

Equivalence of Several Two-stage Methods for Linear Discriminant Analysis*

Peg Howland and Haesun Park[†]

Abstract

Linear discriminant analysis (LDA) has been used for decades to extract features that preserve class separability. It is classically 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 restricts its application to data sets in which the dimension of the data does not exceed the sample size. Recently, the applicability of LDA has been extended by using the generalized singular value decomposition (GSVD) to circumvent the nonsingularity requirement. Alternatively, many studies have taken a two-stage approach in which the first stage reduces the dimension of the data enough so that it can be followed by classical LDA. In this paper, we justify the two-stage approach by establishing its equivalence to the single-stage LDA/GSVD method, provided either principal component analysis or latent semantic indexing is used in the first stage over a certain range of intermediate dimensions. We also present a computationally simpler choice for the first stage, and conclude with a discussion of the relative merits of each approach.

1 Introduction

The goal of linear discriminant analysis (LDA) is to combine features of the original data in a way that most effectively discriminates between classes. With an appropriate extension, it can be applied to the goal of reducing the dimension of a data matrix in a way that most effectively preserves its cluster structure. That is, we want to find a linear transformation G^T that 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 $A \in \mathbb{R}^{m \times n}$,

*This work was supported in part by the 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 National Science Foundation (NSF).

[†]Department of Computer Science and Engineering, University of Minnesota, Minneapolis, MN 55455, {howland,hpark}@cs.umn.edu

and are grouped into k clusters as

(1.1)

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

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, 15] as

$$\begin{aligned} S_w &= \sum_{i=1}^k \sum_{j \in N_i} (a_j - c^{(i)})(a_j - c^{(i)})^T, \\ S_b &= \sum_{i=1}^k \sum_{j \in N_i} (c^{(i)} - c)(c^{(i)} - c)^T \\ &= \sum_{i=1}^k n_i (c^{(i)} - c)(c^{(i)} - c)^T, \text{ and} \\ S_m &= \sum_{j=1}^n (a_j - c)(a_j - c)^T, \end{aligned}$$

respectively. The scatter matrices have the relationship [8]

$$(1.2) \quad S_m = S_w + S_b.$$

Applying G^T to the matrix A transforms the scatter matrices S_w , S_b , and S_m to the $l \times l$ matrices

$$G^T S_w G, \quad G^T S_b G, \quad \text{and} \quad G^T S_m G,$$

respectively.

There are several measures of cluster quality that involve the three scatter matrices [5, 15]. When cluster quality is

high, each cluster is tightly grouped, but well separated from the other clusters. Since

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

measures the closeness of the columns within the clusters, and

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

measures the separation between clusters, an optimal transformation that preserves the given cluster structure would maximize $\text{trace}(G^T S_b G)$ and minimize $\text{trace}(G^T S_w G)$.

This simultaneous optimization can be approximated by finding a transformation G that maximizes

$$(1.3) \quad J_1(G) = \text{trace}((G^T S_w G)^{-1} G^T S_b G).$$

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, for 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. Such is the case for the image databases of facial recognition, as well as for gene expression data. This is referred to as the small sample size problem, or the problem of undersampled data.

Classical LDA expresses the solution in terms of an eigenvalue problem when S_w is nonsingular. By reformulating the problem in terms of the generalized singular value decomposition (GSVD) [17, 13, 6], the LDA/GSVD algorithm [7] has extended the applicability to the case when S_w is singular. Another way to make classical LDA 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 LDA stage is preceded by a stage in which the cluster structure is ignored. A common approach for the first part of this process is rank reduction by the truncated singular value decomposition (SVD). This is the main tool in principal component analysis (PCA) [4], as well as in latent semantic indexing (LSI) [3, 1] of documents. Both Swets and Weng [14] and Belhumeur et al. [2] have utilized PCA plus LDA

for facial feature extraction. More recently, Torkkola [16] implemented LSI plus LDA for document classification. A drawback of these two-stage approaches is that experimentation has been needed to determine which intermediate reduced dimension produces optimal results after the second stage.

Moreover, since either PCA or LSI ignores the cluster structure in the first stage, theoretical justification for such two-stage approaches has been lacking. Yang and Yang [18] supplied theoretical justification for PCA plus LDA, for a single discriminant vector. In this paper, we justify the two-stage approach that uses either PCA or LSI, followed by LDA. We do this by establishing the equivalence of the single-stage LDA/GSVD to the two-stage method, provided that the intermediate dimension after the first stage falls within a specific range. In this range S_w remains singular, and hence LDA/GSVD is required for the second stage. We explain how, in addition to its role in the LDA/GSVD algorithm, the GSVD provides a mathematical framework for understanding the singular case. We also present a computationally simpler choice for the first stage, which uses QR decomposition (QRD) rather than the SVD. After confirming the equivalence of these approaches experimentally, we discuss the relative merits of each. We conclude that QRD plus LDA, which uses QRD as a pre-processing step for LDA/GSVD, provides a fast algorithm for LDA/GSVD.

2 LDA based on the GSVD

It is well-known that the J_1 criterion (1.3) is maximized when the columns of G are the l eigenvectors of $S_w^{-1} S_b$ corresponding to the l largest eigenvalues [5]. In other words, classical discriminant analysis solves

$$(2.4) \quad S_w^{-1} S_b x_i = \lambda_i x_i$$

for the x_i 's corresponding to the largest λ_i 's. For these l eigenvectors, the maximum achieved is $J_1(G) = \lambda_1 + \dots + \lambda_l$. Since $\text{rank}(S_b)$ of the eigenvalues of $S_w^{-1} S_b$ are greater than zero, if $l \geq \text{rank}(S_b)$, this optimal G preserves $\text{trace}(S_w^{-1} S_b)$ exactly upon dimension reduction.

Assuming the cluster structure given in (1.1), [7] defines the $m \times n$ matrices

$$(2.5) \quad \begin{aligned}H_w &= [A_1 - c^{(1)} e^{(1)T}, A_2 - c^{(2)} e^{(2)T}, \dots, A_k - c^{(k)} e^{(k)T}] \\ H_b &= [(c^{(1)} - c) e^{(1)T}, (c^{(2)} - c) e^{(2)T}, \dots, (c^{(k)} - c) e^{(k)T}]\end{aligned}$$

$$(2.6) \quad H_m = [a_1 - c, \dots, a_n - c] = A - c e^T,$$

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

$$(2.7) \quad S_w = H_w H_w^T, \quad S_b = H_b H_b^T, \quad \text{and} \quad S_m = H_m H_m^T.$$

Another way to define H_b that satisfies (2.7) is

$$(2.8) \quad H_b = [\sqrt{n_1}(c^{(1)} - c), \sqrt{n_2}(c^{(2)} - c), \dots, \sqrt{n_k}(c^{(k)} - c)]$$

and using this $m \times k$ form reduces the storage requirements and computational complexity of the LDA/GSVD algorithm.

As the product of an $m \times n$ matrix and an $n \times m$ matrix, S_w is singular when $m > n$ [11]. This means that J_1 cannot be applied when the number of available data points is smaller than the dimension of the data. Expressing λ_i as α_i^2/β_i^2 , the eigenvalue problem (2.4) becomes

$$(2.9) \quad \beta_i^2 H_b H_b^T x_i = \alpha_i^2 H_w H_w^T x_i.$$

This has the form of a problem that can be solved using the GSVD of the matrix pair (H_b^T, H_w^T) . Paige and Saunders [13] defined the GSVD for any two matrices with the same number of columns, which we restate as follows.

THEOREM 2.1. *Suppose two matrices $H_b^T \in \mathbb{R}^{k \times m}$ and $H_w^T \in \mathbb{R}^{n \times m}$ are given. Then for*

$$K = \begin{pmatrix} H_b^T \\ H_w^T \end{pmatrix} \quad \text{and} \quad t = \text{rank}(K),$$

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

$$U^T H_b^T Q = \Sigma_b \left(\underbrace{W^T R}_t, \underbrace{0}_{m-t} \right)$$

and

$$V^T H_w^T Q = \Sigma_w \left(\underbrace{W^T R}_t, \underbrace{0}_{m-t} \right),$$

where

$$\Sigma_b = \begin{pmatrix} I_b & & \\ & D_b & \\ & & O_b \end{pmatrix}, \quad \Sigma_w = \begin{pmatrix} O_w & & \\ & D_w & \\ & & I_w \end{pmatrix},$$

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

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

are identity matrices, where

$$r = t - \text{rank}(H_w^T) \quad \text{and} \quad s = \text{rank}(H_b^T) + \text{rank}(H_w^T) - t,$$

$$O_b \in \mathbb{R}^{(k-r-s) \times (t-r-s)} \quad \text{and} \quad O_w \in \mathbb{R}^{(n-t+r) \times r}$$

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

$$D_b = \text{diag}(\alpha_{r+1}, \dots, \alpha_{r+s})$$

and

$$D_w = \text{diag}(\beta_{r+1}, \dots, \beta_{r+s})$$

satisfy

$$1 > \alpha_{r+1} \geq \dots \geq \alpha_{r+s} > 0, \quad 0 < \beta_{r+1} \leq \dots \leq \beta_{r+s} < 1,$$

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

This form of GSVD is related to that of Van Loan [17] as

$$(2.10) \quad U^T H_b^T X = (\Sigma_b, 0) \quad \text{and} \quad V^T H_w^T X = (\Sigma_w, 0),$$

where

$$\begin{matrix} X \\ m \times m \end{matrix} = Q \begin{pmatrix} R^{-1}W & 0 \\ 0 & I_{m-t} \end{pmatrix}.$$

This implies that

$$X^T H_b H_b^T X = \begin{pmatrix} \Sigma_b^T \Sigma_b & 0 \\ 0 & 0 \end{pmatrix}$$

and

$$X^T H_w H_w^T X = \begin{pmatrix} \Sigma_w^T \Sigma_w & 0 \\ 0 & 0 \end{pmatrix}.$$

Letting x_i represent the i th column of X , and defining

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

and

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

we see that (2.9) is satisfied for $1 \leq i \leq t$. Since

$$H_b H_b^T x_i = 0 \quad \text{and} \quad H_w H_w^T x_i = 0$$

for the remaining $m-t$ columns of X , (2.9) 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 (H_b^T, H_w^T) . They correspond to the generalized singular values, or the α_i/β_i quotients, as follows. The first r columns correspond to infinite values, and the next s columns correspond to finite and nonzero values. The following $t-r-s$ columns correspond to zero values, and the last $m-t$ columns correspond to the arbitrary values. This correspondence between generalized singular vectors and values is illustrated in Figure 1.

A question that remains is which columns of X to include in the solution G . If S_w is nonsingular, both $r=0$ and $m-t=0$, so $s = \text{rank}(H_b^T)$ generalized singular values are finite and nonzero, and the rest are zero. Hence we include in G the leftmost s columns of X . For the case

$\frac{\alpha_i}{\beta_i} = \infty$	$\infty > \frac{\alpha_i}{\beta_i} > 0$	$\frac{\alpha_i}{\beta_i} = 0$	α_i, β_i arbitrary
$\text{null}(S_w)$		$\text{null}(S_b)$	$\text{null}(S_w)$
–		–	\cap
$\text{null}(S_b)$		$\text{null}(S_w)$	$\text{null}(S_b)$
r	$r+s$	t	m
X			

Figure 1: Generalized singular vectors and their corresponding generalized singular values.

when S_w is singular, [7] argues in terms of the simultaneous optimization

$$(2.11) \quad \max_G \text{trace}(G^T S_b G) \text{ and } \min_G \text{trace}(G^T S_w G)$$

that criterion J_1 is approximating. Letting g_j represent a column of G , we write

$$\text{trace}(G^T S_b G) = \sum g_j^T S_b g_j$$

and

$$\text{trace}(G^T S_w G) = \sum g_j^T S_w g_j.$$

If x_i is one of the leftmost r vectors, then $x_i \in \text{null}(S_w) - \text{null}(S_b)$. Because $x_i^T S_b x_i > 0$ and $x_i^T S_w x_i = 0$, including this vector in G increases the trace we want to maximize while leaving the trace we want to minimize unchanged. On the other hand, for the rightmost $m - t$ vectors, $x_i \in \text{null}(S_w) \cap \text{null}(S_b)$. Adding the column x_i to G has no effect on these traces, since $x_i^T S_w x_i = 0$ and $x_i^T S_b x_i = 0$, and therefore does not contribute to either maximization or minimization in (2.11). We conclude that, whether S_w is singular or nonsingular, G should be comprised of the leftmost $r + s = \text{rank}(H_b^T)$ columns of X , which are shaded in Figure 1.

As a practical matter, LDA/GSVD includes the first $k - 1$ columns of X in G . This is due to the fact that $\text{rank}(H_b) \leq k - 1$, which is clear from the definition of H_b given in (2.8). If $\text{rank}(H_b) < k - 1$, including extra columns in G (some which correspond to the $t - r - s$ zero generalized singular values and, possibly, some which correspond to the arbitrary generalized singular values) will have approximately no effect on cluster preservation. As summarized in Algorithm 2.1, we first compute 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 2.1 [13]. The major steps are limited to the complete orthogonal decomposition [6, 10] of

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

which produces orthogonal matrices P and Q and a non-singular 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. Finally, we assign the leftmost $k - 1$ generalized singular vectors to G .

ALGORITHM 2.1. (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 (2.8) and (2.5), respectively.
2. Compute the complete orthogonal decomposition

$$P^T K Q = \begin{pmatrix} R & 0 \\ 0 & 0 \end{pmatrix},$$

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 $U^T P(1 : k, 1 : t) W = \Sigma_b$.
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. $y = G^T a$

3 Rank reduction based on the truncated SVD

As mentioned in the introduction, two-stage approaches to dimension reduction typically use the truncated SVD in the first stage. Either PCA or LSI may be used; they differ only in that PCA centers the data by subtracting the global centroid from each column of A . In this section, we express both methods in terms of the maximization of $J_2(G) = \text{trace}(G^T S_m G)$.

If we let $G \in \mathbb{R}^{m \times l}$ be any matrix with full column rank, then essentially $J_2(G)$ has no upper bound and maximization is meaningless. Now, let us restrict the solution to the case when G has orthonormal columns. Then there exists $G' \in \mathbb{R}^{m \times (m-l)}$ such that (G, G') is an orthogonal matrix. In addition, since S_m is positive semidefinite, we have

$$\begin{aligned} \text{trace}(G^T S_m G) &\leq \text{trace}(G^T S_m G) + \text{trace}((G')^T S_m G') \\ &= \text{trace}(S_m). \end{aligned}$$

Reserving the following notation for the SVD of A :

$$A = U \Sigma V^T,$$

let the SVD of H_m be given by

$$H_m = A - ce^T = \tilde{U} \tilde{\Sigma} \tilde{V}^T.$$

Then

$$S_m = H_m H_m^T = \tilde{U} \tilde{\Sigma} \tilde{\Sigma}^T \tilde{U}^T.$$

Hence the columns of \tilde{U} form an orthonormal set of eigenvectors of S_m corresponding to the nonincreasing eigenvalues on the diagonal of $\Lambda = \tilde{\Sigma} \tilde{\Sigma}^T = \text{diag}(\tilde{\sigma}_1^2, \dots, \tilde{\sigma}_n^2)$. For

$$p = \text{rank}(H_m),$$

if we denote the first p columns of \tilde{U} by \tilde{U}_p , and let $\Lambda_p = \text{diag}(\tilde{\sigma}_1^2, \dots, \tilde{\sigma}_p^2)$, we have

$$\begin{aligned} J_2(\tilde{U}_p) &= \text{trace}(\tilde{U}_p^T S_m \tilde{U}_p) \\ &= \text{trace}(\tilde{U}_p^T \tilde{U}_p \Lambda_p) \\ &= \tilde{\sigma}_1^2 + \dots + \tilde{\sigma}_p^2 \\ (3.12) \quad &= \text{trace}(S_m). \end{aligned}$$

This means that we preserve $\text{trace}(S_m)$ if we take \tilde{U}_p as G . Clearly, the same is true for \tilde{U}_l with $l \geq p$, so PCA to a dimension of at least $\text{rank}(H_m)$ preserves $\text{trace}(S_m)$.

Now we show that LSI also preserves $\text{trace}(S_m)$. Suppose x is an eigenvector of S_m corresponding to the eigenvalue $\lambda \neq 0$. Then

$$S_m x = \sum_{j=1}^n (a_j - c)(a_j - c)^T x = \lambda x.$$

This means $x \in \text{span}\{a_j - c | 1 \leq j \leq n\}$, and hence $x \in \text{span}\{a_j | 1 \leq j \leq n\}$. Accordingly,

$$\text{range}(\tilde{U}_p) \subseteq \text{range}(A).$$

For

$$q = \text{rank}(A),$$

we can write $A = U_q \Sigma_q V_q^T$, where U_q and V_q denote the first q columns of U and V , respectively, and $\Sigma_q = \Sigma(1 : q, 1 : q)$. Then $\text{range}(A) = \text{range}(U_q)$, which implies that

$$\text{range}(\tilde{U}_p) \subseteq \text{range}(U_q).$$

Hence

$$\tilde{U}_p = U_q W$$

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

$$\begin{aligned} J_2(\tilde{U}_p) &= J_2(U_q W) \\ &= \text{trace}(W^T U_q^T S_m U_q W) \\ &\leq \text{trace}(U_q^T S_m U_q) \\ &= J_2(U_q). \end{aligned}$$

Since $J_2(\tilde{U}_p) = \text{trace}(S_m)$ from (3.12), we preserve $\text{trace}(S_m)$ if we take U_q as G . The same argument holds for U_l with $l \geq q$, so LSI to any dimension greater than or equal to $\text{rank}(A)$ also preserves $\text{trace}(S_m)$.

Finally, in the range of reduced dimensions for which PCA and LSI preserve $\text{trace}(S_m)$, they preserve $\text{trace}(S_w)$ and $\text{trace}(S_b)$ as well. This follows from the scatter matrix relationship (1.2) and the inequalities

$$\begin{aligned} \text{trace}(G^T S_w G) &\leq \text{trace}(S_w) \\ \text{trace}(G^T S_b G) &\leq \text{trace}(S_b), \end{aligned}$$

which are satisfied for any G with orthonormal columns, since S_w and S_b are positive semidefinite. In summary, the individual traces of S_m , S_w , and S_b are preserved by using PCA to reduce to a dimension of at least $\text{rank}(H_m)$, or by using LSI to reduce to a dimension of at least $\text{rank}(A)$.

4 LSI Plus LDA

In this section, we establish the equivalence of the LDA/GSVD method to a two-stage approach composed of LSI followed by LDA, and denoted by LSI + LDA. As discussed in the previous section, for $q = \text{rank}(A)$, we can write $A = U_q \Sigma_q V_q^T$, where U_q and V_q denote the first q columns of U and V , respectively, and $\Sigma_q = \Sigma(1 : q, 1 : q)$. Then the q -dimensional representation of A after the LSI stage is

$$B = U_q^T A,$$

and the second stage applies LDA to B . Letting the superscript B denote matrices after the LSI stage, we have

$$H_b^B = U_q^T H_b \quad \text{and} \quad H_w^B = U_q^T H_w.$$

Hence

$$S_b^B = U_q^T H_b H_b^T U_q \quad \text{and} \quad S_w^B = U_q^T H_w H_w^T U_q.$$

Suppose

$$S_b^B x = \lambda S_w^B x;$$

i.e. x and λ are an eigenvector-eigenvalue pair of the generalized eigenvalue problem that LDA solves in the second stage. Then, for $\lambda = \alpha^2/\beta^2$,

$$\beta^2 U_q^T H_b H_b^T U_q x = \alpha^2 U_q^T H_w H_w^T U_q x.$$

Suppose the matrix (U_q, U'_q) is orthogonal. Then $(U'_q)^T A = (U'_q)^T U_q \Sigma_q V_q^T = 0$, and accordingly, $(U'_q)^T H_b = 0$ and $(U'_q)^T H_w = 0$, since the columns of both H_b and H_w are linear combinations of the columns of A . Hence

$$\begin{aligned} \beta^2 \begin{pmatrix} U_q^T \\ (U'_q)^T \end{pmatrix} H_b H_b^T U_q x &= \begin{pmatrix} \beta^2 U_q^T H_b H_b^T U_q x \\ 0 \end{pmatrix} \\ &= \begin{pmatrix} \alpha^2 U_q^T H_w H_w^T U_q x \\ 0 \end{pmatrix} \\ &= \alpha^2 \begin{pmatrix} U_q^T \\ (U'_q)^T \end{pmatrix} H_w H_w^T U_q x, \end{aligned}$$

which implies

$$\beta^2 H_b H_b^T (U_q x) = \alpha^2 H_w H_w^T (U_q x).$$

That is, $U_q x$ and α/β are a generalized singular vector and value of the generalized singular value problem that LDA solves when applied to A . To show that these $U_q x$ vectors include *all* the LDA solution vectors for A , we show that $\text{rank}(S_m^B) = \text{rank}(S_m)$. From the definition (2.6), we have

$$H_m = A - ce^T = A(I - \frac{1}{n} ee^T) = U_q \Sigma_q V_q^T (I - \frac{1}{n} ee^T)$$

and

$$H_m^B = U_q^T H_m,$$

and hence

$$H_m = U_q H_m^B.$$

Since H_m and H_m^B have the same null space, their ranks are the same. This means that the number of non-arbitrary generalized singular value pairs is the same for LDA/GSVD

applied to B , which produces $t = \text{rank}(S_m^B)$ pairs, and LDA/GSVD applied to A , which produces $t = \text{rank}(S_m)$ pairs.

So if G is an optimal LDA transformation for B , which is the q -dimensional representation of the matrix A via LSI, then $U_q G$ is an optimal LDA transformation for A . In other words, LDA applied to A produces

$$Y = (U_q G)^T A = G^T U_q^T A = G^T B,$$

which is the same result as applying LSI to reduce the dimension to q , followed by LDA. Finally, we note that if the dimension after the LSI stage is at least $\text{rank}(A)$, that is $B = U_l^T A$ for $l \geq q$, the equivalency argument remains unchanged.

5 PCA Plus LDA

Just as we did in the previous section for LSI, we now show that a two-stage approach in which PCA is followed by LDA is equivalent to LDA applied directly to A . In this case, for $p = \text{rank}(H_m)$, the p -dimensional representation of A after the PCA stage is

$$B = \tilde{U}_p^T A,$$

and the second stage applies LDA/GSVD to B . Letting the superscript B denote matrices after the PCA stage, we have

$$H_b^B = \tilde{U}_p^T H_b \quad \text{and} \quad H_w^B = \tilde{U}_p^T H_w.$$

Hence

$$S_b^B = \tilde{U}_p^T H_b H_b^T \tilde{U}_p \quad \text{and} \quad S_w^B = \tilde{U}_p^T H_w H_w^T \tilde{U}_p.$$

Suppose

$$S_b^B x = \lambda S_w^B x;$$

i.e. x and λ are an eigenvector-eigenvalue pair of the generalized eigenvalue problem that LDA solves in the second stage. Then, for $\lambda = \alpha^2/\beta^2$,

$$\beta^2 \tilde{U}_p^T H_b H_b^T \tilde{U}_p x = \alpha^2 \tilde{U}_p^T H_w H_w^T \tilde{U}_p x.$$

Suppose the matrix $(\tilde{U}_p, \tilde{U}'_p)$ is orthogonal. Then

$$\begin{aligned} (\tilde{U}'_p)^T H_m = 0 &\Rightarrow (\tilde{U}'_p)^T S_m \tilde{U}'_p = 0 \\ &\Rightarrow (\tilde{U}'_p)^T S_w \tilde{U}'_p + (\tilde{U}'_p)^T S_b \tilde{U}'_p = 0 \\ &\Rightarrow (\tilde{U}'_p)^T S_w \tilde{U}'_p = 0 \quad \text{and} \quad (\tilde{U}'_p)^T S_b \tilde{U}'_p = 0 \\ &\Rightarrow (\tilde{U}'_p)^T H_w = 0 \quad \text{and} \quad (\tilde{U}'_p)^T H_b = 0. \end{aligned}$$

Here we use (2.7), the scatter matrix relationship (1.2), and

the fact that S_w and S_b are positive semidefinite. Hence

$$\begin{aligned} \beta^2 \begin{pmatrix} \tilde{U}_p^T \\ (\tilde{U}'_p)^T \end{pmatrix} H_b H_b^T \tilde{U}_p x &= \begin{pmatrix} \beta^2 \tilde{U}_p^T H_b H_b^T \tilde{U}_p x \\ 0 \end{pmatrix} \\ &= \begin{pmatrix} \alpha^2 \tilde{U}_p^T H_w H_w^T \tilde{U}_p x \\ 0 \end{pmatrix} \\ &= \alpha^2 \begin{pmatrix} \tilde{U}_p^T \\ (\tilde{U}'_p)^T \end{pmatrix} H_w H_w^T \tilde{U}_p x, \end{aligned}$$

which implies

$$\beta^2 H_b H_b^T (\tilde{U}_p x) = \alpha^2 H_w H_w^T (\tilde{U}_p x).$$

That is, $\tilde{U}_p x$ and α/β are a generalized singular vector and value of the generalized singular value problem that LDA solves when applied to A . As in the previous section, we need to show that we obtain *all* the LDA solution vectors for A in this way. From

$$(5.13) \quad S_m^B = \tilde{U}_p^T S_m \tilde{U}_p = \tilde{\Sigma}_p^2,$$

we have that LDA/GSVD applied to B produces $\text{rank}(S_m^B) = p$ non-arbitrary generalized singular value pairs. That is the same number of non-arbitrary pairs as LDA/GSVD applied to A .

So if G is an optimal LDA transformation for B , which is the p -dimensional representation of the matrix A via PCA, then $\tilde{U}_p G$ is an optimal LDA transformation for A . In other words, LDA applied to A produces

$$Y = (\tilde{U}_p G)^T A = G^T \tilde{U}_p^T A = G^T B,$$

which is the same result as applying PCA to reduce the dimension to p , followed by LDA. Note that if the dimension after the PCA stage is at least $\text{rank}(H_m)$, that is $B = \tilde{U}_l^T A$ for $l \geq p$, the equivalency argument remains unchanged.

An additional consequence of (5.13) is that

$$\text{null}(S_m^B) = \{0\}.$$

Due to the relationship (1.2) and the fact that S_w and S_b are positive semidefinite,

$$\text{null}(S_m^B) = \text{null}(S_w^B) \cap \text{null}(S_b^B).$$

Thus the PCA stage eliminates only the joint null space, as illustrated in Figure 2, which is why we don't lose any discriminatory information before applying LDA.

6 QRD Plus LDA

To simplify the computation in the first stage, we note that the same argument holds if we use the reduced QR decomposition [6]

$$A = QR,$$

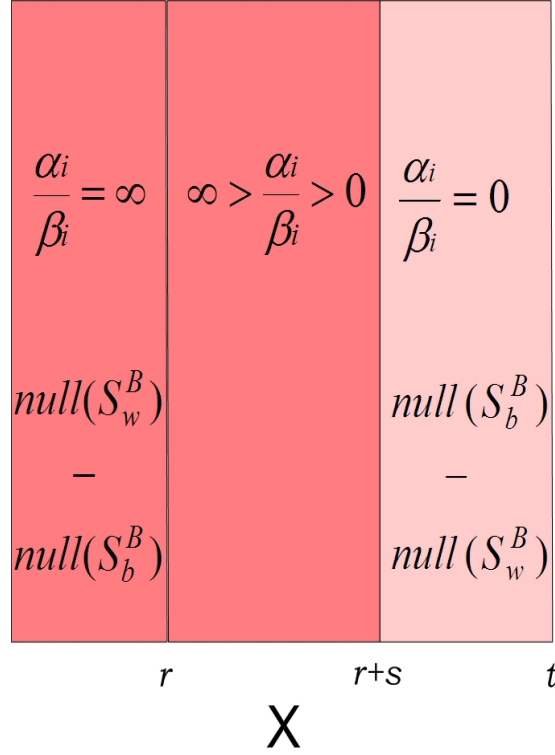


Figure 2: Generalized singular vectors and their corresponding generalized singular values when applying LDA/GSVD to B .

where $Q \in \mathbb{R}^{m \times n}$ and $R \in \mathbb{R}^{n \times n}$, and let Q play the role that U_q or \tilde{U}_p played before. That is, we use the reduced QR decomposition instead of the SVD.

The n -dimensional representation of A after the QRD stage is

$$B = Q^T A,$$

and the second stage applies LDA to B . Once again letting the superscript B denote matrices after the first stage, we have

$$H_b^B = Q^T H_b \quad \text{and} \quad H_w^B = Q^T H_w.$$

Hence

$$S_b^B = Q^T H_b H_b^T Q \quad \text{and} \quad S_w^B = Q^T H_w H_w^T Q.$$

Suppose

$$S_b^B x = \lambda S_w^B x;$$

i.e. x and λ are an eigenvector-eigenvalue pair of the generalized eigenvalue problem that LDA solves in the second stage. Then, for $\lambda = \alpha^2/\beta^2$,

$$\beta^2 Q^T H_b H_b^T Q x = \alpha^2 Q^T H_w H_w^T Q x.$$

Suppose the matrix (Q, Q') is orthogonal. Then $(Q')^T A = (Q')^T QR = 0$, and accordingly, $(Q')^T H_b = 0$ and $(Q')^T H_w = 0$, since the columns of both H_b and H_w are linear combinations of the columns of A . Hence

$$\begin{aligned} \beta^2 \begin{pmatrix} Q^T \\ (Q')^T \end{pmatrix} H_b H_b^T Qx &= \begin{pmatrix} \beta^2 Q^T H_b H_b^T Qx \\ 0 \end{pmatrix} \\ &= \begin{pmatrix} \alpha^2 Q^T H_w H_w^T Qx \\ 0 \end{pmatrix} \\ &= \alpha^2 \begin{pmatrix} Q^T \\ (Q')^T \end{pmatrix} H_w H_w^T Qx, \end{aligned}$$

which implies

$$\beta^2 H_b H_b^T (Qx) = \alpha^2 H_w H_w^T (Qx).$$

That is, Qx and α/β are a generalized singular vector and value of the generalized singular value problem that LDA solves when applied to A . Similar to our argument for LSI, we can show that we obtain *all* the LDA solution vectors for A in this way, by writing

$$H_m^B = Q^T H_m \quad \text{and} \quad H_m = Q H_m^B.$$

So if G is an optimal LDA transformation for B , which is the n -dimensional representation of the matrix A after QRD, then QG is an optimal LDA transformation for A . In other words, LDA applied to A produces

$$Y = (QG)^T A = G^T Q^T A = G^T B,$$

which is the same result as applying QRD to reduce the dimension to n , followed by LDA.

7 Experimental Results

To confirm our theoretical results, we use five categories of abstracts from the MEDLINE¹ database (see Table 1). Each category has 40 documents. There are 7519 terms after preprocessing with stemming and removal of stop words [9]. Since 7519 exceeds the number of documents (200), S_w is singular and classical discriminant analysis breaks down. However, LDA/GSVD and the equivalent two-stage methods circumvent this singularity problem.

Table 2 confirms the preservation of the traces of individual scatter matrices upon dimension reduction by the methods we use in the first stage. Specifically, since $\text{rank}(A) = 198$, using LSI to reduce the dimension to 198 preserves the values of $\text{trace}(S_w)$ and $\text{trace}(S_b)$ from the full space. Likewise, PCA reduction to $\text{rank}(H_m) = 197$ and QRD reduction to $n = 200$ preserve the individual traces. The effect of these first stages is further illustrated by the lack of significant differences in misclassification rates

Table 1: 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

Table 2: Traces and Misclassification Rate (in %) with L_2 norm similarity

Method	Full	LSI	PCA	QRD
Dim	7519 × 200	198 × 200	197 × 200	200 × 200
trace(S_w)	73048	73048	73048	73048
trace(S_b)	6229	6229	6229	6229
centroid	5%	5%	5%	5%
1NN	40%	40%	40%	41%
3NN	51%	52%	51%	52%

resulting from each method, as compared to the full space. Here we use a centroid-based classification method [12], which assigns a document to the cluster to whose centroid it is closest, and K Nearest Neighbor classification [15] for $K=1$ and $K=3$. Closeness is determined by L_2 norm or Euclidean distance.

To confirm the equivalence of the two-stage methods and the single-stage LDA/GSVD, we report trace values and misclassification rates for these in Table 3. Since S_w is singular, we cannot compute $\text{trace}(S_w^{-1} S_b)$ of the J_1 criterion. However, we observe that $\text{trace}(S_w)$ and $\text{trace}(S_b)$ are identical for LDA/GSVD and each two-stage method, and they sum to the final reduced dimension of $k - 1 = 4$. Classification results after dimension reduction by each method do not differ significantly, whether obtained by centroid-based or KNN classification.

8 Conclusion

To address the problem of dimension reduction of very high-dimensional or undersampled data, we have compared four seemingly different methods. Our results are summarized in Table 4, where $q = \text{rank}(A)$, $p = \text{rank}(H_m)$, and the complete orthogonal decomposition is referred to as URV. After showing that both LSI and PCA maximize $J_2(G) = \text{trace}(G^T S_m G)$ over all G with $G^T G = I$, we confirmed the preservation of $\text{trace}(S_w)$ and $\text{trace}(S_b)$ with either method or the computationally simpler QRD. The most significant results show the equivalence of the single-stage

¹<http://www.ncbi.nlm.nih.gov/pubmed>

Table 3: Traces and Misclassification Rate (in %) with L_2 norm similarity

Method	LDA/ GSVD	LSI→ 198 + LDA/GSVD	PCA→ 197 + LDA/GSVD	QRD→ 200 + LDA/GSVD
Dim	4×200	4×200	4×200	4×200
trace(S_w)	0.05	0.05	0.05	0.05
trace(S_b)	3.95	3.95	3.95	3.95
centroid	1%	1%	1%	1%
1NN	1%	1%	1%	1%
3NN	1.5%	1.5%	1.5%	1%

LDA/GSVD, which extends the applicability of the criterion $J_1(G) = \text{trace}((G^T S_w G)^{-1} G^T S_b G)$ to singular S_w , to any of the two-stage methods. This provides theoretical justification for the increasingly common approach of either LSI + LDA or PCA + LDA, although most studies have reduced the intermediate dimension below that required for equivalence.

Regardless of which of the three approaches is taken in the first stage, LDA/GSVD provides both a method for circumventing the singularity that occurs in the second stage, and a mathematical framework for understanding the singular case. When applied to the reduced representation in the second stage, the solution vectors correspond one-to-one with those obtained using the single-stage LDA/GSVD. Hence the second stage is a straightforward application of LDA/GSVD to a smaller representation of the original data matrix. Given the relative expense of LDA/GSVD and the two-stage methods, we observe that, in general, QRD is a significantly cheaper pre-processing step for LDA/GSVD than either LSI or PCA. However, if $\text{rank}(A) \ll n$, LSI may be cheaper than the reduced QR decomposition, and will avoid the centering of the data required in PCA. Therefore, the appropriate two-stage method provides a fast algorithm for LDA/GSVD.

Table 4: Comparison of Two-stage Methods for LDA

Method	LDA/ GSVD	LSI→ q + LDA/GSVD	PCA→ p + LDA/GSVD	QRD→ n + LDA/GSVD
Stage 1		max trace(S_m)	max trace(S_m)	max trace(S_m)
Stage 2	max tr($S_w^{-1} S_b$)	max tr($S_w^{-1} S_b$)	max tr($S_w^{-1} S_b$)	max tr($S_w^{-1} S_b$)
cost	URV of $(H_b, H_w)^T$	thin SVD of A	thin SVD of $A - ce^T$	reduced QRD of A

References

- [1] M.W. Berry, S.T. Dumais, and G.W. O'Brien. Using linear algebra for intelligent information retrieval. *SIAM Review*, 37(4):573–595, 1995.
- [2] P.N. Belhumeur, J.P. Hespanha, and D.J. Kriegman. Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):711–720, 1997.
- [3] S. Deerwester, S. Dumais, G. Furnas, T. Landauer, and R. Harshman. Indexing by latent semantic analysis. *J. American Society for Information Science*, 41:391–407, 1990.
- [4] R. Duda, P. Hart, and D. Stork. *Pattern Classification*. John Wiley & Sons, Inc., second edition, 2001.
- [5] K. Fukunaga. *Introduction to Statistical Pattern Recognition*. Academic Press, second edition, 1990.
- [6] G.H. Golub and C.F. Van Loan. *Matrix Computations*. Johns Hopkins University Press, third edition, 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 Anal. Appl.*, 25(1):165–179, 2003.
- [8] A.K. Jain and R.C. Dubes. *Algorithms for Clustering Data*. Prentice Hall, 1988.
- [9] G. Kowalski. *Information Retrieval Systems : Theory and Implementation*. Kluwer Academic Publishers, 1997.
- [10] C.L. Lawson and R.J. Hanson. *Solving Least Squares Problems*. SIAM, 1995.
- [11] J. Ortega. *Matrix Theory: A Second Course*. Plenum Press, 1987.
- [12] H. Park, M. Jeon, and J.B. Rosen. Lower dimensional representation of text data based on centroids and least squares. *BIT Numerical Mathematics*, 42(2):1–22, 2003.
- [13] C.C. Paige and M.A. Saunders. Towards a generalized singular value decomposition. *SIAM J. Numer. Anal.*, 18(3):398–405, 1981.
- [14] D.L. Swets and J. Weng. Using discriminant eigenfeatures for image retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(8):831–836, 1996.
- [15] S. Theodoridis and K. Koutroumbas. *Pattern Recognition*. Academic Press, 1999.
- [16] K. Torkkola. Linear discriminant analysis in document classification. In *IEEE ICDM Workshop on Text Mining*, 2001.
- [17] C.F. Van Loan. Generalizing the singular value decomposition. *SIAM J. Numer. Anal.*, 13(1):76–83, 1976.
- [18] J. Yang and J.Y. Yang. Why can LDA be performed in PCA transformed space? *Pattern Recognition*, 36(2):563–566, 2003.