Generalizing Discriminant Analysis Using the Generalized Singular Value Decomposition

Peg Howland and Haesun Park

Abstract—Discriminant analysis has been used for decades to extract features that preserve class separability. It is commonly defined as an optimization problem involving covariance matrices that represent the scatter within and between clusters. The requirement that one of these matrices be nonsingular limits its application to data sets with certain relative dimensions. We examine a number of optimization criteria, and extend their applicability by using the generalized singular value decomposition to circumvent the nonsingularity requirement. The result is a generalization of discriminant analysis that can be applied even when the sample size is smaller than the dimension of the sample data. We use classification results from the reduced representation to compare the effectiveness of this approach with some alternatives, and conclude with a discussion of their relative merits.

Index Terms—Linear discriminant analysis, latent semantic indexing, principal component analysis, generalized singular value decomposition, QR decomposition, trace optimization.

1 Introduction

The goal of discriminant analysis is to combine features of the original data in a way that most effectively discriminates between classes. When combining features, the dimension of the data is reduced in a way that most effectively preserves its cluster structure. These linear combinations of the components of a data vector are traditionally expressed as inner products with the columns of a linear transformation G. That is, we want to find G whose transpose maps an m-dimensional data vector a to a vector y in the l-dimensional space ($l \ll m$):

$$G^T : a \in \mathbb{R}^{m\times 1} \rightarrow y \in \mathbb{R}^{l\times 1}.$$

Assuming that the given data are already clustered, we seek a transformation that optimally preserves this cluster structure in the reduced dimensional space.

For simplicity of discussion, we will assume that data vectors $a_1, \dots, a_n$ form columns of a matrix $A \in \mathbb{R}^{m\times n}$ and are grouped into k clusters (classes) as

$$A = (A_1, A_2, \dots, A_k) \text{ where } A_i \in \mathbb{R}^{m\times n_i} \text{ and } \sum_{i=1}^{k} n_i = n. \quad (1)$$

Let N_i denote the set of column indices that belong to cluster i. The centroid $c^{(i)}$ is computed by taking the average of the columns in cluster i, i.e.,

$$c^{(i)} = \frac{1}{n_i}\sum_{j\in N_i} a_j,$$

and the global centroid c is defined as

$$c = \frac{1}{n}\sum_{j=1}^{n} a_j.$$

Then, the within-cluster, between-cluster, and mixture scatter matrices are defined [5, Section 10.2] [18, Section 5.5] as

$$S_W = \sum_{i=1}^{k}\sum_{j\in N_i}(a_j - c^{(i)})(a_j - c^{(i)})^T,$$

$$S_B = \sum_{i=1}^{k}\sum_{j\in N_i}(c^{(i)} - c)(c^{(i)} - c)^T = \sum_{i=1}^{k} n_i (c^{(i)} - c)(c^{(i)} - c)^T, \text{ and}$$

$$S_M = \sum_{j=1}^{n}(a_j - c)(a_j - c)^T,$$

respectively. These $m \times m$ scatter matrices have the relationship

$$S_M = S_W + S_B, \quad (2)$$

which we will prove later, using the notation of (11)-(14). Applying G^T to the matrix A transforms the scatter matrices to the $l \times l$ matrices

$$S_W^Y = G^T S_W G, \quad S_B^Y = G^T S_B G, \quad \text{and} \quad S_M^Y = G^T S_M G,$$

where the superscript Y denotes values in the l-dimensional space.

There are several measures of cluster quality that involve the three scatter matrices [5, Section 10.2] [18, Section 5.5]. When cluster quality is high, each cluster is tightly grouped, but well separated from the other clusters. Since

$$\mathrm{trace}(S_W) = \sum_{i=1}^{k}\sum_{j\in N_i}(a_j - c^{(i)})^T(a_j - c^{(i)}) = \sum_{i=1}^{k}\sum_{j\in N_i}\|a_j - c^{(i)}\|_2^2$$

• *The authors are with the Department of Computer Science and Engineering, University of Minnesota, Minneapolis, MN 55455. E-mail: {howland, hpark}@cs.umn.edu.*

Manuscript received 10 Feb. 2003; revised 21 Aug. 2003; accepted 7 Oct. 2003.
Recommended for acceptance by T. Tan.
For information on obtaining reprints of this article, please send e-mail to: tpami@computer.org, and reference IEEECS Log Number 118264.

measures the closeness of the columns within the clusters and

$$\begin{aligned}\mathrm{trace}(S_B) &= \sum_{i=1}^{k}\sum_{j\in N_i}(c^{(i)}-c)^T(c^{(i)}-c)\\ &= \sum_{i=1}^{k}\sum_{j\in N_i}\|c^{(i)}-c\|_2^2\end{aligned}$$

measures the separation between clusters, an optimal transformation that preserves the given cluster structure would maximize $\mathrm{trace}(S_B^Y)$ and minimize $\mathrm{trace}(S_W^Y)$.

This simultaneous optimization is commonly approximated by finding a transformation G that maximizes $\mathrm{trace}((S_W^Y)^{-1}S_B^Y)$. Although this criterion is a less obvious choice than the quotient

$$\mathrm{trace}(G^T S_B G)/\mathrm{trace}(G^T S_W G),$$

it is formulated to be invariant under nonsingular linear transformations, a property that will prove useful below. However, this criterion cannot be applied when the matrix S_W is singular, a situation that occurs frequently in many applications. For example, in handling document data in information retrieval, it is often the case that the number of terms in the document collection is larger than the total number of documents (i.e., $m > n$ in the term-document matrix A) and, therefore, the matrix S_W is singular. Furthermore, in applications where the data points are in a very high dimensional space and collecting data is expensive, S_W is singular because the value for n must be kept relatively small.

One way to make classical discriminant analysis applicable to the data matrix $A \in \mathbb{R}^{m\times n}$ with $m > n$ (and, hence, S_W singular) is to perform dimension reduction in two stages. The discriminant analysis stage is preceded by a stage in which the cluster structure is ignored. The most popular method for the first part of this process is rank reduction by the singular value decomposition (SVD). This is the main tool in principal component analysis (PCA) [4], as well as in latent semantic indexing (LSI) [3], [2] of documents. Both Swets and Weng [17] and Belhumeur et al. [1] have utilized PCA plus LDA for facial feature extraction. More recently, Torkkola [19] implemented LSI plus LDA for document classification. However, the overall performance of these two-stage approaches will be sensitive to the reduced dimension in the first stage. Yang and Yang [21] supplied theoretical justification for PCA plus LDA, provided that the intermediate dimension after the first stage falls within a specific range. Depending on this intermediate dimension, S_W may remain singular, and classical discriminant analysis may not apply in the second stage. We discuss the two-stage approach in greater detail in Section 4.2.

In this paper, we extend discriminant analysis in a way that provides the optimal reduced dimension theoretically, without introducing another stage as described above. We consider the set of criteria involving

$$\mathrm{trace}((S_2^Y)^{-1}S_1^Y), \tag{3}$$

where S_1 and S_2 are chosen from S_W, S_B, and S_M. Classical discriminant analysis expresses their solution in terms of a generalized eigenvalue problem when S_2 is nonsingular. By reformulating the problem in terms of the generalized singular value decomposition (GSVD) [20], [14], [6, Section 8.7.3], we extend the applicability to the case when S_2 is singular. We also establish the equivalence among alternative choices for S_1 and S_2. In addition to the two-stage approach described above, we present a second alternative approach that optimizes the trace of an individual scatter matrix and show how this can be achieved efficiently. Finally, we present experimental results demonstrating the capabilities of the GSVD approach and comparing its effectiveness to the alternatives.

2 Generalized Singular Value Decomposition

The following theorem introduces the GSVD as it was originally defined by Van Loan [20].

Theorem 1. *Suppose two matrices $K_A \in \mathbb{R}^{p\times m}$ with $p \geq m$ and $K_B \in \mathbb{R}^{n\times m}$ are given. Then, there exist orthogonal matrices $U \in \mathbb{R}^{p\times p}$ and $V \in \mathbb{R}^{n\times n}$ and a nonsingular matrix $X \in \mathbb{R}^{m\times m}$ such that*

$$U^T K_A X = diag(\alpha_1, ..., \alpha_m) \text{ and } V^T K_B X = diag(\beta_1, ..., \beta_q),$$

where $q = \min(n, m)$, $\alpha_i \geq 0$ for $1 \leq i \leq m$, and $\beta_i \geq 0$ for $1 \leq i \leq q$.

This formulation cannot be applied to the matrix pair K_A and K_B when the dimensions of K_A do not satisfy the assumed restrictions. Paige and Saunders [14] developed a more general formulation which can be defined for any two matrices with the same number of columns. We restate theirs as follows.

Theorem 2. *Suppose two matrices $K_A \in \mathbb{R}^{p\times m}$ and $K_B \in \mathbb{R}^{n\times m}$ are given. Then, for*

$$K = \begin{pmatrix} K_A \\ K_B \end{pmatrix} \quad \text{and} \quad t = rank(K),$$

there exist orthogonal matrices $U \in \mathbb{R}^{p\times p}$, $V \in \mathbb{R}^{n\times n}$, $W \in \mathbb{R}^{t\times t}$, and $Q \in \mathbb{R}^{m\times m}$ such that

$$U^T K_A Q = \Sigma_A(\underbrace{W^T R}_{t}, \underbrace{0}_{m-t}) \text{ and } V^T K_B Q = \Sigma_B(\underbrace{W^T R}_{t}, \underbrace{0}_{m-t}),$$

where

$$\underset{p\times t}{\Sigma_A} = \begin{pmatrix} I_A & & \\ & D_A & \\ & & O_A \end{pmatrix}, \quad \underset{n\times t}{\Sigma_B} = \begin{pmatrix} O_B & & \\ & D_B & \\ & & I_B \end{pmatrix},$$

and $R \in \mathbb{R}^{t\times t}$ is nonsingular with its singular values equal to the nonzero singular values of K. The matrices

$$I_A \in \mathbb{R}^{r\times r} \quad \text{and} \quad I_B \in \mathbb{R}^{(t-r-s)\times(t-r-s)}$$

are identity matrices, where

$$r = rank\begin{pmatrix} K_A \\ K_B \end{pmatrix} - rank(K_B)$$

and

$$s = rank(K_A) + rank(K_B) - rank\begin{pmatrix} K_A \\ K_B \end{pmatrix},$$
$$O_A \in \mathbb{R}^{(p-r-s)\times(t-r-s)}, \quad and \quad O_B \in \mathbb{R}^{(n-t+r)\times r}$$

are zero matrices with possibly no rows or no columns, and

$D_A = diag(\alpha_{r+1}, \ldots, \alpha_{r+s})$ *and* $D_B = diag(\beta_{r+1}, \ldots, \beta_{r+s})$ *satisfy*

$$1 > \alpha_{r+1} \geq \cdots \geq \alpha_{r+s} > 0, \quad 0 < \beta_{r+1} \leq \cdots \leq \beta_{r+s} < 1, \quad (4)$$

and $\alpha_i^2 + \beta_i^2 = 1$ *for* $i = r+1, \ldots, r+s$.

This form of GSVD is related to that of Van Loan by writing [14]

$$U^T K_A X = (\Sigma_A, 0) \quad \text{and} \quad V^T K_B X = (\Sigma_B, 0), \qquad (5)$$

where

$$X_{m\times m} = Q\begin{pmatrix} R^{-1}W & 0 \\ 0 & I \end{pmatrix}.$$

From the form in (5), we see that

$$K_A = U(\Sigma_A, 0)X^{-1} \quad \text{and} \quad K_B = V(\Sigma_B, 0)X^{-1},$$

which imply that

$$K_A^T K_A = X^{-T}\begin{pmatrix} \Sigma_A^T\Sigma_A & 0 \\ 0 & 0 \end{pmatrix}X^{-1}$$

and

$$K_B^T K_B = X^{-T}\begin{pmatrix} \Sigma_B^T\Sigma_B & 0 \\ 0 & 0 \end{pmatrix}X^{-1}.$$

Defining

$$\alpha_i = 1, \beta_i = 0 \text{ for } i = 1, \ldots, r$$

and

$$\alpha_i = 0, \beta_i = 1 \text{ for } i = r+s+1, \ldots, t,$$

we have, for $1 \leq i \leq t$,

$$\beta_i^2 K_A^T K_A x_i = \alpha_i^2 K_B^T K_B x_i, \qquad (6)$$

where x_i represents the ith column of X. For the remaining $m-t$ columns of X, both $K_A^T K_A x_i$ and $K_B^T K_B x_i$ are zero, so (6) is satisfied for arbitrary values of α_i and β_i when $t+1 \leq i \leq m$. The columns of X are the generalized singular vectors for the matrix pair (K_A, K_B). In terms of the generalized singular values, or the α_i/β_i quotients, r of them are infinite, s are finite and nonzero, and $t-r-s$ are zero.

3 Generalization of Linear Discriminant Analysis

In this section, we utilize the GSVD to extend several criteria from discriminant analysis. We also establish the equivalence for various choices of scatter matrices.

3.1 Optimization of $\mathbf{J}_1 = \mathrm{trace}(\mathbf{S}_2^{-1}\mathbf{S}_1)$ Criteria

For now, we will focus our discussion on the criteria for optimizing

$$J_1(G) = \mathrm{trace}((G^T S_2 G)^{-1}(G^T S_1 G)) \qquad (7)$$

over G, where S_1 and S_2 are chosen from S_W, S_B, and S_M. When S_2 is assumed to be nonsingular, it is symmetric positive definite. According to results from the symmetric-definite generalized eigenvalue problem [6, Section 8.7], there exists a nonsingular matrix $X \in \mathbb{R}^{m\times m}$ such that

$$X^T S_1 X = \Lambda = \mathrm{diag}(\lambda_1 \ldots \lambda_m) \quad \text{and} \quad X^T S_2 X = I_m. \qquad (8)$$

Letting x_i denote the ith column of X, we have

$$S_1 x_i = \lambda_i S_2 x_i, \qquad (9)$$

which means that λ_i and x_i are an eigenvalue-eigenvector pair of $S_2^{-1}S_1$. Since S_1 is positive semidefinite, $\lambda_i \geq 0$ for $1 \leq i \leq m$. From (8), we see that only the largest $q = \mathrm{rank}(S_1)$ λ_is are nonzero. In addition, by using a permutation matrix to order Λ (and likewise X), we can assume that $\lambda_1 \geq \cdots \geq \lambda_q > \lambda_{q+1} = \cdots = \lambda_m = 0$.

We have

$$\begin{aligned} J_1(G) &= \mathrm{trace}((G^T S_2 G)^{-1} G^T S_1 G) \\ &= \mathrm{trace}((G^T X^{-T} X^{-1} G)^{-1} G^T X^{-T} \Lambda X^{-1} G) \\ &= \mathrm{trace}((\tilde{G}^T \tilde{G})^{-1} \tilde{G}^T \Lambda \tilde{G}), \end{aligned}$$

where $\tilde{G} = X^{-1}G$. The matrix $\tilde{G}$ has full column rank provided G does, so it has the reduced QR factorization $\tilde{G} = QR$, where $Q \in \mathbb{R}^{m\times l}$ has orthonormal columns and R is nonsingular [6, Section 5.2.6]. Hence,

$$\begin{aligned} J_1(G) &= \mathrm{trace}((R^T R)^{-1} R^T Q^T \Lambda Q R) \\ &= \mathrm{trace}(R^{-1} Q^T \Lambda Q R) \\ &= \mathrm{trace}(Q^T \Lambda Q R R^{-1}) \\ &= \mathrm{trace}(Q^T \Lambda Q). \end{aligned}$$

This shows that, once we have simultaneously diagonalized S_1 and S_2, the maximization of $J_1(G)$ depends only on an orthonormal basis for $\mathrm{range}(X^{-1}G)$, i.e.,

$$\begin{aligned} \max_G J_1(G) &= \max_{Q^T Q = I_l} \mathrm{trace}(Q^T \Lambda Q) \\ &\leq \lambda_1 + \cdots + \lambda_q \\ &= \mathrm{trace}(S_2^{-1} S_1). \end{aligned}$$

(Here, we consider only maximization. Similar arguments will hold when J_1 is *minimized* for some choices of S_1 and S_2.) For any l satisfying $l \geq q$, this upper bound on $J_1(G)$ is achieved for

$$Q = \begin{pmatrix} I_l \\ 0 \end{pmatrix} \quad \text{or} \quad G = X\begin{pmatrix} I_l \\ 0 \end{pmatrix}R.$$

Note that the transformation G is not unique. That is, J_1 satisfies the invariance property $J_1(G) = J_1(GW)$ for any nonsingular matrix $W \in \mathbb{R}^{l\times l}$ since

$$\begin{aligned} J_1(GW) &= \mathrm{trace}((W^T G^T S_2 G W)^{-1}(W^T G^T S_1 G W)) \\ &= \mathrm{trace}(W^{-1}(G^T S_2 G)^{-1} W^{-T} W^T (G^T S_1 G) W) \\ &= \mathrm{trace}((G^T S_2 G)^{-1}(G^T S_1 G) W W^{-1}) \\ &= J_1(G). \end{aligned}$$

Hence, the maximum $J_1(G)$ is also achieved for

$$G = X\begin{pmatrix} I_l \\ 0 \end{pmatrix}.$$

This means that, for $l \geq \mathrm{rank}(S_1)$,

$$\mathrm{trace}((G^T S_2 G)^{-1} G^T S_1 G) = \mathrm{trace}(S_2^{-1} S_1), \qquad (10)$$

whenever $G \in \mathbb{R}^{m\times l}$ consists of l eigenvectors of $S_2^{-1}S_1$ corresponding to the l largest eigenvalues.

Now, a limitation of the J_1 criteria in many applications, including information retrieval, is that the matrix S_2 must be nonsingular. Recalling the partitioning of A into k clusters given in (1), we define the $m \times n$ matrices

$$H_W = (A_1 - c^{(1)}e^{(1)^T}, A_2 - c^{(2)}e^{(2)^T}, \ldots, A_k - c^{(k)}e^{(k)^T}), \quad (11)$$

$$H_B = ((c^{(1)} - c)e^{(1)^T}, (c^{(2)} - c)e^{(2)^T}, \ldots, (c^{(k)} - c)e^{(k)^T}), \quad (12)$$

$$H_M = (a_1 - c, \ldots, a_n - c) = A - ce^T = H_W + H_B, \qquad (13)$$

where $e^{(i)} = (1, \ldots, 1)^T \in \mathbb{R}^{n_i \times 1}$ and $e = (1, \ldots, 1)^T \in \mathbb{R}^{n\times 1}$. Then, the scatter matrices can be expressed as

$$S_W = H_W H_W^T, \quad S_B = H_B H_B^T, \quad \text{and} \quad S_M = H_M H_M^T. \quad (14)$$

For S_2 to be nonsingular, we can only allow the case $m \leq n$ since S_2 is the product of an $m \times n$ matrix and an $n \times m$ matrix [13, Section 2.3]. Thus, J_1 cannot be applied when the number of available data vectors n is smaller than the dimension m of the data. We seek a solution which does not impose this restriction, and which (for numerical reasons explained in Section 3.3) can be found without explicitly forming S_1 and S_2 from H_W, H_B, and H_M. Toward these ends, we express λ_i as α_i^2/β_i^2 and the problem (9) generalizes to

$$\beta_i^2 S_1 x_i = \alpha_i^2 S_2 x_i. \qquad (15)$$

This has the form of a problem that can be solved using the GSVD, as described in Section 2.

3.2 Generalization of $\mathbf{J}_1 = \mathrm{trace}(\mathbf{S}_2^{-1}\mathbf{S}_1)$ Criteria for Singular $\mathbf{S}_2$

Continuing with the J_1 criteria, we first consider the case where

$$(S_1, S_2) = (S_B, S_W).$$

From (14) and the definition of H_B given in (12), $\mathrm{rank}(S_B) \leq k - 1$. To approximate G that satisfies both

$$\max_G \mathrm{trace}(G^T S_B G) \quad \text{and} \quad \min_G \mathrm{trace}(G^T S_W G), \qquad (16)$$

we use the GSVD to solve for the x_is in (15). For nonsingular S_W, the generalized singular vectors are eigenvectors of $S_W^{-1}S_B$, so we choose the x_is which correspond to the $k-1$ largest λ_is, where $\lambda_i = \alpha_i^2/\beta_i^2$. When the GSVD construction orders the singular value pairs as in (4), the generalized singular values, or the α_i/β_i quotients, are in nonincreasing order. Therefore, the first $k-1$ columns of X are all we need. The steps for this case are summarized in Algorithm 1, called LDA/GSVD and adapted from [7]. Our algorithm first computes the matrices H_B and H_W from the data matrix A. We then solve for a very limited portion of the GSVD of the matrix pair (H_B^T, H_W^T). This solution is accomplished by following the construction in the proof of Theorem 2 [14]. The major steps are limited to the complete orthogonal decomposition [6, Section 5.4.2], [12, Chapter 3] of

$$K = \begin{pmatrix} H_B^T \\ H_W^T \end{pmatrix},$$

which produces orthogonal matrices P and Q and a nonsingular matrix R, followed by the singular value decomposition of a leading principal submatrix of P whose size is much smaller than that of the data matrix. (This $k \times t$ submatrix is specified in Algorithm 1 (Fig. 1) using the colon notation of MATLAB.[1])

When $m > n$, the scatter matrix S_W is singular. Hence, the eigenvectors of $S_W^{-1}S_B$ are undefined, and classical discriminant analysis fails. Consider a generalized singular vector x_i that lies in the null space of S_W. From (15), we see that either x_i also lies in the null space of S_B or the corresponding β_i equals zero. We will discuss each of these cases separately.

When

$$x_i \in \mathrm{null}(S_W) \cap \mathrm{null}(S_B),$$

(15) is satisfied for arbitrary values of α_i and β_i. As explained in Section 2, this will be the case for the rightmost $m - t$ columns of X. To determine whether these columns should be included in G, consider

$$\mathrm{trace}(G^T S_B G) = \sum g_j^T S_B g_j$$

and

$$\mathrm{trace}(G^T S_W G) = \sum g_j^T S_W g_j,$$

where g_j represents the jth column of G. Since $x_i^T S_W x_i = 0$ and $x_i^T S_B x_i = 0$, adding the column x_i to G does not contribute to either maximization or minimization in (16). For this reason, we do not include these columns of X in our solution.

When

$$x_i \in \mathrm{null}(S_W) - \mathrm{null}(S_B),$$

then $\beta_i = 0$. As discussed in Section 2, this implies that $\alpha_i = 1$ and, hence, that the generalized singular value α_i/β_i is infinite. The leftmost columns of X will correspond to these. Including these columns in G increases $\mathrm{trace}(G^T S_B G)$, while leaving $\mathrm{trace}(G^T S_W G)$ unchanged. We conclude that, even when S_W is singular, the rule regarding which columns of X to include in G remains the same as for the nonsingular case. The experiments summarized in Section 5 demonstrate that Algorithm LDA/GSVD works very well even when S_W is singular, thus extending its applicability beyond that of classical discriminant analysis.

1. http://www.mathworks.com.

Algorithm 1 LDA/GSVD

Given a data matrix $A \in \mathbb{R}^{m\times n}$ with k clusters and an input vector $a \in \mathbb{R}^{m\times 1}$, compute the matrix $G \in \mathbb{R}^{m\times(k-1)}$ which preserves the cluster structure in the reduced dimensional space, using

$$J_1(G) = \text{trace}((G^T S_W G)^{-1} G^T S_B G).$$

Also compute the $k-1$ dimensional representation y of a.

1) Compute H_B and H_W from A according to

$$H_B = (\sqrt{n_1}(c^{(1)} - c), \sqrt{n_2}(c^{(2)} - c), \ldots, \sqrt{n_k}(c^{(k)} - c))$$

and (11), respectively. (Using this equivalent but $m \times k$ form of H_B reduces complexity.)

2) Compute the complete orthogonal decomposition

$$P^T K Q = \begin{pmatrix} R & 0 \\ 0 & 0 \end{pmatrix}, \text{ where } K = \begin{pmatrix} H_B^T \\ H_W^T \end{pmatrix} \in \mathbb{R}^{(k+n)\times m}$$

3) Let $t = \text{rank}(K)$.

4) Compute W from the SVD of $P(1:k, 1:t)$, which is

$$U^T P(1:k, 1:t) W = \Sigma_A.$$

5) Compute the first $k-1$ columns of $X = Q \begin{pmatrix} R^{-1}W & 0 \\ 0 & I \end{pmatrix}$, and assign them to G.

6) $y = G^T a$

Fig. 1. LDA/GSVD algorithm.

3.3 Equivalence of $\mathbf{J}_1 = \text{trace}(\mathbf{S}_2^{-1}\mathbf{S}_1)$ Criteria for Various $\mathbf{S}_1$ and $\mathbf{S}_2$

For the case when

$$(S_1, S_2) = (S_M, S_W),$$

if we follow the analysis in Section 3.1 literally, it appears that we would have to include $\text{rank}(S_M)$ (which is not less than or equal to $k-1$) columns of X in G. However, using the relation (2), the generalized eigenvalue problem

$$S_M x_i = \lambda_i S_W x_i$$

can be rewritten as

$$S_B x_i = (\lambda_i - 1) S_W x_i, \quad \text{where} \quad \lambda_i \geq 1 \quad \text{for} \quad 1 \leq i \leq m.$$

In this case, the eigenvector matrix is the same as for the case of $(S_1, S_2) = (S_B, S_W)$, but the eigenvalue matrix is $\Lambda - I$. Since the same permutation can be used to put $\Lambda - I$ in nonincreasing order as was used for Λ, x_i corresponds to the ith largest eigenvalue of $S_W^{-1} S_B$. Therefore, when S_W is nonsingular, the solution is the same as for $(S_1, S_2) = (S_B, S_W)$.

When $m > n$, the scatter matrix S_W is singular. For a generalized singular vector $x_i \in \text{null}(S_W)$, $S_M x_i = S_B x_i$. Hence, we include the same columns in G as we did in the case of $(S_1, S_2) = (S_B, S_W)$. Alternatively, we can show that the solutions are the same by deriving a GSVD of the matrix pair (H_M^T, H_W^T) that has the same generalized singular vectors as (H_B^T, H_W^T). To do this, we establish the following two properties of H_B and H_W.

Property 1. $H_W H_B^T = 0$.

Proof. From the definitions (11)-(12), we have

$$\begin{aligned} H_W H_B^T &= \sum_{i=1}^{k} \sum_{j \in N_i} (a_j - c^{(i)})(c^{(i)} - c)^T \\ &= \sum_{i=1}^{k} (n_i c^{(i)} c^{(i)^T} - n_i c^{(i)} c^T - n_i c^{(i)} c^{(i)^T} + n_i c^{(i)} c^T) \\ &= 0. \end{aligned}$$

□

Remark 1. This provides the following simple proof of the scatter matrix relationship (2).

Proof. From (11)-(14), we have

$$\begin{aligned} S_M &= H_M H_M^T = (H_W + H_B)(H_W + H_B)^T \\ &= H_W H_W^T + H_B H_B^T + H_B H_W^T + H_W H_B^T \\ &= S_W + S_B. \end{aligned}$$

□

Property 2. *For* $K = (H_B, H_W)^T \in \mathbb{R}^{2n\times m}$, $t = \text{rank}(K) \leq n$.

Proof. For $K^T = (H_B, H_W) \in \mathbb{R}^{m\times 2n}$, we have

$\mathrm{rank}(K^T) + \dim(\mathrm{null}(K^T)) = 2n,$ or $\dim(\mathrm{null}(K^T)) = 2n - t.$

Hence, $t \leq n$ if and only if $\dim(\mathrm{null}(K^T)) \geq n$. Suppose $z_1 \in \mathrm{null}(H_B)$ and $z_2 \in \mathrm{null}(H_W)$. Then,

$$(H_B, H_W)\begin{pmatrix} z_1 \\ 0 \end{pmatrix} = (H_B, H_W)\begin{pmatrix} 0 \\ z_2 \end{pmatrix} = 0.$$

This shows that

$$\dim(\mathrm{null}(H_B, H_W)) \geq \dim(\mathrm{null}(H_B)) + \dim(\mathrm{null}(H_W)).$$

Property 1 implies

$$\dim(\mathrm{null}(H_W)) \geq \mathrm{rank}(H_B^T).$$

Combining this with

$$\dim(\mathrm{null}(H_B)) = n - \mathrm{rank}(H_B),$$

we have

$$\dim(\mathrm{null}(H_B, H_W)) \geq n - \mathrm{rank}(H_B) + \mathrm{rank}(H_B^T) = n.$$

□

Now, we proceed with the GSVD derivation. For the case of $(S_1, S_2) = (S_B, S_W)$, consider the GSVD of the pair (H_B^T, H_W^T), which is given by

$$U^T H_B^T X = (\Sigma_B, 0) \quad \text{and} \quad V^T H_W^T X = (\Sigma_W, 0),$$

where

$$\Sigma_B \text{ and } \Sigma_W \in \mathbb{R}^{n \times t}, \quad \Sigma_B^T \Sigma_B + \Sigma_W^T \Sigma_W = I_t,$$

and

$$t = \mathrm{rank}\begin{pmatrix} H_B^T \\ H_W^T \end{pmatrix}.$$

Then, we have

$$\begin{aligned} H_M^T &= U(\Sigma_B, 0)X^{-1} + V(\Sigma_W, 0)X^{-1} \\ &= U(\Sigma_B + U^T V \Sigma_W, 0)X^{-1}. \end{aligned}$$

In addition,

$$H_W H_B^T = X^{-T}\begin{pmatrix} \Sigma_W^T \\ 0 \end{pmatrix} V^T U(\Sigma_B, 0)X^{-1} = 0_m$$

implies

$$\Sigma_W^T V^T U \Sigma_B = 0_t.$$

Hence,

$$\begin{aligned} &(\Sigma_B + U^T V \Sigma_W)^T(\Sigma_B + U^T V \Sigma_W) \\ &= \Sigma_B^T \Sigma_B + \Sigma_W^T (V^T U U^T V)\Sigma_W + \Sigma_W^T V^T U \Sigma_B + \Sigma_B^T U^T V \Sigma_W \\ &= \Sigma_B^T \Sigma_B + \Sigma_W^T \Sigma_W = I_t, \end{aligned}$$

which means $\Sigma_B + U^T V \Sigma_W$ has orthonormal columns. This can only be true if $\Sigma_B + U^T V \Sigma_W$ has no more columns than rows, i.e., if $t \leq n$ as shown above in Property 2.

There exists $\hat{U}_2$ such that $(\Sigma_B + U^T V \Sigma_W, \hat{U}_2) \in \mathbb{R}^{n \times n}$ is orthogonal. Hence,

$$H_M^T = U(\Sigma_B + U^T V \Sigma_W, \hat{U}_2)\begin{pmatrix} I_t & 0 \\ 0 & 0 \end{pmatrix} X^{-1},$$

and we can write

$$\hat{U}^T H_M^T X = (\Sigma_M, 0),$$

where

$$\hat{U} = U(\Sigma_B + U^T V \Sigma_W, \hat{U}_2) \text{ is orthogonal and } \Sigma_M = \begin{pmatrix} I_t \\ 0 \end{pmatrix}.$$

Together with

$$V^T H_W^T X = (\Sigma_W, 0),$$

this forms a GSVD of the matrix pair (H_M^T, H_W^T), which has the same generalized singular vectors as (H_B^T, H_W^T). As expected, each of the t nontrivial generalized singular values is infinite, finite and greater than one, or equal to one. Note that this form of GSVD for (H_M^T, H_W^T) does not satisfy the condition $\Sigma_M^T \Sigma_M + \Sigma_W^T \Sigma_W = I$ of the Paige and Saunders [14] formulation because each $\lambda_i \geq 1$. However, the invariance property and nonuniqueness of the generalized singular vector matrix X can be used to convert it to the Paige and Saunders form.

Note that, if S_W is nonsingular, in the m-dimensional space,

$$\begin{aligned} \mathrm{trace}(S_W^{-1} S_M) &= \mathrm{trace}(S_W^{-1}(S_W + S_B)) \\ &= m + \mathrm{trace}(S_W^{-1} S_B) \end{aligned} \tag{17}$$

and, in the l-dimensional space,

$$\begin{aligned} \mathrm{trace}((S_W^Y)^{-1} S_M^Y) &= \mathrm{trace}((S_W^Y)^{-1}(S_W^Y + S_B^Y)) \\ &= l + \mathrm{trace}((S_W^Y)^{-1} S_B^Y). \end{aligned} \tag{18}$$

This confirms that the solutions are the same for both $(S_1, S_2) = (S_B, S_W)$ and $(S_1, S_2) = (S_M, S_W)$. From Section 3.1, when G includes the eigenvectors of $S_W^{-1} S_B$ corresponding to the $l \geq k-1$ largest eigenvalues, then

$$\mathrm{trace}(S_W^{-1} S_B) = \mathrm{trace}((S_W^Y)^{-1} S_B^Y).$$

By subtracting (18) from (17) for any $l \geq k-1$, we get

$$\mathrm{trace}((S_W^Y)^{-1} S_M^Y) + (m - l) = \mathrm{trace}(S_W^{-1} S_M). \tag{19}$$

In other words, each additional eigenvector beyond the leftmost $k-1$ will add one to $\mathrm{trace}((S_W^Y)^{-1} S_M^Y)$. This shows that we do not preserve the cluster structure when measured by $\mathrm{trace}(S_W^{-1} S_M)$, although we do preserve $\mathrm{trace}(S_W^{-1} S_B)$. According to (19), $\mathrm{trace}(S_W^{-1} S_M)$ will be preserved only if we include all m eigenvectors of $S_W^{-1} S_M$. This, together with Section 3.1, shows indirectly that $\mathrm{rank}(S_M) = m$. That is, S_M is nonsingular whenever S_W is.

For the case

$$(S_1, S_2) = (S_W, S_M),$$

we want to minimize $\mathrm{trace}(S_M^{-1} S_W)$. A similar argument shows that the solution is the same as for $(S_1, S_2) = (S_B, S_W)$, even when S_W is singular. However, since we are minimizing in this case, the generalized singular values are in nondecreasing order, taking on reciprocal values of those for (H_M^T, H_W^T).

Algorithm 2 Orthogonal Centroid

Given a data matrix $A \in \mathbb{R}^{m \times n}$ with k clusters and an input vector $a \in \mathbb{R}^{m \times 1}$, compute a k-dimensional representation y of a.

1) Compute the centroid $c^{(i)}$ of the ith cluster, $1 \le i \le k$.
2) Set $C = (c^{(1)}, c^{(2)}, \dots, c^{(k)})$.
3) Compute the matrix Q_k in the reduced QR decomposition $C = Q_k R$.
4) $y = Q_k^T a$.

Fig. 2. Orthogonal centroid algorithm.

Having shown the equivalence of the J_1 criteria for various (S_1, S_2), we conclude that

$$(S_1, S_2) = (S_B, S_W)$$

should be used for the sake of computational efficiency. The LDA/GSVD algorithm reduces computational complexity further by using an $m \times k$ form of H_B rather than that presented in (12), and it avoids a potential loss of information [6, p. 239, Example 5.3.2] by not explicitly forming S_B and S_W as cross-products of H_B and H_W.

4 Alternative Approaches

4.1 Orthogonal Centroid

Simpler criteria for preserving cluster structure, such as $\min \operatorname{trace}(G^T S_W G)$ and $\max \operatorname{trace}(G^T S_B G)$, involve only one of the scatter matrices. A straightforward minimization of $\operatorname{trace}(G^T S_W G)$ seems meaningless since the optimum always reduces the dimension to one, even when the solution is restricted to the case when G has orthonormal columns. On the other hand, with the same restriction, maximization of $\operatorname{trace}(G^T S_B G)$ produces an equivalent solution to the Orthogonal Centroid method, which is introduced and shown to give promising reduced dimensional classification results in [15] and is summarized in Algorithm 2 (Fig. 2).

Let

$$J_2(G) = \operatorname{trace}(G^T S_B G).$$

If we let $G \in \mathbb{R}^{m \times l}$ be any matrix with full column rank, then, essentially, there is no upper bound and maximization is also meaningless. Now, let us restrict the solution to the case when G has orthonormal columns. Then, there exists $\hat{G} \in \mathbb{R}^{m \times (m-l)}$ such that $\left(G, \ \hat{G}\right)$ is an orthogonal matrix. In addition, since S_B is positive semidefinite, we have

$$\begin{aligned} \operatorname{trace}(G^T S_B G) &\le \operatorname{trace}(G^T S_B G) + \operatorname{trace}(\hat{G}^T S_B \hat{G}) \\ &= \operatorname{trace}(S_B). \end{aligned}$$

If the SVD of H_B is given by $H_B = U \Sigma V^T$, then $S_B U = U \Sigma \Sigma^T$. Hence, the columns of U form an orthonormal set of eigenvectors of S_B corresponding to the nonincreasing eigenvalues λ_i on the diagonal of $\Lambda = \Sigma \Sigma^T$. For $q = \operatorname{rank}(S_B)$, if we let U_q denote the first q columns of U and $\Lambda_q = \operatorname{diag}(\lambda_1, \dots, \lambda_q)$, we have

$$\begin{aligned} J_2(U_q) &= \operatorname{trace}(U_q^T S_B U_q) \\ &= \operatorname{trace}(U_q^T U_q \Lambda_q) \\ &= \lambda_1 + \cdots + \lambda_q \\ &= \operatorname{trace}(S_B). \end{aligned}$$

This means that we preserve $\operatorname{trace}(S_B)$ if we take U_q as G.

Now, we show that this solution is equivalent to the solution of the Orthogonal Centroid method, which does not involve the computation of eigenvectors. Defining the centroid matrix

$$C = (c^{(1)}, c^{(2)}, \dots, c^{(k)})$$

as in Algorithm 2, C has the reduced QR decomposition $C = Q_k R$, where $Q_k \in \mathbb{R}^{m \times k}$ has orthonormal columns and $R \in \mathbb{R}^{k \times k}$ [6, Section 5.2.6]. Suppose x is an eigenvector of S_B corresponding to the nonzero eigenvalue λ. Then,

$$S_B x = \sum_{i=1}^{k} n_i (c^{(i)} - c)(c^{(i)} - c)^T x = \lambda x.$$

This means $x \in \operatorname{span}\{c^{(i)} - c \mid 1 \le i \le k\}$ and, hence, $x \in \operatorname{span}\{c^{(i)} \mid 1 \le i \le k\}$. Accordingly,

$$\operatorname{range}(U_q) \subseteq \operatorname{range}(C) \subseteq \operatorname{range}(Q_k),$$

which implies that

$$U_q = Q_k W$$

for some matrix $W \in \mathbb{R}^{k \times q}$ with orthonormal columns. This yields

$$\begin{aligned} J_2(U_q) &= \operatorname{trace}(W^T Q_k^T S_B Q_k W) \\ &\le \operatorname{trace}(Q_k^T S_B Q_k) \\ &= J_2(Q_k), \end{aligned}$$

where the inequality follows from the positive semidefiniteness of $Q_k^T S_B Q_k$. Hence,

$$J_2(Q_k) = \operatorname{trace}(S_B),$$

and Q_k plays the same role as U_q. In other words, instead of computing the eigenvectors, we simply need to compute Q_k, which is much cheaper. Therefore, by computing a reduced QR decomposition of the centroid matrix, we obtain a solution that maximizes $\operatorname{trace}(G^T S_B G)$ over all G with orthonormal columns.

4.2 Two-Stage Approach

As mentioned in the Introduction, another approach for dealing with the singularity of S_W when $m > n$ uses LSI/SVD as a first stage, followed by the discriminant analysis stage. The goal of the first stage is to reduce the dimension of the data matrix enough so that the new S_W is nonsingular, and classical LDA can be performed. LSI/SVD uses the truncated SVD to find a $\mathrm{rank}\text{-}l$ approximation of A. That is, if $l \leq \mathrm{rank}(A)$, then

$$A \approx U_l \Sigma_l V_l^T,$$

where the columns of U_l are the first l left singular vectors, Σ_l is a diagonal matrix with the l largest singular values in nonincreasing order along its diagonal, and the columns of V_l are the first l right singular vectors. LSI/SVD typically uses $\Sigma_l V_l^T$ as the reduced dimensional representation of A or, equivalently, it computes the l-dimensional representation of $a \in \mathbb{R}^{m\times 1}$ as $y = U_l^T a$.

It is well known that the truncated SVD provides the closest approximation to A in the Frobenius or L_2 norm. However, unless performed locally on each cluster as in [9], [16], LSI ignores the cluster structure while reducing the dimension to l. Since there is no theoretical optimum value of l, potentially expensive testing may be required to determine the intermediate reduced dimensional representation of A that should be the input for the LDA stage.

To avoid losing discriminatory information in the first stage, LSI can reduce the dimension from m to n so that the data matrix becomes square. This is analogous to the PCA plus LDA approach of [21]. However, S_W remains singular after stage one, so classical LDA cannot be used in stage two. This provides another role for LDA/GSVD in addition to its single-stage role.

5 Experimental Results

In this section, we demonstrate the effectiveness of the LDA/GSVD and Orthogonal Centroid algorithms, which use the J_1 criterion with $(S_1, S_2) = (S_B, S_W)$ and the J_2 criterion with $G^TG = I$, respectively. For LDA/GSVD, we confirm its mathematical equivalence to J_1 using an alternative choice of (S_1, S_2), and we illustrate the discriminatory power of J_1 via two-dimensional projections. Just as important, we validate our extension of J_1 to the singular case. For Orthogonal Centroid, its preservation of $\mathrm{trace}(S_B)$ is shown to be a very effective compromise for the simultaneous optimization of two traces approximated by J_1. Our final tests show the sensitivity of the two-stage approach to the reduced dimension in its first stage.

5.1 Equivalence of J_1 for (S_B, S_W) and (S_M, S_W)

In Table 1, we use clustered data that are artificially generated by an algorithm adapted from [10, Appendix H]. The data consist of 2,000 vectors in a space of dimension 150, with $k = 7$ clusters. LDA/GSVD reduces the dimension from 150 to $k - 1 = 6$. We compare the LDA/GSVD criterion, $J_1 = \mathrm{trace}(S_W^{-1}S_B)$, with the alternative J_1 criterion, $\mathrm{trace}(S_W^{-1}S_M)$. The trace values confirm our theoretical findings, namely, that the generalized eigenvectors that optimize the alternative J_1 also optimize LDA/GSVD's J_1, and including an additional eigenvector increases $\mathrm{trace}(S_W^{-1}S_M)$ by one.

TABLE 1
Traces and Misclassification Rates (in Percent) with L_2 Norm Similarity

Method	Full	trace($S_W^{-1}S_B$)	trace($S_W^{-1}S_M$)	
Dim	150×2000	6×2000	6×2000	7×2000
trace(S_W)	299700	1.97	1.48	1.98
trace(S_B)	22925	4.03	3.04	3.04
trace(S_M)	322630	6.00	4.52	5.02
trace($S_W^{-1}S_B$)	12.6	12.6	12.6	12.6
trace($S_W^{-1}S_M$)	162.6	18.6	18.6	19.6
centroid	2.6 %	2.2 %	2.0 %	2.0 %
5nn	18.7 %	2.2 %	2.2 %	2.4 %
15nn	10.1 %	1.8 %	1.9 %	2.1 %

We also report misclassification rates for a centroid-based classification method [7] and the k nearest neighbor (knn) classification method [18, Section 2.6], which are summarized in Algorithms 3 (Fig. 3) and 4 (Fig. 4). (Note that the classification parameter of knn differs from the number of clusters k.) These are obtained using the L_2 norm, or Euclidean distance, similarity measure. While these rates differ slightly with the choice of S_B or S_M and the reduction to six or seven rows using the latter, they establish no advantage of using S_M over S_B, even when we include an additional eigenvector to bring us closer to the preservation of $\mathrm{trace}(S_W^{-1}S_M)$. These results bolster our choice of $J_1 = \mathrm{trace}(S_W^{-1}S_B)$ in the LDA/GSVD algorithm since it limits the GSVD computation to a composite matrix with $k + n$ rows, rather than one with $2n$ rows.

5.2 Discriminatory Power of J_1

To further illustrate the power of the J_1 criterion, we apply it to the same 2,000 data vectors as in Section 5.1, this time reducing the dimension from 150 to two. Even though the optimal reduced dimension is six, $J_1 = \mathrm{trace}(S_W^{-1}S_B)$ does surprisingly well at discriminating among seven classes, as seen in Fig. 5. As expected, the alternative $J_1 = \mathrm{trace}(S_W^{-1}S_M)$ does equally well in Fig. 6. Although the classes overlap in these two figures, the misclassification rate using either J_1 formulation is approximately one third. In contrast, Fig. 7 shows that the truncated SVD, as used in LSI, is not the best discriminator.

5.3 Comparison to the Orthogonal Centroid Method in the Singular Case

Another set of experiments validates our extension of J_1 to the singular case. For this purpose, we use five categories of abstracts from the MEDLINE[2] database (see Table 2). Each category has 40 documents. There are 7,519 terms after preprocessing with stemming and removal of stop words [11]. Since 7,519 exceeds the number of documents (200), S_W is singular and classical discriminant analysis breaks down.

2. http://www.ncbi.nlm.nih.gov/PubMed.

Algorithm 3 Centroid-based Classification

Given a data matrix A with k clusters and k corresponding centroids, $c^{(i)}$ for $1 \leq i \leq k$, find the index j of the cluster to which a vector q belongs.

- find the index j such that $sim(q, c^{(i)})$, $1 \leq i \leq k$, is minimum (or maximum), where $sim(q, c^{(i)})$ is the similarity measure between q and $c^{(i)}$.

  (For example, $sim(q, c^{(i)}) = \|q - c^{(i)}\|_2$ using the L_2 norm, and we take the index with the minimum value. Using the cosine measure, $sim(q, c^{(i)}) = cos(q, c^{(i)}) = \frac{q^T c^{(i)}}{\|q\|_2 \|c^{(i)}\|_2}$, and we take the index with the maximum value.)

Fig. 3. Centroid-based classification algorithm.

Algorithm 4 k Nearest Neighbor (knn) Classification

Given a data matrix $A = (a_1, a_2, \ldots, a_n)$ with k clusters, find the cluster to which a vector q belongs.

1) From the similarity measure $sim(q, a_j)$ for $1 \leq j \leq n$, find the k nearest neighbors of q. (We use k to distinguish the algorithm parameter from the number of clusters k.)
2) Among these k vectors, count the number belonging to each cluster.
3) Assign q to the cluster with the greatest count in the previous step.

Fig. 4. k nearest neighbor (knn) classification algorithm.

However, our LDA/GSVD method circumvents this singularity problem.

The LDA/GSVD algorithm dramatically reduces the dimension 7,519 to four, or one less than the number of clusters. The Orthogonal Centroid method reduces the dimension to five. Table 3 shows classification results using the L_2 norm similarity measure. LDA/GSVD produces the lowest misclassification rate using both centroid-based and nearest neighbor classification methods. Because the J_1 criterion is not defined in this case, we compute the ratio $\mathrm{trace}(S_B)/\mathrm{trace}(S_W)$ as a rough optimality measure. We observe that the ratio is strikingly higher for LDA/GSVD reduction than for the other methods. These experimental results confirm that the LDA/GSVD algorithm effectively extends the applicability of the J_1 criterion to cases that classical discriminant analysis cannot handle. In addition, the Orthogonal Centroid algorithm preserves $\mathrm{trace}(S_B)$ from the full dimension without the expense of computing eigenvectors. Taken together, the results for these two methods demonstrate the potential for dramatic and efficient dimension reduction without compromising cluster structure.

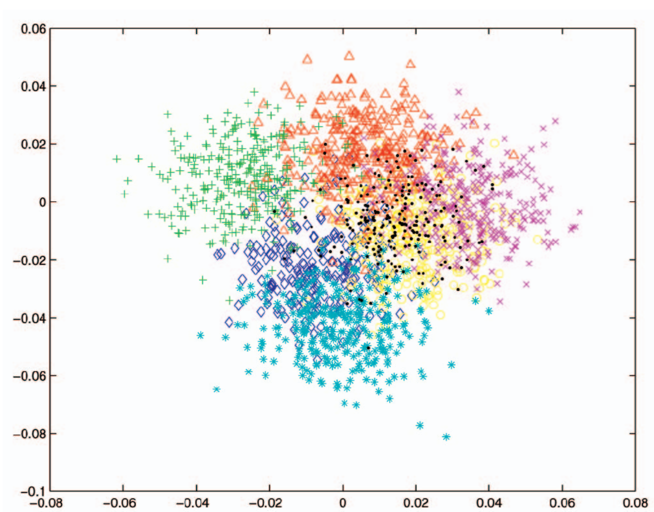


Fig. 5. Max $\mathrm{trace}(S_W^{-1} S_B)$ projection onto two dimensions.

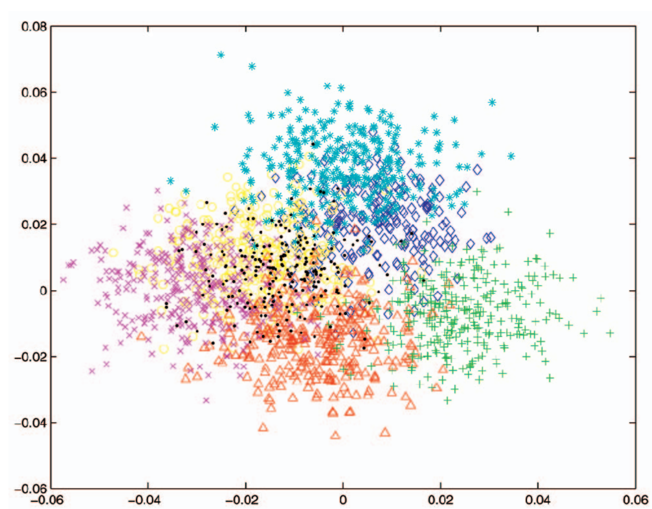


Fig. 6. Max $\mathrm{trace}(S_W^{-1} S_M)$ projection onto two dimensions.

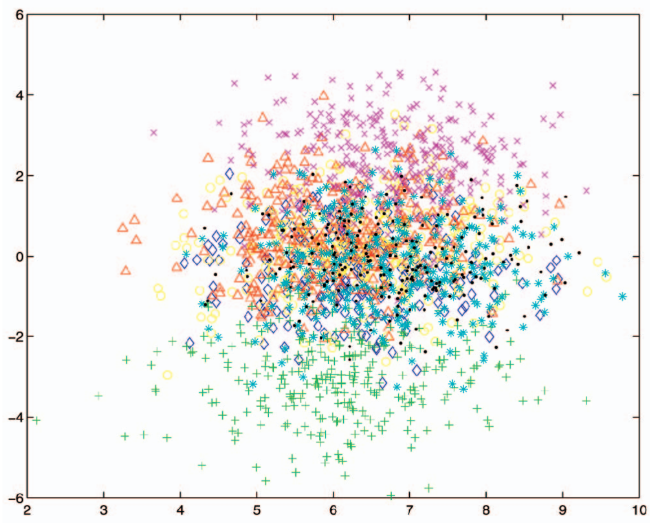


Fig. 7. Two-dimensional representation using $\Sigma_2 V_2^T$ from truncated SVD.

5.4 Comparison to the Two-Stage Approach in the Singular Case

A final set of experiments also uses the MEDLINE database of Table 2. The results are summarized in Tables 4 and 5. Table 4 compares our LDA/GSVD method with two-stage approaches whose LSI stage reduces the data dimension to $n = 200$. Although the data matrix is square after the LSI stage, S_W remains singular. We first note that, if the second stage uses the GSVD, then the final trace values are identical to those of the single stage of LDA/GSVD, and the misclassification rates are almost identical. However, if MATLAB's eig function is used for the second stage, the trace values are scaled quite differently, and the classification results are slightly better for centroid and 1nn classification and considerably worse for 3nn classification. Clearly, the intermediate reduction to a square matrix produces widely varying results depending on which LDA algorithm is used in the second stage.

In Table 5, the dimension reduction methods vary only in the intermediate dimension after the LSI stage. Since $\mathrm{rank}(S_W) = 195$, we include it in our range of LSI dimensions and conclude with LSI to the LDA/GSVD optimum dimension of 4. Our rough optimality measure, $\mathrm{trace}(S_B)/\mathrm{trace}(S_W)$, declines as the LSI dimension decreases and misclassification rates increase over the same range. After LSI reduces to $\mathrm{rank}(S_W) = 195$, the eig function produces results comparable to LDA/GSVD alone. However, two $m \times m$ scatter matrices are needed as input to the eig function. These tests clearly show the sensitivity of the two-stage approach to the dimension chosen in the LSI stage.

These tests of the two-stage approach also bring up several issues in its usage. First of all, what LSI dimension will result in nonsingular S_W? Second, when choosing the generalized eigenvectors to include as columns of the G matrix in the LDA stage, what is the meaning of negative generalized eigenvalues? This is in contrast to the GSVD approach, for which we have infinite, finite and positive, zero, and arbitrary generalized singular values and a rationale for the inclusion or exclusion of the corresponding generalized singular vectors in the solution matrix G. Third, which algorithm should be used for the LDA stage, particularly when the LSI dimension is close to n so that the LSI representation is square but S_W is singular? These issues are explored in [8].

TABLE 2
MEDLINE Data Set

class	category	no. of documents
1	heart attack	40
2	colon cancer	40
3	diabetes	40
4	oral cancer	40
5	tooth decay	40
	dimension	7519×200

6 Conclusion

Our experimental results verify that the J_1 criterion, when applicable, effectively optimizes classification in the

TABLE 3
Traces and Misclassification Rate with L_2 Norm Similarity

Method		Full	Orthogonal Centroid	LDA/GSVD
Dim		7519×200	5×200	4×200
trace values	$\mathrm{trace}(S_W)$	73048	4210	0.05
	$\mathrm{trace}(S_B)$	<u>6229</u>	<u>6229</u>	3.95
	$\frac{\mathrm{trace}(S_B)}{\mathrm{trace}(S_W)}$	0.09	1.5	<u>79</u>
misclassification rate in %	centroid	5	5	1
	1nn	40	3	1

TABLE 4
Traces and Misclassification Rate (in Percent) with L_2 Norm Similarity

Method	Full	LDA/GSVD	LSI→ 200 LDA/GSVD	LSI→ 200 LDA/EIG
Dim	7519×200	4×200	4×200	4×200
trace(S_W)	73048	0.05	0.05	3.17×10^{-54}
trace(S_B)	6229	3.95	3.95	6.11×10^{-25}
$\frac{\text{trace}(S_B)}{\text{trace}(S_W)}$	0.09	79	79	1.92×10^{29}
centroid	5%	1%	1%	0%
1nn	40%	1%	0%	0%
3nn	51%	1.5%	1.5%	19%

reduced dimensional space, while our LDA/GSVD extends the applicability to cases that classical discriminant analysis cannot handle. In addition, our LDA/GSVD algorithm never explicitly forms the scatter matrices, which results in two advantages. First, we avoid the numerical problems inherent in forming cross-product matrices. Second, we reduce the storage requirements considerably. Specifically, any algorithm that forms scatter matrices requires $m \times m$ for each, but Q in Step 2 is the only $m \times m$ matrix we store. Instead of scatter matrices, we store $H_B : k \times n$ and $H_W : m \times n$, which will be smaller when $m > n$.

A disadvantage of methods that involve the GSVD is that its computation is costly. Computationally, the most expensive part of Algorithm LDA/GSVD is Step 2, where a complete orthogonal decomposition is needed. Assuming $k \leq n$, $t \leq m$, and $t = \mathcal{O}(n)$, the complete orthogonal decomposition of K costs $\mathcal{O}(nmt)$ when $m \leq n$, and $\mathcal{O}(m^2 t)$ when $m > n$ [6]. Therefore, a fast algorithm needs to be developed for Step 2.

For Orthogonal Centroid, the most expensive step is the reduced QR decomposition of C, which costs $\mathcal{O}(mk^2)$ [6]. By solving a simpler eigenvalue problem and avoiding the computation of eigenvectors, Orthogonal Centroid is significantly cheaper than LDA/GSVD. Our experiments show it to be a very reasonable compromise.

Compared to the two-stage approach of LSI followed by LDA, our one-stage LDA/GSVD avoids the potentially costly experimentation involved in determining the dimension for LSI. Short of experimenting with various LSI dimensions, one could reduce the data to dimension n so that the matrix is square, but classification results after the LDA stage may vary widely depending on the LDA method chosen. Our preliminary results show that use of the GSVD may have numerical advantages in this context as well.

It bears repeating that dimension reduction is only a preprocessing stage. Since classification and retrieval will be the dominating parts computationally, the expense of dimension reduction should be weighed against its effectiveness in reducing the cost involved in those processes.

Finally, the GSVD provides a mathematical framework to aid in understanding the singular case. In addition to forming the basis of the new LDA/GSVD algorithm, it clearly shows that at most $k-1$ generalized eigenvectors are needed, even in the singular case.

TABLE 5
Traces and Misclassification Rate (in Percent) with L_2 Norm Similarity

Method	LSI→ 195 LDA/EIG	LSI→ 150 LDA/EIG	LSI→ 50 LDA/EIG	LSI→ 20 LDA/EIG	LSI→ 4 LDA/EIG
Dim	4×200	4×200	4×200	4×200	4×200
trace(S_W)	14.07	313	1446	2963	6962
trace(S_B)	850.60	2903	4555	5124	3473
$\frac{\text{trace}(S_B)}{\text{trace}(S_W)}$	60.42	9.27	3.15	1.73	0.50
centroid	1%	5%	6%	8%	34.5%
1nn	2%	3.5%	4%	8%	24%
3nn	1%	2.5%	3.5%	7.5%	33.5%

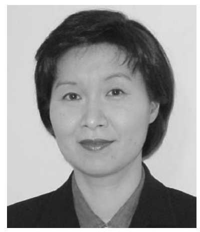
ACKNOWLEDGMENTS

This work was supported in part by 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.N. Belhumeur, J.P. 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, July 1997.
[2] M.W. Berry, S.T. Dumais, and G.W. O'Brien, "Using Linear Algebra for Intelligent Information Retrieval," *SIAM Rev.,* vol. 37, no. 4, pp. 573-595, 1995.
[3] S. Deerwester, S. Dumais, G. Furnas, T. Landauer, and R. Harshman, "Indexing by Latent Semantic Analysis," *J. Am. Soc. for Information Science,* vol. 41, pp. 391-407, 1990.
[4] R. Duda, P. Hart, and D. Stork, *Pattern Classification,* second ed. John Wiley & Sons, 2001.
[5] K. Fukunaga, *Introduction to Statistical Pattern Recognition,* second ed. Academic Press, 1990.
[6] G.H. Golub and C.F. Van Loan, *Matrix Computations,* third ed. Johns Hopkins Univ. Press, 1996.
[7] 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 Applications,* vol. 25, no. 1, pp. 165-179, 2003.
[8] P. Howland and H. Park, "Equivalence of Several Two-Stage Methods for Linear Discriminant Analysis," Technical Report 041, Dept. of Computer Science and Eng., Univ. of Minnesota, 2003.
[9] D. Hull, "Improving Text Retrieval for the Routing Problem Using Latent Semantic Indexing," *Proc. 17th Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval,* pp. 282-291, 1994.
[10] A.K. Jain and R.C. Dubes, *Algorithms for Clustering Data.* Prentice Hall, 1988.
[11] G. Kowalski, *Information Retrieval Systems: Theory and Implementation.* Kluwer Academic, 1997.
[12] C.L. Lawson and R.J. Hanson, *Solving Least Squares Problems.* SIAM, 1995.
[13] J. Ortega, *Matrix Theory: A Second Course.* Plenum Press, 1987.
[14] C.C. Paige and M.A. Saunders, "Towards a Generalized Singular Value Decomposition," *SIAM J. Numerical Analysis,* vol. 18, no. 3, pp. 398-405, 1981.
[15] H. Park, M. Jeon, and J.B. Rosen, "Lower Dimensional Representation of Text Data Based on Centroids and Least Squares," *BIT Numerical Math.,* vol. 42, no. 2, pp. 1-22, 2003.
[16] H. Schütze, D. Hull, and J. Pedersen, "A Comparison of Classifiers and Document Representations for the Routing Problem," *Proc. 18th Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval,* pp. 229-237, 1995.
[17] 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.
[18] S. Theodoridis and K. Koutroumbas, *Pattern Recognition.* Academic Press, 1999.
[19] K. Torkkola, "Linear Discriminant Analysis in Document Classification," *Proc. IEEE ICDM Workshop Text Mining,* 2001.
[20] C.F. Van Loan, "Generalizing the Singular Value Decomposition," *SIAM J. Numerical Analysis,* vol. 13, no. 1, pp. 76-83, 1976.
[21] J. Yang and J.Y. Yang, "Why Can LDA Be Performed in PCA Transformed Space?" *Pattern Recognition,* vol. 36, no. 2, pp. 563-566, 2003.

Peg Howland is a PhD candidate in computer science at the University of Minnesota, where she is currently also an instructor. She was awarded the Guidant Fellowship in 2001-2002. She received the MS degree in applied mathematics and BS degree with highest honors in mathematics from Purdue University, in 1978 and 1976, respectively. Her research interests include using numerical linear algebra to find lower dimensional representations of high-dimensional data, as encountered in information retrieval, facial recognition, and bioinformatics. This builds on methods applied in other contexts, e.g., discriminant analysis projections from pattern recognition and matrix factorization techniques from applied statistics/psychometrics.



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. She received the MS and PhD degrees in computer science from Cornell University, Ithaca, New York, in 1985 and 1987, respectively. She has been on the faculty of the Department of Computer Science and Engineering, University of Minnesota, Twin Cities since 1987 where she is currently a professor. Her current research interests include numerical linear algebra, pattern recognition, data analysis, information retrieval, and bioinformatics. Dr. Park served on the editorial board of the *SIAM Journal on Scientific Computing* from 1993 to 1999. Currently, she is on the editorial board of *Mathematics of Computation*, *BIT Numerical Mathematics*, and *Computational Statistics and Data Analysis* special issue on numerical linear algebra and statistics. She has recently served on the committees of several meetings including the program committee for the SIAM Conference on Data Mining 2004. From November 2003, Dr. Park will serve at the US National Science Foundation as the ACR (Advanced Computational Research) program director, Directorate for Computer and Information Science and Engineering, the National Science Foundation, Arlington, Virginia.