoul, Korea, in 1981 summa cum laude and with the university president's medal for the top graduate, and the MS and PhD degrees in computer science from Cornell University, Ithaca, New York, in 1985 and 1987, respectively. She was on the faculty of the Department of Computer Science and Engineering, University of Minnesota, Twin Cities, from 1987 to 2005. Since July 2005, she has been a professor in the College of Computing, Georgia Institute of Technology, Atlanta. Dr. Park has published more than 100 refereed journal and conference proceedings papers. Her current research interests include numerical algorithms, pattern recognition, data mining, information retrieval, and bioinformatics. She served on numerous conference committees and editorial boards of journals. Currently, she is on the editorial board of *BIT Numerical Mathematics*, the *SIAM Journal on Matrix Analysis and Applications*, and the *International Journal on Bioinformatics Research and Applications*. From 2003 to 2005, Dr. Park served as a program director for the Computing and Communication Foundations Division at the US National Science Foundation, Arlington, Virginia.