Penalized Orthogonal Iteration for Sparse Estimation of Generalized Eigenvalue Problem

Sungkyu Jung a, Jeongyoun Ahn b, and Yongho Jeon c

aDepartment of Statistics, Seoul National University, Seoul, South Korea; bDepartment of Statistics, University of Georgia, Athens, GA; cDepartment of Applied Statistics, Yonsei University, Seoul, South Korea

ABSTRACT
We propose a new algorithm for sparse estimation of eigenvectors in generalized eigenvalue problems (GEPs). The GEP arises in a number of modern data-analytic situations and statistical methods, including principal component analysis (PCA), multiclass linear discriminant analysis (LDA), canonical correlation analysis (CCA), sufficient dimension reduction (SDR), and invariant coordinate selection. We propose to modify the standard generalized orthogonal iteration with a sparsity-inducing penalty for the eigenvectors. To achieve this goal, we generalize the equation-solving step of orthogonal iteration to a penalized convex optimization problem. The resulting algorithm, called penalized orthogonal iteration, provides accurate estimation of the true eigenspace, when it is sparse. Also proposed is a computationally more efficient alternative, which works well for PCA and LDA problems. Numerical studies reveal that the proposed algorithms are competitive, and that our tuning procedure works well. We demonstrate applications of the proposed algorithm to obtain sparse estimates for PCA, multiclass LDA, CCA, and SDR. Supplementary materials for this article are available online.

1. Introduction
Many statistical problems can be cast into the mathematical framework of a generalized eigenvalue problem (GEP). In particular, we focus on the symmetric-definite GEP, posed as follows. Suppose that $A \in \mathbb{R}^{p \times p}$ is a symmetric matrix, and $B \in \mathbb{R}^{p \times p}$ is a symmetric positive-definite matrix. While specific statistical contexts determine the exact nature of $A$ and $B$, a solution of GEP is given by a $d$-dimensional subspace $\mathcal{U}_d$ that is spanned by the generalized eigenvectors $u_1, \ldots, u_d$ corresponding to the $d$ largest generalized eigenvalues. The following equations define a generalized eigen-pair $(\lambda_j, u_j)$

$$Au_j = \lambda_j Bu_j,$$

where the generalized eigenvalues, $\lambda_1 \geq \cdots \geq \lambda_d$, satisfy $\lambda_j = u_j^T A u_j / u_j^T B u_j$. The generalized eigenvectors are orthogonal with respect to $B$, that is, $u_i^T B u_j = 1$ for $i = j$, and 0 for $i \neq j$. In the special case of $B = I_p$, the GEP is the standard eigenvalue problem.

Immediate applications of the GEP are to multivariate analysis problems, including the principal component analysis (PCA), canonical correlation analysis (CCA), multiclass linear discriminant analysis (LDA), invariant coordinate selection (Tyler et al. 2009) and sufficient dimension reduction (SDR, Li 2007). The GEP also appears frequently in nonlinear dimension reduction (Kokiopoulou, Chen, and Saad 2011) and in computer vision and image processing (Zhang, Chow, and Zhao 2013). We refer to the online supplementary materials for detailed description of some selected statistical GEP problems.

When $p$ is large, it is often desirable to find a sparse representation of $\mathcal{U}_d$, so that its basis vectors are sparse, that is, the vectors have many zero loadings in their entries. We propose an efficient algorithm for estimating sparse generalized eigenvectors from noisy observations of $A$ and $B$. It is well-known that in the standard eigenvalue problem, for example, for PCA, the standard eigenvectors of the sample covariance matrix $A$ (while $B = I_p$) is inconsistent with the true principal directions for large $p$ (Johnstone and Lu 2009; Jung and Marron 2009). Thus, in the high-dimension, low-sample-size situations, sparsity-inducing methods have a potential in improving the estimation accuracy, as well as in providing interpretable eigenvectors through variable selection.

Our approach to obtain a sparse solution of the GEP is to extend the generalized orthogonal iteration, a standard numerical method of solving the GEP (Golub and Van Loan 1996). The generalized orthogonal iteration consists of iterating two steps: one involving solutions of linear equations, and an orthogonalization step. One of our main ideas is to transform the equation-solving step to a minimization of a quadratic objective function, so that a sparsity-inducing penalty can be easily incorporated. We propose to use $\ell_1$-norm or $\ell_2,1$-norm penalty on the eigenvector matrix to induce element-wise or coordinate-wise sparsity. These penalty functions are those used in lasso and group-lasso regressions (see Hastie, Tibshirani, and Friedman 2009). The proposed method is called penalized orthogonal iteration (POI). We use the block coordinate descent algorithm in solving the penalized minimization problem at each iteration. We also study a computationally efficient alternative to POI.
that comes down to solving just one minimization problem. We establish sufficient conditions under which the solutions of this alternative method, called fast POI, correspond to the solutions of the GEP. The conditions are satisfied under the situations for PCA and multiclass LDA.

The solutions of POI and fast POI depend on the choice of a tuning parameter, dictating the degrees of penalization. Larger values of the tuning parameter result in more sparse solutions of the generalized eigenvectors. When the tuning parameter is zero, POI becomes the generalized orthogonal iteration. An eigenvalue-based cross-validation procedure is proposed and is seen to work well in numerical studies.

To the best of authors’ knowledge, there have been only a few proposals for sparse GEP in the literature (Sriperumbudur, Torres, and Lanckriet 2011; Song, Babu, and Palomar 2015; Tan et al. 2016; Gaynanova, Booth, and Wells 2017; Han and Clemmensen 2016; Safo et al. 2018). We briefly discuss their approaches and limitations, compared to our proposal.

Since finding the largest eigenvalue $\lambda_1$ that solves (1) is equivalent to maximizing $\bf u^T A u$ subject to $\bf u^T B u = 1$, Sriperumbudur, Torres, and Lanckriet (2011) formulated both an $\ell_0$-constrained GEP and an $\ell_0$-penalized GEP

$$\max_{\bf u} \bf u^T A u, \quad \text{subject to} \quad \bf u^T B u = 1, \quad \|\bf u\|_0 \leq s, \quad \text{(2)}$$

$$\max_{\bf u} \bf u^T A u - \rho \|\bf u\|_0, \quad \text{subject to} \quad \bf u^T B u = 1, \quad \text{(3)}$$

for $1 \leq s \leq p$ and $\rho > 0$, where $\|\bf u\|_0$ is the number of nonzero elements of $\bf u$. While Sriperumbudur, Torres, and Lanckriet (2011) only proposed an algorithm to solve a variant of (3), Song, Babu, and Palomar (2015) later proposed several approximate solutions of (3). Recently, Tan et al. (2016) proposed to solve (2), by truncating the steepest ascent iterates in maximizing the Rayleigh coefficient $\bf u \mapsto \bf u^T A u/\bf u^T B u$. Gaynanova, Booth, and Wells (2017) pointed out a natural difference between the penalized and constrained optimizations for sparse GEP, similar to (2) and (3) but with $\ell_1$-norm. Safo et al. (2018) proposed to estimate $\bf u$ via minimizing $\|\bf u\|_1$ subject to a constraint $\|\bf A u - \hat{\lambda}_n \bf B u\|_\infty \leq \rho$, where $\hat{\lambda}_n$ is the nonsparse solution of (1). As it is evident in Sriperumbudur, Torres, and Lanckriet (2011), Song, Babu, and Palomar (2015), and Tan et al. (2016) who limit themselves for solving only one eigen-pair, we are unclear how (2) or (3) generalizes to simultaneously solving for multiple eigenvectors, $\bf u_1, \ldots, u_d$. When multiple eigenvectors are needed, as is typical in practice, these methods are not readily applicable, at least not without a clever modification. Our algorithm is designed to estimate $\bf u_1, \ldots, u_d$ altogether, and works well when $d > 1$.

Han and Clemmensen (2016) assumed $\bf B$ to be positive definite, and transformed the GEP to a regular eigen-decomposition of $\bf B^{-1} A$ (or $\bf B^{-1/2} A \bf B^{-1/2}$) while applying an $\ell_1$-penalty to achieve sparsity. However, their method is not directly applicable to the large-$p$-small-$n$-case, due to the numerically unstable inverse of the large matrix $\bf B$. They used alternating direction method of multipliers for optimization, which causes their method to be computationally expensive.

Chen, Zou, and Cook (2010) proposed to solve sufficient dimension reduction (SDR) problems by maximizing $\text{trace} (\bf U^T A U) - \rho_s (\bf U)$, for $\bf U \in \mathbb{R}^{p \times d}$ satisfying $\bf U^T B U = I_d$, in which the penalty function $\rho_s$ enforces coordinate-wise sparsity (10). While Chen, Zou, and Cook’s (2010) formulation is similar to our fast POI with (10), their computation is much slower than any of our proposed algorithms, perhaps due to using both penalization and constraint.

Our proposed algorithms provide sparse solutions of the original GEP (1), produces any number of eigenpairs simultaneously, is computationally efficient even for high-dimensional data, and is applicable to a number of statistical problems including PCA, LDA, CCA, SDR, and invariant coordinate selection.

The rest of the paper is organized as follows. In Section 2, we introduce the proposed sparse GEP methodology. Numerical algorithms are discussed in Section 3. We demonstrate applications of our proposal to a number of statistical problems including PCA, LDA, SDR, and CCA in Section 4, in which some of the most promising competitors are numerically compared. All proofs are contained in the Appendix. The online supplementary material contains additional numerical results.

2. Methodology

2.1. Setting

In most applications, the population matrices $\bf A$ and $\bf B$ in (1) are symmetric nonnegative definite, while $\bf B$ is often positive definite. For some applications, the rank of $\bf A$ is much smaller than the dimension $p$ of the matrices.

We are interested in estimating the generalized eigenvectors $\bf u_1, \ldots, u_d$ corresponding to the $d$ largest generalized eigenvalues, where $d \leq \text{rank}(\bf A)$. Our proposed methods estimate the subspace $U_d = \text{span}(\bf u_1, \ldots, u_d)$. In order for $U_d$ to be identifiable, one must assume the following on the size of the eigenvalues:

$$\lambda_1 \geq \cdots \geq \lambda_d > \lambda_{d+1} \geq \cdots \geq \lambda_p \geq 0.$$

Let $U_d = [\bf u_1, \ldots, u_d]$ and $\Lambda_d = \text{diag}(\lambda_1, \ldots, \lambda_d)$. The generalized eigenvectors are $\bf B$-orthogonal, and thus in general do not form an orthogonal basis of $U_d$. However, one can readily obtain an orthogonal basis of $U_d$ from $U_d$, and, conversely, obtain $U_d$ from any orthogonal basis of $U_d$. The detail follows. Throughout the paper, the notation $O(p,d)$ is used for the set of semionthogonal matrices: $O(p,d) = \{\bf X \in \mathbb{R}^{p \times d} : \bf X^T \bf X = I_d\}$. Let $O(d) = O(d,d)$ be the set of $d \times d$ orthogonal matrices.

The following equation is equivalent to (1)

$$\bf A U_d = B U_d \Lambda_d. \quad \text{(4)}$$

By the QR decomposition, we have $U_d = Q_d R_d$, for a $Q_d \in O(p,d)$, and for a $d \times d$ upper-triangular matrix $R_d$. Then the GEP in (4) is equivalently written as

$$\bf A Q_d = B Q_d \Lambda_d^* \quad \text{(5)}$$

where $\Lambda_d^* = R_d \Lambda_d R_d^{-1}$ is a $d \times d$ upper-triangular matrix. Note that the diagonal elements of $\Lambda_d^*$ are the same as those of $\Lambda_d$.

Now suppose we have an arbitrary basis matrix $\tilde{Q}_d \in O(p,d)$ such that $\text{span}(\tilde{Q}_d) = U_d$. The generalized eigenvectors satisfying (4) can be recovered, as follows:
Proposition 1. Assume that $\tilde{Q}_d$ is an arbitrary $p \times d$ orthogonal matrix with the same column space as $U_d$, where $(A_d, U_d)$ is the solution to (4). For $\tilde{A}_d = \tilde{Q}_d A \tilde{Q}_d$ and $\tilde{B}_d = \tilde{Q}_d^T B \tilde{Q}_d$, let $T$ and $D$, respectively, be the matrix of eigenvectors and the diagonal matrix of eigenvalues of the following GEP

$$\tilde{A}_d T = \tilde{B}_d TD$$

with $T^T \tilde{B}_d T = I_d$. Then span($U_d$) = span($\tilde{Q}_d T$). If the diagonal values of $D$ are distinct and in the decreasing order, then $U_d = \tilde{Q}_d T$ and $A_d = D$.

In the next subsection, we discuss our approaches in estimating an orthogonal basis of $U_d$, that is, $Q_d$ in (5), from noisy versions of $A$ and $B$. Proposition 1 can then be used to obtain the estimates of the generalized eigenvector and eigenvalue pair $(U_d, A_d)$. In practice the matrices $A$ and $B$ in (4) are replaced by their empirical counterparts computed from a sample. We treat $A$ and $B$ to be the empirical matrices, with which we attempt to solve the GEP. In most applications, the empirical matrices $A$ and $B$ are nonnegative definite by construction. We require $B$ to be positive definite. If not, we add a scaled identity matrix $\epsilon I_p$, for a small $\epsilon > 0$, and $B + \epsilon I_p$ is treated as $B$. We recommend using $\epsilon = \min(\log p/\text{rank}(B), \sigma_B/2)$ where the $\sigma_B$ is the smallest positive eigenvalue of $B$.

### 2.2. Proposed Method

We consider two penalized optimization approaches to obtain sparse solutions of the GEP. The first approach is a generalization of the widely used orthogonal iteration and can be applied to almost all situations. The second approach aims to provide efficient computation for some high-dimensional problems, such as PCA and multiclass LDA.

#### 2.2.1. Penalized Orthogonal Iteration

We begin by reviewing the standard generalized orthogonal iteration in solving the GEP (5). Given $(A, B)$ and an initial value $Q_0 \in O(p, d)$, the standard generalized orthogonal iteration (Golub and Van Loan 1996, sec. 8.7.3) finds $Q$, a solution of (5), by iterating the following two steps until convergence. For $r = 1, 2, \ldots$.

Step 1. Solve $B\tilde{Z}_r = A\tilde{Q}_r - 1$ for $\tilde{Z}_r$.

Step 2. Obtain $Q_r$ by QR decomposition $\tilde{Z}_r = \tilde{Q}_r \tilde{R}_r$.

The iteration stops when a distance between $Q_{r-1}$ and $Q_r$ is smaller than a threshold. We use the projection distance between the subspaces, defined later in (22).

The POI modifies Step 1 so that it can be cast into a convex optimization framework with a sparsity-inducing penalty. Let $p_\lambda(Z)$ be a penalty function on the matrix $Z$. Here, $\lambda \geq 0$ represents the degrees of penalization, yielding $p_\lambda(Z) = 0$ if $\lambda = 0$. We propose to replace Step 1 by

$$\hat{Z}_r = \min_{Z \in \mathbb{R}^{p \times d}} \left\{ \text{trace} \left( \frac{1}{2} Z'BZ - Z'A\tilde{Q}_{r-1} \right) + p_\lambda(Z) \right\}.$$  (7)

Note that if $\lambda = 0$, then $\hat{Z}_r$ of (7) is the solution of the original equation

$$B\hat{Z}_r = A\tilde{Q}_{r-1}.$$  (8)

The POI optimization problem (7) is motivated by the fact that the linear equation system (8) is the first-order condition of a quadratic optimization problem (7) without the penalty term. A natural choice for the sparsity-inducing penalty function $p_\lambda(Z)$ in (7) is a lasso penalty (Tibshirani 1996),

$$p_\lambda(Z) = \sum_{j=1}^d \lambda_j \| z_j \|_1,$$  (9)

where $Z = [z_1, \ldots, z_d]$ and $\lambda_j > 0$. While (9) generally produces element-wise sparse solutions, it may be more reasonable to assume that there exists a small subset of coordinate indices that are relevant to all of the $d$ largest generalized eigenvalues. For this purpose, we use the $\ell_{2,1}$-norm penalty for $p_\lambda(Z)$, also known as the group-lasso penalty (Yuan and Lin 2006). Let $z_i^*$ be the $i$th row of $Z (i = 1, \ldots, p)$. Then we set for $\lambda > 0$,

$$p_\lambda(Z) = \lambda \sum_{i=1}^p \| z_i \|_2.$$  (10)

We have implemented the POI with the penalty (9) for elementwise sparsity and (10) for coordinate-wise sparsity. Other choices of penalty functions can be used as well (see, e.g., Tibshirani 2011).

In search of a sparse subspace basis, coordinate-wise sparsity is preferred to element-wise sparsity, since any element-wise sparse basis matrix $Q$ is in general no longer element-wise sparse if arbitrarily rotated, for example, in the orthogonalization step of the POI. In contrast, $QV$ for any $V \in O(d)$ is coordinate-wise sparse as long as $Q$ is coordinate-wise sparse. Since the basis of a subspace can only be coordinate-wise sparse, utilizing the coordinate-wise sparsity has a clear advantage in variable selection and its interpretation (see Bouveyron, Latouche, and Mattei 2016).

Remark 1. In the PCA context (i.e., $B = I_p$ and $A = \tilde{\Sigma}$, the sample covariance matrix), the POI overlaps with the sparse PCA algorithm of Ma (2013), who proposed to add a thresholding step to the orthogonal iteration for the standard eigenvalue problem. In particular, if the penalty function $p_\lambda$ is given by an $\ell_{2,1}$-norm, then our method coincides with Ma’s method with soft-thresholding.

#### 2.2.2. Fast POI

A computationally simpler alternative to the POI is to solve an alternative form of (7) just once. For this, we propose to replace $A\tilde{Q}_{r-1}$ in (7) with the $p \times d$ matrix, denoted by $V$, whose columns contain the eigenvectors of $A$ that correspond to the $d$ largest eigenvalues of $A$. This results in the following

$$\hat{Z} = \min_{Z \in \mathbb{R}^{p \times d}} \left\{ \text{trace} \left( \frac{1}{2} Z'BZ - Z'V \right) + p_\lambda(Z) \right\}.$$  (11)

The orthogonal basis matrix $\hat{Q}$ is obtained by the QR decomposition of $\hat{Z}$. There is no “outer” iteration for this approach, which yields much faster computation. The penalty function $p_\lambda(Z)$ can be chosen to be either (9) or (10). The solutions from this approach is referred to as fast POI solutions.

Note that the POI solves the GEP in (5) if $\lambda = 0$. On the other hand, the fast POI solution with $\lambda = 0$ does not in general
solve the original GEP. The following proposition identifies a sufficient condition under which the fast POI solution with no penalty solves the original GEP.

**Proposition 2.** Suppose \( A \) and \( B \) are both \( p \times p \) symmetric nonnegative matrices. For any \( 1 \leq d < p \), let \( V \in \mathcal{O}(p, d) \) be the eigenvector-matrix of \( A \) corresponding to the \( d \) largest eigenvalues of \( A \), and \( Q \in \mathcal{O}(p, d) \) satisfy the generalized eigen-equation (5). If

(a) \( B = I_p \), or
(b) \( B \) is positive definite and \( \text{rank}(A) = d \),

then the column space of \( B^{-1}V \) is exactly the column space of \( Q \).

The condition (a) corresponds to the standard eigenvalue problem, arising in the context of PCA. The rank condition in (b) is satisfied when there are \( K = d + 1 \) groups in a multiclass LDA problem.

In our experience, obtaining a fast POI solution requires only a fraction of time that is needed for the corresponding POI solution. This is because POI typically requires many iterations until convergence. Despite its quick computing time, we found that fast POI in the problems of PCA and multiclass LDA provides good approximations of the POI solutions.

**Remark 2.** After we formulated the fast POI criterion, we found that, in the special case of multiclass LDA problems, our minimization approach (11) is similar to the sparse LDA methods of Mai, Yang, and Zou (2017) and Gaynanova, Booth, and Wells (2016). Using the notation used in (11), Mai et al’s solutions are given by replacing \( V \) with the mean difference matrix with columns \( \mu_k - \mu_1, k = 2, \ldots, K \); Gaynanova et al’s approach is equivalent to (11) if \(-Z'V \) is replaced by \( \frac{1}{2}Z'AZ = Z'D \), where \( D \) consists of weighted mean differences \( \mu_i - \mu_j \). Both approaches are dependent on the order of group labels and, consequently, their solutions change if different label orders are used. In Gaynanova et al’s approach, the inclusion of \( \frac{1}{2}Z'AZ \) is to ensure the objective function to be bounded below. Since we use \( B + \epsilon I_p \), if \( B \) is not full rank, the objective function (11) is bounded. These methods are further compared numerically in Section 4.2.

### 2.3. Eigenvalue Estimation

Once \( \hat{Q} \) is obtained either by POI or fast POI, one might be interested in estimating the corresponding generalized eigenvalues. We illustrate two approaches here. First, one can estimate the generalized eigenvalues by directly using **Proposition 1**. Let \( \hat{T} \) and \( \hat{D} \) be the solution to the GEP (6) with \( Q \) replaced by \( \hat{Q} \). Then the plug-in estimate of the eigenvector matrix is \( \hat{U} = \hat{Q} \hat{T} \), and the eigenvalue matrix estimate is \( \hat{A} = \hat{D} \). If the elements of \( A \) stand for covariances, then using **Proposition 1** is equivalent to estimating \( \lambda_j \) by the empirical variance contained in \( A \) projected on \( \hat{u}_j \), normalized by that of \( B \)

\[
\hat{\lambda}_j = \hat{u}_j A \hat{u}_j / \hat{u}_j B \hat{u}_j, \quad j = 1, \ldots, d. \tag{12}
\]

A more sophisticated approach is to use the eigen-equation (4) directly. For this, we estimate the generalized eigenvalue matrix \( A \) by the solution of the following minimization problem

\[
\hat{A} = \arg \min_A \left\| A \hat{U} - \hat{B} \hat{A} \right\|_F^2, \quad (A \text{ is diagonal}). \tag{13}
\]

This problem has a closed-form solution.

**Lemma 3.** Let \( \epsilon_i \) and \( \beta_j \) be the \( j \)th column vectors of \( \hat{A} \hat{U} \) and \( \hat{B} \hat{U} \). Then the solution \( \hat{A} \) of (13) is \( \hat{A} = \text{diag}(\hat{\lambda}_1, \ldots, \hat{\lambda}_d) \), where \( \hat{\lambda}_j = (\beta_j^T \beta_j)^{-1} \beta_j^T \alpha_j \).

The eigenvalue estimates \( \hat{A} \) of (12) and \( \hat{A} \) of (13) are in general different. They are the same when \( \hat{u}_i \) are the exact solutions of the GEP (4), which are generally nonsparse. In our experiments, these two estimates were numerically close to each other. We used (12) in all of our numerical analyses.

### 2.4. Tuning Parameter Selection

The tuning parameter \( \lambda \) in the penalty function plays an important role in the estimation. When \( \lambda = 0 \), our proposal provides nonsparse solutions, while the estimated subspaces tend to have more sparse bases as \( \lambda \) increases. In the following, we consider a general cross-validation approach that can be used for any regularized GEP. Note that for (9) we set \( \hat{\lambda}_j = \lambda \), for all \( j = 1, \ldots, d \).

Suppose that we have a dataset that can be split into two subsets that produce independent pairs \( (A^{(r)}, B^{(r)}) \), \( r = 1, 2 \). We use the data indexed by \( r = 1 \) to train the estimates \( \hat{U}_1 \) and \( \hat{A}_1 \) for various values of \( \lambda \). The data indexed by \( r = 2 \) is used to evaluate these estimates, to tune \( \lambda \). In particular, for a given \( \lambda \), we define the cross-validation score as

\[
\text{CV}(\lambda) = \text{trace} \left[ (\hat{U}_1^T B^{(2)} \hat{U}_1)^{-1} \hat{U}_1^T A^{(2)} \hat{U}_1 \right]. \tag{14}
\]

We choose the \( \lambda \) with the largest average cross-validation score, based on repeated random splits, or by predefining the training and tuning sets. Candidate values of \( \lambda \) are found in an interval \((0, \lambda_{\text{max}})\). The upper bound \( \lambda_{\text{max}} \) is set so that \( \hat{U}_1 = 0 \) for any \( \lambda > \lambda_{\text{max}} \), and depends on the data and the penalty term used. We defer the discussion on the choice of \( \lambda_{\text{max}} \) to Section 3 (see Remarks 3 and 4).

We interpret the CV score (14) as follows. Note that when \( B^{(2)} \) and \( A^{(2)} \) are replaced by \( B^{(1)} \) and \( A^{(1)} \), \( \text{CV}(\lambda) \) is simply the sum of estimated eigenvalues (12). Heuristically, maximizing \( \text{CV}(\lambda) \) is equivalent to finding the subspace with the largest sum of “prediction eigenvalues.” We delve further into the following data-analytic situations.

For PCA, \( \text{CV}(\lambda) \) represents the total variance contained in the \( d \)-dimensional eigenspace. To be specific, let \( X \) be the centered \( n \times p \) data matrix, then we set \( A^{(2)} = X^T X \) and \( B^{(2)} = I_p \), which in turn leads to

\[
\text{CV}(\lambda) = \text{trace} (\hat{U}_1^T A^{(2)} \hat{U}_1) = \text{trace} (X^T X) - \left\| X(I_p - \hat{U}_1 \hat{U}_1^T) \right\|_F^2,
\]

where the second term is sometimes called the reconstruction error of the subspace spanned by \( \hat{U}_1 \). Thus, maximizing
CV(λ) is equivalent to minimizing the reconstruction error, an approach that can be found in the literature on PCA (see, e.g., Shen and Huang 2008; Josse and Husson 2012). In our numerical studies, we found that CV(λ) is typically a concave function and is negatively correlated with the distance to the true eigenspace from the estimate associated with λ, that is, the larger CV(λ), the better the estimation. These are illustrated in Figure 1 (top two panels) with simulated data, further discussed in Section 4.

In a situation with multiple groups (see Section 4.2), we set $\Lambda^{(2)} = S_B$ as the between-group covariance, and $B^{(2)} = S_W$ as the within-group covariance, estimated from the tuning set. The groups are more clearly separated for larger $\hat{z}$-the subspaces spanned by $z$. We provide algorithms to solve the POI (7) when the penalty is $\lambda$. Specifically, the solution (7) is solved, with $\lambda = (\lambda)$.

In short, the algorithm updates $\lambda$ in a cyclic fashion. In particular, let $S_j(z) = \text{sign}(z)(|z| - \lambda)_+$. The $j$th coordinate $z_j$ of $z$ is updated by

$$z_j \leftarrow S_j \left( \frac{a_j^T q_j - b_j^T z + b_{jj} z_j}{b_{jj}} \right), \quad (16)$$

where $b_{jj}$ is the $jj$th diagonal element of $B$, and $a_j^T$ (or $b_j^T$) is the $j$th row of the symmetric matrix $A$ (or $B$, respectively).

**Remark 3.** For both POI and fast POI, setting too large $\lambda_j$ gives the trivial solution $z_j = 0$. It can be shown that $\lambda_j^* = \max_{i=1, \ldots, p} |a_i^T q_j|$ is the maximum value of $\lambda_j$ which gives a nontrivial solution of (15). However, $\lambda_j^*$ depends on the iterate $Q_{r-1}$. Since the solution $z_j$ of (15) is sparse for large $\lambda_j$, we take the maximum of $\lambda_j^*$ over the possible values of “sparse” $Q_{r-1}$ has only one nonzero entry. For $Q_{0}(p, d) = \{Q \in O(p, d) : \#(i, j) : q_{ij} \neq 0 = d\}$, where $a_{ij} = \delta_{(i, j)}$th element of $A$.

### 3.2. Algorithm for Coordinate-Wise Sparse Estimation

Since the problem (7) is convex and the penalty function (10) is block-separable, the block coordinate descent algorithm is guaranteed to converge to an optimal solution (Tseng 1993). Denote $q_j$ and $q_j^*$ for the $j$th column and $r$th row of $Q_{r-1}$, respectively. Define $z_j^* = [z_1^*, \ldots, z_j^*, \ldots, z_d^*]$ and $q_j^* = [q_1^*, \ldots, q_j^*, \ldots, q_p^*]$. Then,

$$\text{trace} \left( \frac{1}{2} Z' BZ - Z' A Q_{r-1} \right) = \sum_{j=1}^d \left( \frac{1}{2} z_j B z_j' - z_j A q_j^* \right)$$

$$= \frac{1}{2} \sum_{j=1}^d \sum_{j=1}^d z_j^* b_{jj} z_j - \sum_{i=1}^d z_j^* \delta_i,$$

(18)

where $a_{ij}$ (or $b_{ij}$) is the $(i, j)$th element of $A$ (or $B$, respectively) and $\delta_i = \sum_{j=1}^p a_{ij} q_j$. Using the fact that $B$ is symmetric, for the $g$th block with all $z_i, i \neq g$, fixed, the problem becomes

$$\min_{z_g} \left\{ \frac{1}{2} z_g^T b_{gg} z_g - z_g^T a_g + \lambda \|z_g\|_1 \right\},$$

(19)

where $a_g = \delta_g - \sum_{i \neq g} b_{gg} z_i$, and its solution is obtained by

$$\hat{z}_g = \left( b_{gg} + \lambda \|a_g\|_1 \right)^{-1} a_g.$$  

(20)

In short, the algorithm updates $Z$ in a cyclic fashion, with the initial value given by $Z = Q_{r-1}$. For $g = 1, \ldots, p$, the
Figure 1. The proposed tuning procedure works well for our simulated data. Top two panels are from PCA models; bottom three panels are from multiclass LDA models. * indicates the location of tuned \( \lambda \) that maximizes CV\( (\lambda) \). ◦ indicates the location of tuned \( \lambda \) that minimizes the tuning misclassification error rate.

Remark 4. It is easy to see that \( \lambda_0 = \max_{g=1,...,p} \| \delta_g \|_2 \) is the maximum value of \( \lambda \) that gives a nontrivial solution of (19).

Since \( \delta_g = \sum_{j=1}^{\rho} a_{gj} q_j \) depends on the iterate \( Q_{r-1} \) of the POI, we take

\[
\lambda_{\text{max}} = \max_{g=1,...,p} \left\{ \max_{Q \in \mathbb{C}_0(q,d)} \left\| \sum_{j=1}^{p} a_{gj} q_j \right\|_2 \right\},
\]  

(21)
which provides an upper bound for the maximum value of $\lambda$. Note that in (21) we used $Q \in \mathcal{O}_0(p, d)$ whose entrywise $L_0$-norm is $d$ (i.e., as sparse as possible). To simplify (21), for each fixed $g$, denote $a_{g,j}(i)$ for the $j$th largest element among the $g$th row of $A$, in the absolute value. Then

$$\max_{Q \in \mathcal{O}_0(p, d)} \left\| \sum_{j=1}^{p} a_{g,j} Q_j \right\|_2 = \left( \sum_{j=1}^{d} g_{j}^2 \right)^{1/2},$$

which in turn gives, for POI solutions,

$$\lambda_{\max} = \max_{g=1,\ldots,p} \left( \sum_{j=1}^{d} g_{j}^2 \right)^{1/2}.$$  

For an upper bound of the tuning parameter used in fast POI, it is straightforward to see that $\lambda_{\max} = \max_{g=1,\ldots,p} (\sum_{j=1}^{d} g_{j}^2)^{1/2}$, where $v_{ij}$ is the $(i,j)$th element of $V$.

4. Applications to Multivariate Analysis

In this section, we demonstrate the application of the proposed algorithms to PCA, multiclass LDA, SDR, and CCA.

4.1. Sparse Principal Component Analysis

The standard PCA amounts to solving the ordinary eigen-decomposition of the covariance matrix $\Sigma$. By setting $A = \Sigma$, $B = I_p$, the problem (1) becomes the ordinary eigen-decomposition problem, and the solution $(u_i, \lambda_i)$ corresponds to the principal component (PC) direction and variance pair.

We demonstrate the application of our method to sparse PCA, using a large-scale genomic data. The dataset we use is adopted from Ciriello et al. (2015), and consists of 272 cases of 1000 variables, where the first 500 variables are gene expression levels from the original dataset, and the latter 500 are normally distributed random noises. Since it is typical that in performing PCA one does not know which dimension $d$ to choose, we explore $d = 1\text{–}20$, and estimate the principal subspace of dimension $d$ using POI with coordinate-wise sparse penalty. A graphical summary of this study can be found in Figure 2. It can be seen that the eigenvalue and eigenvector estimates are stable across different choices of $d$, and our method has correctly screened out the noise variables for all choices of $d$. See the online supplementary materials for details.

We now present a simulation study to compare the performance of our application to sparse PCA with competing methods (Zou, Hastie, and Tibshirani 2006; Shen and Huang 2008; Song, Babu, and Palomar 2015). We simulate from $p$-variate normal distribution with mean zero and covariance matrix $\Sigma$. Let $d$ stand for the number of distinguishable PCs. We use $d = 3$ or 5. Let $\Sigma = U_d A_d U_d^T + I_d$, where $A_d$ is the diagonal matrix consisting of eigenvalues, satisfying $\text{diag}(A_d^{1/2}) = \{3, 5, \ldots, 5 - d + 1\}$. We use three models for the $p \times d$ eigenvector matrix $U_d$, defined below.

Model I: The eigenvector matrix $U_d$ has only $s = 10$ nonzero rows. Specifically, the first $s$ elements in each column of $U_d$ are sampled independently from $N_s(0, I_s)$. The eigenvectors under this model are coordinate-wise sparse.

Model II: For $s = 5$, the first $ds$ rows of $U_d$ are $s^{-1/2} I_d \otimes I_1$, and the rest of rows consist of zeros. Under this model, eigenvectors are not only orthogonal, but also combinations of disjoint sets of coordinates.

Model III: For $s = 5$, the eigenvector matrix $U_d$ has $ds$ nonzero rows, formed similar to a block-lower-triangular matrix. Specifically, the nonzero rows are given by the QR decomposition of $A_d \otimes I_s$, where $A_d$ is a $d \times d$ lower-triangular matrix, with all lower-triangular elements being one.

Model I calls for a coordinate-wise sparse estimation, Model II for an element-wise sparse estimation, while Model III does not clearly favor any choice of penalty. Both Models II and III seem unnatural to be conceived as a model underlying any real data. As argued in Bouveyron, Latouche, and Mattei (2016) coordinate-wise sparse PC directions are more natural to interpret than element-wise sparse directions.

The estimates from our methods are denoted by POI-L, POI-C, FastPOI-L, and FastPOI-C, where “L” stands for using the lasso penalty (9), and “C” stands for using the coordinate-wise group lasso penalty (10). For each method, we define a candidate

Figure 2. Sparse PCA by POI with coordinate-wise sparse penalty for a genomic dataset. The analysis is repeated for subspace dimension $d = 1$–20. Shown are estimated eigenvalues (left), nonzero coordinates shown as lighter color (middle), and $|\hat{q}_{i,i/d}|$ for $i = 1, 2, 3$, where $\hat{q}_{i,i/d}$ is the $i$th PC direction estimate when estimating PC subspace of dimension $d$ (right).
set of tuning parameters $L = \{(0.75)^i \lambda_{\text{max}} : i = 0, 1, \ldots, t, \infty\}$. We set $t = 31$, and $\lambda_{\text{max}}$ as the upper bound for $L$ as defined in Remarks 3 and 4. For each choice of $\lambda \in L$, we compute the cross-validation score (14), using an independent tuning set of data.

The performance of each method is measured by a distance between the estimated subspace, spanned by $\hat{U}_d$, and the true eigenspace $U_d$. The principal angles $\theta_i$ between two subspaces are often used to measure the distance. The principal angles between $\hat{U} \in O(p, d_1)$ and $U \in O(p, d_2)$ are defined as $\theta_i = \arccos(\sigma_i(\hat{U}^T U))$, where $\sigma_i(A)$ is the $i$th largest singular value of $A$. We use the projection metric

$$\rho(\hat{U}, U) = \max(\sin \theta_i : i = 1, \ldots, \min(d_1, d_2)).$$

If $d_1 = d_2 = d$, then the projection metric is equivalent to the difference between two corresponding projection matrices:

$$\rho(\hat{U}, U) = \|\hat{U} \hat{U}^T - UU^T\|,$$

where $\| \cdot \|$ is the natural spectral norm.

Let $\hat{U}(\lambda)$ be the estimate of $U$, when tuning parameter $\lambda$ is used. Denote the cross-validated estimate by $\hat{U}(\hat{\lambda})$ where $\hat{\lambda} = \arg \max_{\lambda \in L} CV(\lambda)$. The performances of subspace estimates are measured by the following criteria:

1. The minimal distance to the truth: $\min_{\lambda \in L} \rho(\hat{U}(\lambda), U)$.
2. The distance from the cross-validated estimate: $\rho(\hat{U}(\hat{\lambda}), U)$.

For each model we choose the number of PCs as $d = 3$ or 5, and choose the number of variables as $p = 200$ or 500. We use a small sample size of $n = 100$ for both training and tuning. The true number of PCs is treated as known. The empirical performances based on 100 repetitions of the experiments are summarized in Tables 1 and 2. Table 1 summarizes the potential of each method, while Table 2 shows the actual numerical performance with the automatic tuning parameter selection.

We note several observations from the simulation studies. First, POI solutions are potentially closer to the truth than Fast POI solutions. Second, as expected, our methods with coordinate-wise sparsity-inducing penalty (POI-C and FastPOI-C) work well for the coordinate-wise sparse models (I and II), especially for the larger subspace dimension $d = 5$. In contrast, the lasso-type penalty works well for Model II, the eigenvectors of which are strictly element-wise sparse. Finally, POI-C provides much more accurate estimates than the competing methods for coordinate-wise sparse models (Model I).

In terms of the variable selection performance of the estimates, POI-C and FastPOI-C are both clearly superior than any other methods for Model I, and are comparable to the best performing methods for Models II and III. Numerical results for the variable selection performance can be found in the online supplementary materials.

### 4.2. Sparse Subspace Learning for Multiclass LDA

We now apply our algorithm in the estimation of sparse discriminating basis for multigroup data. Suppose that $x$ follows a $K$-mixture of multivariate normal distributions, each with mean $\mu_j$ and a common variance $\Sigma = \Sigma_1$, for all $i = 1, \ldots, K$, where $K > 1$. Assuming that the group membership $y$ is also observed, write $\Sigma_T = \text{cov}(x)$ for the total-covariance matrix and $\Sigma_W = \Sigma$ for the within-group covariance matrix. The between-group covariance matrix is $\Sigma_B = \Sigma_T - \Sigma_W$, whose rank is at most $K - 1$. Then the problem of finding the discriminant subspace is equivalent to the GEP with $A = \Sigma_B$ and $B = \Sigma_W$.

We demonstrate the performance of fast POI, but not POI. There are three reasons for this. First of all, fast POI is well-suited for the case where the rank of $A$ is known. In multiclass LDA, the rank of $A$ is typically $K-1$. Second, the performance of fast POI was comparable or even superior to that of POI in our preliminary numerical studies. Third, POI requires considerably longer computation times than fast POI. The performance of fast POI estimates are compared with recently proposed linear sparse classifiers (Clemmensen et al. 2011; Mai, Yang, and Zou 2017; Gaynanova, Booth, and Wells 2016).

To simulate data, we use mixtures of normal distributions $N(\mu_i, \Sigma)$. We set the mean matrix $\mu = (\mu_1, \ldots, \mu_K)$ and $\Sigma$ as follows. Throughout, the dimension is set to be $p = 200$. Let $C(\rho)$ be the compound symmetry covariance model: $C(\rho) = (1 - \rho)I_p + \rho I_p$. Let $R(\rho)$ be the AR(1) model: $[R(\rho)]_{ij} = \rho^{|i-j|}$. Define $V = (v_1, v_2, v_3)$ and $W = (w_1, w_2, w_3)$ by

$$v_1 = (2, 1, 2, 1, 2, 0, \ldots, 0)_{1 \times p},$$

$$w_1 = (-1, 1, 1, 1, 0, \ldots, 0)_{1 \times p},$$

$$v_2 = (1, -1, 1, -1, 1, 0, \ldots, 0)_{1 \times p}.$$
Our models are given as follows. For Model I, \( \mathbf{\mu} = (\mu_1, \mu_2, \mu_3) \) = \( \mathbf{V}, \mathbf{\Sigma} = \mathbf{I}_p \); Model II, \( \mathbf{\mu} = \mathbf{\Sigma} \mathbf{V}, \mathbf{\Sigma} = C(0.5) \); Model III, \( \mathbf{\mu} = \mathbf{\Sigma} \mathbf{V}, \mathbf{\Sigma} = R(0.5) \); Model IV, \( \mathbf{\mu} = \mathbf{\Sigma} \mathbf{W}, \mathbf{\Sigma} = C(0.5) \); for Model V, let \( \mathbf{W} = 2(\mathbf{w}_1, \mathbf{w}_2, \mathbf{w}_3, \mathbf{w}) \), where \( \mathbf{w} = (\mathbf{w}_1 + \mathbf{w}_2 + \mathbf{w}_3)/3 \), and set \( \mathbf{\mu} = (\mu_1, \mu_2, \mu_3, \mu_4) = \mathbf{\Sigma} \mathbf{W}, \mathbf{\Sigma} = C(0.5) \).

The true generalized eigenvectors are given by solving the generalized eigenvector problem with \( \mathbf{A} = \sum_{k=1}^{K} (\mathbf{\mu}_k - \bar{\mathbf{\mu}})(\mathbf{\mu}_k - \bar{\mathbf{\mu}})^T/K \), where \( \bar{\mathbf{\mu}} = \sum_{k=1}^{K} \mu_k/K \) and \( \mathbf{B}_k = \mathbf{\Sigma} \). Note that although the eigenvectors are different for different models, the true subspaces in Models I–III are the same; they are all spanned by \( \{v_i, v_j, i \neq j\} \). Likewise, the true subspaces in Models IV and V are both spanned by \( \{w_i - w_j, i \neq j\} \). Since the basis of a subspace can only be coordinate-wise sparse, it is expected that coordinate-wise sparse estimates (including FastPOI-C and the method of Gaynanova, Booth, and Wells (2016)) work well. Our simulation result, reported below, concurs.

In Models I–IV, we need to estimate a subspace of dimension 2, since there are \( K = 3 \) groups. In Model V, we still need to estimate the two-dimensional subspace, even though there are \( K = 4 \) groups. In principle, this information is not available to statisticians, and for all model estimation, we estimate the subspace of dimension \( d = K - 1 \); that is, the model is misspecified for Model V. Our simulation shows that our method works well for this case as well.

Our application to sparse LDA as well as the competing methods (Clemmensen et al. 2011; Mai, Yang, and Zou 2017; Gaynanova, Booth, and Wells 2016) first estimate sparse basis vectors, then apply multiclass LDA to the dimension-reduced data. Following their approaches, the standard LDA is applied to the data pair \((\mathbf{X}, \gamma)\) for the classification and prediction. From each model above, we generate a training set of size \( n_1 = 30 \) (for each group), a tuning set of the same size, and a testing set of size \( n = 100K_n \). Our tuning procedure, using (14), is used for all methods. The testing set is used to estimate the misclassification rate. Experiments are repeated 100 times. We report the qualities of subspace learning and classification performance in Tables 3 and 4, respectively.

Table 3 shows that FastPOI-C provides most accurate estimates of the true subspace. As explained above, it makes most sense to assume coordinate-wise sparsity in subspace learning. Thus, the methods of Mai, Yang, and Zou (2017) and Clemmensen et al. (2011), seeking element-wise sparsity, are expected to be inferior in subspace learning. Likewise, the performance of FastPOI-L is inferior to that of FastPOI-C. Note that Gaynanova, Booth, and Wells (2016) also imposed the coordinate-wise sparsity, thus showing better performance in subspace learning than other method, except FastPOI-C.

The proposed FastPOI-C also performed the best in terms of classification. It is evident from Table 4 that FastPOI-C yields the smallest misclassification rates for Models I–IV, and is second to Gaynanova’s classifier for Model V. Given the similarity of FastPOI-C and Gaynanova’s (note Remark 2 and that they both seek coordinate-wise sparse solutions), it is expected that both perform well in classification.

An alternative tuning procedure, minimizing misclassification error rates in the special case of multiclass LDA, provides similar numerical performances. In fact, both choices of the tuning parameter are generally close to each other, as we have seen in Figure 1. Fast POI tends to choose more variables than needed, but shows better sensitivity, that is, more signal vari-

### Table 2. The projection distance from the cross-validated estimate, averaged from 100 repetitions, for principal subspace estimation.

| Model | \( d \) | \( p \) | POI-L | POI-C | FastPOI-L | FastPOI-C | Zou et al. | Song et al. | Shen & Huang |
|-------|------|------|------|------|----------|----------|-----------|------------|-------------|
| I     | 3    | 200  | 0.202| **0.162** | 0.204  | **0.165** | 0.201  | 0.262      | 0.198       |
| II    | 3    | 500  | 0.204| **0.152** | 0.213  | 0.159   | 0.211  | 0.292      | 0.205       |
| III   | 5    | 200  | 0.359| **0.199** | 0.420  | 0.228   | 0.419  | 0.445      | 0.369       |
| IV    | 5    | 500  | 0.482| **0.209** | 0.651  | 0.378   | 0.443  | 0.576      | 0.397       |
| V     | 2    | 300  | 0.111| **0.111** | 0.163  | 0.159   | 0.162  | 0.130      | 0.107       |
|       | 2    | 500  | **0.110** | 0.166  | 0.158  | 0.166  | 0.127  | **0.105**  | **0.106**   |
|       | 5    | 200  | 0.284| 0.538  | 0.354  | 0.459   | 0.293  | 0.276      | 0.232       |
|       | 5    | 500  | 0.376| 0.620  | 0.630  | 0.705   | 0.331  | 0.361      | **0.286**   |

**NOTE:** The standard errors are at most 0.027. Smaller distance indicates more precise estimation. Highlighted are the best performed models (within 2 standard error of the smallest).

### Table 3. The projection distance from the estimate, averaged from 100 repetitions, for sparse discriminant basis learning.

| Model | FastPOI-L | FastPOI-C | Mai et al. | Clemmensen et al. | Gaynanova et al. |
|-------|-----------|-----------|------------|-------------------|------------------|
| I     | **0.328** | **0.313** | 0.371      | 0.622             | 0.377            |
| II    | 0.839     | **0.570** | 0.936      | 0.817             | 0.611            |
| III   | 0.644     | **0.437** | 0.542      | 0.784             | 0.608            |
| IV    | 0.852     | **0.478** | 0.916      | 0.689             | 0.514            |
| V     | 0.712     | **0.359** | 0.869      | **0.365**         | 0.411            |

**NOTE:** The standard errors are at most 0.024. Smaller distance indicates more precise estimation. Highlighted are the best performed models (within 2 standard error of the smallest).
ables (or coordinates) are included in the estimates, than other methods. In terms of an overall variable selection performance, FastPOI-C and Gaynanova, Booth, and Wells’s (2016) method shows similar performances, while both are better than the others. Simulation results for the alternative choices of tuning parameters and for the variable selection performance can be found in the online supplementary materials.

4.3. Sufficient Dimension Reduction

Sufficient dimension reduction (Cook 2009) aims to find a projection of data \( x \) that is sufficient for (i.e., preserves all information about) the conditional distribution of \( y \) given \( x \). Many SDR methods can be cast into a GEP, as shown in Li (2007). In particular, the solution for the sliced inverse regression (SIR, Li 1991), the most well-known method for SDR, is given by the eigenvectors of the GEP (1) where \( A = \text{cov}[E(x−E(x)|y)] \) and \( B = \text{cov}(x) \). We apply our sparse solutions of GEP by POI-C to SIR for a benchmark dataset. In all examples in this subsection, we used \( \lambda = \lambda_{\text{max}}/2 \).

The Tai-Chi data have been sometimes used as a benchmark data for various methods of SDR, for example, in Wu et al. (2009). For our purpose, we randomly generate \((x,y)\) pairs for \( n = 1000 \) times as shown in Figure 3. The first two coordinates of \( x \in \mathbb{R}^p \) are uniformly sampled over the disk with radius 2, while the rest of coordinates are sampled from the standard normal distribution. The binary variable \( y \) depends on the location of \( x \), and is 1 if \( x \) lies on the “yin” part of the Tai-Chi symbol, 0 otherwise. The true subspace only spans over the first two coordinates.

For SDR of the toy data, a direct competitor is Chen, Zou, and Cook’s (2010) sparse estimation for SIR. Note that Tan et al.’s (2016) method is not applicable to this dataset as it only computes one eigenvector, while at least two eigenvectors are needed here. The method of Li (2007) is not compared, as Chen, Zou, and Cook (2010) improves upon Li (2007). Figure 3 displays the projected data onto the estimated subspaces from the original SIR and our adaptation of SIR with POI-C. The graphical result from Chen, Zou, and Cook (2010) is almost identical to POI-C, and both methods greatly improve upon the original SIR for this dataset.

To further compare our approach with Chen, Zou, and Cook’s (2010), we have repeated the experiment for various cases of sample size and dimension. It appears that Chen’s method requires large sample size to work well, and is numerically unstable for small sample sizes. Our method exhibits great performances for any case and requires only a fraction of computation time compared to Chen’s. Relevant numerical results can be found in the online supplementary materials.

4.4. Canonical Correlation Analysis

We briefly demonstrate an application of our methods to sparse CCA. It is well known that CCA can be viewed as a GEP (Gaynanova, Booth, and Wells 2017; Safo et al. 2018). The canonical coefficient vectors are defined as follows. For two random vectors \( x \in \mathbb{R}^p, y \in \mathbb{R}^q \), write \( \Sigma_1 = \text{cov}(x), \Sigma_2 = \text{cov}(y), \) and \( \Sigma_{12} = \text{cov}(x,y) = \Sigma_{21} \). The coefficient vectors for the first pair of canonical variables are \((g_1,h_1) \in \mathbb{R}^p \times \mathbb{R}^q\), maximizing the correlation between \( g^T x \) and \( h^T y \): \( \rho(g,h) = g^T \Sigma_{12} h / (g^T \Sigma_{11} g)^{1/2} (h^T \Sigma_{22} h)^{1/2} \). Since \( \rho(g,h) \) is invariant under individual scaling of \((g,h), \) a Lagrangian formulation of the maximization involves the condition \( g^T \Sigma_{11} g = h^T \Sigma_{22} h = 1, \) and the first-order condition of the Lagrangian coincides with the GEP (1) for

\[
A = \begin{pmatrix} 0 & \Sigma_{12} \\ \Sigma_{21} & 0 \end{pmatrix}, \quad B = \begin{pmatrix} \Sigma_1 & 0 \\ 0 & \Sigma_2 \end{pmatrix},
\]

where the solution \((u,\lambda)\) corresponds to the concatenated coefficient vector \((g^T,h^T)^T\) and canonical correlation coefficient \( \rho(g,h) \), respectively. An alternative formulation of CCA is given by solving the GEP (23) with respect to individual \( g \) or \( h \), leading

![Figure 3](image-url). Projections of a Tai-Chi dataset to the true subspace (left panel), the estimated subspace by the original SIR (Li 1991, middle panel), SIR estimated by POI-C (right panel). The dataset has \( n = 1000 \) observations, each with dimension \( p = 500 \).
to two GEPs:
\[ \Sigma_{12} \Sigma_{21}^{-1} \Sigma_{21} g = \lambda \Sigma_{11} g, \quad \Sigma_{21} \Sigma_{11}^{-1} \Sigma_{11} h = \lambda \Sigma_{22} h. \]  
(24)

In performing a sparse CCA, we use (24) as done in Safo et al. (2018). In the estimation of the pair \((g, h)\) from a sample, we follow the suggestion from Witten, Tibshirani, and Hastie (2009) and Safo et al. (2018) of first standardizing the data, then replacing \(\hat{\Sigma}_1\) by \(I_p\) and \(\hat{\Sigma}_2\) by \(I_q\). With the sample cross-covariance matrix \(\hat{\Sigma}_{12}\), we apply the POI in solving \(\Sigma_{12} \Sigma_{12}^T g = \lambda g\) and \(\Sigma_{21} \Sigma_{21}^T h = \lambda h\). Applying our method, or any existing general sparse GEP methods (Srimerumbudur, Torres, and Lanckriet 2011; Song, Babu, and Palomar 2015), to the GEP (23) typically leads unsatisfactory results partly due to the fact that \(\Lambda\) is not in general nonnegative definite and also because of different scales of \(\Sigma_1\) and \(\Sigma_2\). In a simulation study with \(n = 80\) observations of two random vectors of dimension \(p = 200, q = 150\), in which there is one canonical pair, the performance of POI estimates is comparable to the method of Safo et al. (2018), and are superior than the methods of Witten, Tibshirani, and Hastie (2009) and Gao, Ma, and Zhou (2017). We refer to the online supplementary materials for simulation settings and numerical results.

Appendix: Technical Details

Proof of Proposition 1. There exists a nonsingular matrix \(S\) such that \(U_d = Q_d S\). Equation (4) is written as
\[ A Q_d S = B Q_d S A_d, \]
which yields
\[ (Q_d^T A Q_d) S = (Q_d^T B Q_d) S A_d, \]
moreover, we have \(S' (Q_d^T B Q_d) S = U_d' B U_d = I_d\). Therefore, \(S\) and \(A_d\), respectively, are the matrix of eigenvectors and the diagonal matrix of eigenvalues of the GEP problem (6).

The following elementary lemma is used in the proof of Proposition 2. We use the notation \(C(A)\) to denote the column space of matrix \(A\).

Lemma 4. Let \(M_1, M_2 \in \mathbb{R}^{p \times d}\), \(N \in \mathbb{R}^{d \times d}\), \(L \in \mathbb{R}^{p \times p}\) and assume that both \(N, L\) are invertible.
1. \(C(M) = C(NM)\), then \(C(LM) = C(LM_1)\).
2. \(C(M_1) = C(M_2)\), then \(C(LM_1) = C(LM_2)\).

We omit the proof of Lemma 4.

Proof of Proposition 2. Under Case (a), \(B = I_p\), and the generalized eigen-equation becomes \(A Q_d = Q_d^* A^*\). Thus, \(Q_d = V_d\) by definition, which in turn leads to \(C(B^{-1} V_d) = C(V_d) = C(Q_d)\).

For case (b), define \(C = B^{-\frac{1}{2}} A^{-\frac{1}{2}}\). Then the generalized eigen-equation rewrites to \(C B^2 Q_d = B^2 Q_d A^*_d\). By the QR decomposition of \(B^2 Q_d\), there exist \(Q_d^* \in \mathcal{O}(p, d)\) and a \(k \times k\) upper-triangular matrix \(A^*_d\) satisfying
\[ C Q_d^* = Q_d^* A^*_d \]
(A.1)
Since rank\((C) = k\), Equation (A.1) is equivalent to
\[ C[Q_d^*, Q_d^+] = [Q_d^+, Q_d^+] \begin{pmatrix} A^*_d & 0 \\ 0 & 0 \end{pmatrix}. \]
(A.2)

where \([Q_d^+, Q_d^+] \in \mathcal{O}(p)\). Thus, \(C = Q_d^+ (Q_d^+)^T\), and we get \(C(C) = C(Q_d^+) = C(B^3 Q_d)\). (These are obtained by the definition of eigendecomposition and QR decomposition.) On the other hand, using Lemma 4, it can be shown that \(C(B^2 B^{-1} V_d) = C(B^{-1} V_d) = C(B^2 B^{-1} V_d) = C(C)\).

Since \(B^2\) is invertible and \(C(B^2 B^{-1} V_d) = C(B^2 Q_d)\), again by Lemma 4, we conclude that \(C(B^{-1} V_d) = C(Q_d)\).

Proof of Lemma 3. Notice that each optimization problem is column-wise separable. Each column-wise subproblem is then a least-square problem.

Supplementary Materials

Additional analysis: Document containing an extensive list of statistical GEP problems and additional numerical results (.pdf file).

Matlab routine: Matlab functions that perform the sparse generalized eigenvector estimation as described in the article. (Compressed files .zip ) The files are also available at https://github.com/sungkyujiang/SparseEIG.

Funding

Jeon’s research was supported by Basic Science Research Program of the National Research Foundation of Korea (NRF-2015R1A1A1A05001180) funded by the Korean government.

References

Bouveyron, C., Latouche, P., and Mattei, P.-A. (2016), “Bayesian Variable Selection for Globally Sparse Probabilistic PCA,” arXiv no. 1605.05918. [712,716]
Caliński, T., and Harabasz, J. (1974), “A Dendrite Method for Cluster Analysis,” Communications in Statistics—Theory and Methods, 3, 1–27. [714]
Chen, X., Zou, C., and Cook, R. D. (2010), “Coordinate-Independent Sparse Sufficient Dimension Reduction and Variable Selection,” The Annals of Statistics, 38, 3696–3723. [711,719]
Ciriello, G., Gatza, M. L., Beck, A. H., Wilkerson, M. D., Rhee, S. K., Pastore, A., Zhang, H., McLellan, M., Yau, C., Kandoth, C., and Bowbyr, R. (2015), “Comprehensive Molecular Portraits of Invasive Lobular Breast Cancer,” Cell, 163, 506–519. [716]
Clemmensen, L., Hastie, T., Witten, D., and Ersbøll, B. (2011), “Sparse Discriminant Analysis,” Technometrics, 53, 406–413. [717,718]
Cook, R. D. (2009), Regression Graphics: Ideas for Studying Regressions Through Graphics (Vol. 482), New York: Wiley. [719]
Gao, C., Ma, Z., and Zhou, H. H. (2017), “Sparse CCA: Adaptive Estimation and Computational Barriers,” The Annals of Statistics, 45, 2074–2101. [720]
Gaynanova, I., Booth, J. G., and Wells, M. T. (2016), “Simultaneous Sparse Estimation of Canonical Vectors in the \(p \gg n\) Setting,” Journal of the American Statistical Association, 111, 696–706. [713,717,718,719]
(2017), “ Penalized Versus Constrained Generalized Eigenvalue Problems,” Journal of Computational and Graphical Statistics, 26, 379–387. [711,719]
Golub, G. H., and Van Loan, C. F. (1996), Matrix Computations (3rd ed.), Baltimore, MD: JHU Press. [710,712]
Han, X., and Clemmensen, L. (2016), “Regularized Generalized Eigen-decomposition With Applications to Sparse Supervised Feature Extraction and Sparse Discriminant Analysis,” Pattern Recognition, 49, 43–54. [711]
Hastie, T., Tibshirani, R., and Friedman, J. (2009), Elements of Statistical Learning, New York: Springer-Verlag. [710]
Johnstone, I. M., and Lu, A. Y. (2009), “On Consistency and Sparsity for Principal Components Analysis in High Dimensions,” Journal of the American Statistical Association, 104, 682–693. [710]
Josse, J., and Husson, F. (2012), "Selecting the Number of Components in Principal Component Analysis Using Cross-Validation Approximations," *Computational Statistics & Data Analysis*, 56, 1869–1879. [714]

Jung, S., and Marron, J. S. (2009), "PCA Consistency in High Dimension, Low Sample Size Context," *The Annals of Statistics*, 37, 4104–4130. [710]

Kokiopoulou, E., Chen, J., and Saad, Y. (2011), "Trace Optimization and Eigenproblems in Dimension Reduction Methods," *Numerical Linear Algebra With Applications*, 18, 565–602. [710]

Li, K.-C. (1991), "Sliced Inverse Regression for Dimension Reduction (with discussion)," *Journal of the American Statistical Association*, 86, 316–342. [719]

Li, L. (2007), "Sparse Sufficient Dimension Reduction," *Biometrika*, 94, 603–613. [710,719]

Ma, Z. (2013), "Sparse Principal Component Analysis and Iterative Thresholding," *The Annals of Statistics*, 41, 772–801. [712]

Mai, Q., Yang, Y., and Zou, H. (2017), "Multiclass Sparse Discriminant Analysis," arXiv no. 1504.05845. [713,717,718]

Safo, S. E., Ahn, J., Jeon, Y., and Jung, S. (2018), "Sparse Generalized Eigenvalue Problem With Application to Canonical Correlation Analysis for Integrative Analysis of Methylation and Gene Expression Data," *Biometrics*, 74, 1362–1371. [711,719,720]

Shen, H., and Huang, J. Z. (2008), "Sparse Principal Component Analysis via Regularized Low Rank Matrix Approximation," *Journal of Multivariate Analysis*, 99, 1015–1034. [714,716]

Song, J., Babu, P., and Palomar, D. P. (2015), "Sparse Generalized Eigenvalue Problem via Smooth Optimization," *IEEE Transactions on Signal Processing*, 63, 1627–1642. [711,716,720]

Sriperumbudur, B. K., Torres, D. A., and Lanckriet, G. R. (2011), "A Majorization-Minimization Approach to the Sparse Generalized Eigenvalue Problem," *Machine Learning*, 85, 3–39. [711,720]

Tan, K. M., Wang, Z., Liu, H., and Zhang, T. (2016), "Sparse Generalized Eigenvalue Problem: Optimal Statistical Rates via Truncated Rayleigh Flow," arXiv no. 1604.08697. [711,719]

Tibshirani, R. (1996), "Regression Shrinkage and Selection via the Lasso," *Journal of the Royal Statistical Society*, Series B, 58, 267–288. [712]

——— (2011), "Regression Shrinkage and Selection via the Lasso: A Retrospective," *Journal of the Royal Statistical Society*, Series B, 73, 273–282. [712]

Tseng, P. (1993), "Dual Coordinate Ascent Methods for Non-strictly Convex Minimization," *Mathematical Programming*, 59, 231–247. [714]

Tyler, D. E., Critchley, F., Dümbgen, L., and Oja, H. (2009), "Invariant Coordinate Selection," *Journal of the Royal Statistical Society*, Series B, 71, 549–592. [710]

Witten, D. M., Tibshirani, R., and Hastie, T. (2009), "A Penalized Matrix Decomposition, With Applications to Sparse Principal Components and Canonical Correlation Analysis," *Biostatistics*, 10, 515–534. [720]

Wu, M., Zhang, L., Wang, Z., Christiani, D., and Lin, X. (2009), "Sparse Linear Discriminant Analysis for Simultaneous Testing for the Significance of a Gene Set/Pathway and Gene Selection," *Bioinformatics*, 25, 1145–1151. [719]

Yuan, M., and Lin, Y. (2006), "Model Selection and Estimation in Regression With Grouped Variables," *Journal of the Royal Statistical Society*, Series B, 68, 49–67. [712]

Zhang, Z., Chow, T. W., and Zhao, M. (2013), "Trace Ratio Optimization-Based Semi-supervised Nonlinear Dimensionality Reduction for Marginal Manifold Visualization," *IEEE Transactions on Knowledge and Data Engineering*, 25, 1148–1161. [710]

Zou, H., Hastie, T., and Tibshirani, R. (2006), "Sparse Principal Component Analysis," *Journal of Computational and Graphical Statistics*, 15, 265–286. [716]