Optimal Solutions for Sparse Principal Component Analysis

Alexandre d'Aspremont ASPREMON@PRINCETON.EDU
ORFE, Princeton University,
Princeton, NJ 08544, USA.

Francis Bach FRANCIS.BACH@MINES.ORG
INRIA - Willow project
Département d'Informatique, Ecole Normale Supérieure
45, rue d'Ulm, 75230 Paris, France

Laurent El Ghaoui ELGHAOUI@EECS.BERKELEY.EDU
EECS Department, U.C. Berkeley,
Berkeley, CA 94720, USA.

Editor:

Abstract

Given a sample covariance matrix, we examine the problem of maximizing the variance explained by a linear combination of the input variables while constraining the number of nonzero coefficients in this combination. This is known as sparse principal component analysis and has a wide array of applications in machine learning and engineering. We formulate a new semidefinite relaxation to this problem and derive a greedy algorithm that computes a *full set* of good solutions for all target numbers of non zero coefficients, with total complexity $O(n^3)$, where n is the number of variables. We then use the same relaxation to derive sufficient conditions for global optimality of a solution, which can be tested in $O(n^3)$ per pattern. We discuss applications in subset selection and sparse recovery and show on artificial examples and biological data that our algorithm does provide globally optimal solutions in many cases.

Keywords: PCA, subset selection, sparse eigenvalues, sparse recovery, lasso.

1. Introduction

Principal component analysis (PCA) is a classic tool for data analysis, visualization or compression and has a wide range of applications throughout science and engineering. Starting from a multivariate data set, PCA finds linear combinations of the variables called *principal components*, corresponding to orthogonal directions maximizing variance in the data. Numerically, a full PCA involves a singular value decomposition of the data matrix.

One of the key shortcomings of PCA is that the factors are linear combinations of *all* original variables; that is, most of factor coefficients (or loadings) are non-zero. This means that while PCA facilitates model interpretation and visualization by concentrating the information in a few factors, the factors themselves are still constructed using all variables, hence are often hard to interpret.

In many applications, the coordinate axes involved in the factors have a direct physical interpretation. In financial or biological applications, each axis might correspond to a specific asset or gene. In problems such as these, it is natural to seek a trade-off between the two goals of *statistical fidelity* (explaining most of the variance in the data) and *interpretability* (making sure that the factors involve only a few coordinate axes). Solutions that have only a few nonzero coefficients in the principal components are usually easier to interpret. Moreover, in some applications, nonzero coefficients have a direct cost (*e.g.*, transaction costs in finance) hence there may be a direct trade-off between statistical fidelity and practicality. Our aim here is to efficiently derive *sparse principal components*, i.e, a set of sparse vectors that explain a maximum amount of variance. Our belief is that in many applications, the decrease in statistical fidelity required to obtain sparse factors is small and relatively benign.

In what follows, we will focus on the problem of finding sparse factors which explain a maximum amount of variance, which can be written:

$$\max_{\|z\|\leq 1} z^T \Sigma z - \rho\,\mathbf{Card}(z) \tag{1}$$

in the variable $z \in \mathbf{R}^n$, where $\Sigma \in \mathbf{S}_n$ is the (symmetric positive semi-definite) sample covariance matrix, ρ is a parameter controlling sparsity, and $\mathbf{Card}(z)$ denotes the cardinal (or ℓ_0 norm) of z, i.e. the number of non zero coefficients of z.

While PCA is numerically easy, each factor requires computing a leading eigenvector, which can be done in $O(n^2)$, sparse PCA is a hard combinatorial problem. In fact, Moghaddam et al. (2006b) show that the subset selection problem for ordinary least squares, which is NP-hard (Natarajan, 1995), can be reduced to a sparse generalized eigenvalue problem, of which sparse PCA is a particular intance. Sometimes ad hoc "rotation" techniques are used to post-process the results from PCA and find interpretable directions underlying a particular subspace (see Jolliffe (1995)). Another simple solution is to *threshold* the loadings with small absolute value to zero (Cadima and Jolliffe, 1995). A more systematic approach to the problem arose in recent years, with various researchers proposing nonconvex algorithms (e.g., SCoTLASS by Jolliffe et al. (2003), SLRA by Zhang et al. (2002) or D.C. based methods (Sriperumbudur et al., 2007) which find modified principal components with zero loadings. The SPCA algorithm, which is based on the representation of PCA as a regression-type optimization problem (Zou et al., 2006), allows the application of the LASSO (Tibshirani, 1996), a penalization technique based on the ℓ_1 norm. With the exception of simple thresholding, all the algorithms above require solving non convex problems. Recently also, d'Aspremont et al. (2007b) derived an ℓ_1 based semidefinite relaxation for the sparse PCA problem (1) with a complexity of $O(n^4\sqrt{\log n})$ for a given ρ. Finally, Moghaddam et al. (2006a) used greedy search and branch-and-bound methods to solve small instances of problem (1) exactly and get good solutions for larger ones. Each step of this greedy algorithm has complexity $O(n^3)$, leading to a total complexity of $O(n^4)$ for a full set of solutions.

Our contribution here is twofold. We first derive a greedy algorithm for computing a *full set* of good solutions (one for each target sparsity between 1 and n) at a total numerical cost of $O(n^3)$ based on the convexity of the of the largest eigenvalue of a symmetric matrix. We then derive *tractable* sufficient conditions for a vector z to be a *global* optimum of (1).

This means in practice that, given a vector z with support I, we can test if z is a globally optimal solution to problem (1) by performing a few binary search iterations to solve a one dimensional convex minimization problem. In fact, we can take any sparsity pattern candidate from any algorithm and test its optimality. This paper builds on the earlier conference version (d'Aspremont et al., 2007a), providing new and simpler conditions for optimality and describing applications to subset selection and sparse recovery.

While there is certainly a case to be made for ℓ_1 penalized maximum eigenvalues (à la d'Aspremont et al. (2007b)), we strictly focus here on the ℓ_0 formulation. However, it was shown recently (see Candès and Tao (2005), Donoho and Tanner (2005) or Meinshausen and Yu (2006) among others) that there is in fact a deep connection between ℓ_0 constrained extremal eigenvalues and LASSO type variable selection algorithms. Sufficient conditions based on sparse eigenvalues (also called restricted isometry constants in Candès and Tao (2005)) guarantee consistent variable selection (in the LASSO case) or sparse recovery (in the decoding problem). The results we derive here produce upper bounds on sparse extremal eigenvalues and can thus be used to prove consistency in LASSO estimation, prove perfect recovery in sparse recovery problems, or prove that a particular solution of the subset selection problem is optimal. Of course, our conditions are only sufficient, not necessary (which would contradict the NP-Hardness of subset selection) and the duality bounds we produce on sparse extremal eigenvalues cannot always be tight, but we observe that the duality gap is often small.

The paper is organized as follows. We begin by formulating the sparse PCA problem in Section 2. In Section 3, we write an efficient algorithm for computing a full set of candidate solutions to problem (1) with total complexity $O(n^3)$. In Section 4 we then formulate a convex relaxation for the sparse PCA problem, which we use in Section 5 to derive tractable sufficient conditions for the global optimality of a particular sparsity pattern. In Section 6 we detail applications to subset selection, sparse recovery and variable selection. Finally, in Section 7, we test the numerical performance of these results.

Notation

For a vector $z \in \mathbf{R}$, we let $\|z\|_1 = \sum_{i=1}^n |z_i|$ and $\|z\| = \left(\sum_{i=1}^n z_i^2\right)^{1/2}$, $\mathbf{Card}(z)$ is the cardinality of z, i.e. the number of nonzero coefficients of z, while the support I of z is the set $\{i : \; z_i \neq 0\}$ and we use I^c to denote its complement. For $\beta \in \mathbf{R}$, we write $\beta_+ = \max\{\beta, 0\}$ and for $X \in \mathbf{S}_n$ (the set of symmetric matrix of size $n \times n$) with eigenvalues λ_i, $\mathbf{Tr}(X)_+ = \sum_{i=1}^n \max\{\lambda_i, 0\}$. The vector of all ones is written $\mathbf{1}$, while the identity matrix is written $\mathbf{I}$. The diagonal matrix with the vector u on the diagonal is written $\mathbf{diag}(u)$.

2. Sparse PCA

Let $\Sigma \in \mathbf{S}_n$ be a symmetric matrix. We consider the following sparse PCA problem:

$$\phi(\rho) \equiv \max_{\|z\| \leq 1} z^T \Sigma z - \rho \, \mathbf{Card}(z) \tag{2}$$

in the variable $z \in \mathbf{R}^n$ where $\rho > 0$ is a parameter controlling sparsity. We assume without loss of generality that $\Sigma \in \mathbf{S}_n$ is positive semidefinite and that the n variables are ordered by decreasing marginal variances, i.e. that $\Sigma_{11} \geq \ldots \geq \Sigma_{nn}$. We also assume that we are given a square root A of the matrix Σ with $\Sigma = A^T A$, where $A \in \mathbf{R}^{n\times n}$ and we denote by $a_1,\ldots,a_n \in \mathbf{R}^n$ the columns of A. Note that the problem and our algorithms are invariant by permutations of Σ and by the choice of square root A. In practice, we are very often given the data matrix A instead of the covariance Σ.

A problem that is directly related to (2) is that of computing a cardinality constrained maximum eigenvalue, by solving:

$$\begin{array}{ll} \text{maximize} & z^T\Sigma z \\ \text{subject to} & \mathbf{Card}(z) \leq k \\ & \|z\| = 1, \end{array} \tag{3}$$

in the variable $z \in \mathbf{R}^n$. Of course, this problem and (2) are related. By duality, an upper bound on the optimal value of (3) is given by:

$$\inf_{\rho\in P} \phi(\rho) + \rho k.$$

where P is the set of penalty values for which $\phi(\rho)$ has been computed. This means in particular that if a point z is provably optimal for (2), it is also globally optimum for (3) with $k = \mathbf{Card}(z)$.

We now begin by reformulating (2) as a relatively simple convex maximization problem. Suppose that $\rho \geq \Sigma_{11}$. Since $z^T\Sigma z \leq \Sigma_{11}(\sum_{i=1}^n |z_i|)^2$ and $(\sum_{i=1}^n |z_i|)^2 \leq \|z\|^2 \mathbf{Card}(z)$ for all $z \in \mathbf{R}^n$, we always have:

$$\begin{array}{rl} \phi(\rho) & = \max_{\|z\|\leq 1} z^T\Sigma z - \rho\,\mathbf{Card}(z) \\ & \leq (\Sigma_{11} - \rho)\,\mathbf{Card}(z) \\ & \leq 0, \end{array}$$

hence the optimal solution to (2) when $\rho \geq \Sigma_{11}$ is $z = 0$. From now on, we assume $\rho \leq \Sigma_{11}$ in which case the inequality $\|z\| \leq 1$ is tight. We can represent the sparsity pattern of a vector z by a vector $u \in \{0,1\}^n$ and rewrite (2) in the equivalent form:

$$\begin{array}{rcl} \phi(\rho) & = & \displaystyle\max_{u\in\{0,1\}^n} \lambda_{\max}(\mathbf{diag}(u)\Sigma\,\mathbf{diag}(u)) - \rho\mathbf{1}^T u \\ & = & \displaystyle\max_{u\in\{0,1\}^n} \lambda_{\max}(\mathbf{diag}(u)A^TA\,\mathbf{diag}(u)) - \rho\mathbf{1}^T u \\ & = & \displaystyle\max_{u\in\{0,1\}^n} \lambda_{\max}(A\,\mathbf{diag}(u)A^T) - \rho\mathbf{1}^T u, \end{array}$$

using the fact that $\mathbf{diag}(u)^2 = \mathbf{diag}(u)$ for all variables $u \in \{0,1\}^n$ and that for any matrix B, $\lambda_{\max}(B^TB) = \lambda_{\max}(BB^T)$. We then have:

$$\begin{array}{rcl} \phi(\rho) & = & \displaystyle\max_{u\in\{0,1\}^n} \lambda_{\max}(A\,\mathbf{diag}(u)A^T) - \rho\mathbf{1}^T u \\ & = & \displaystyle\max_{\|x\|=1}\ \max_{u\in\{0,1\}^n} x^TA\,\mathbf{diag}(u)A^Tx - \rho\mathbf{1}^T u \\ & = & \displaystyle\max_{\|x\|=1}\ \max_{u\in\{0,1\}^n} \sum_{i=1}^n u_i((a_i^Tx)^2 - \rho). \end{array}$$

Hence we finally get, after maximizing in u (and using $\max_{v\in\{0,1\}} \beta v = \beta_+$):

$$\phi(\rho) = \max_{\|x\|=1} \sum_{i=1}^n ((a_i^Tx)^2 - \rho)_+, \tag{4}$$

which is a nonconvex problem in the variable $x \in \mathbf{R}^n$. We then select variables i such that $(a_i^T x)^2 - \rho > 0$. Note that if $\Sigma_{ii} = a_i^T a_i < \rho$, we must have $(a_i^T x)^2 \leq \|a_i\|^2 \|x\|^2 < \rho$ hence variable i will never be part of the optimal subset and we can remove it.

3. Greedy Solutions

In this section, we focus on finding a good solution to problem (2) using greedy methods. We first present very simple preprocessing solutions with complexity $O(n \log n)$ and $O(n^2)$. We then recall a simple greedy algorithm with complexity $O(n^4)$. Finally, our first contribution in this section is to derive an approximate greedy algorithm that computes a full set of (approximate) solutions for problem (2), with total complexity $O(n^3)$.

3.1 Sorting and Thresholding

The simplest ranking algorithm is to sort the diagonal of the matrix Σ and rank the variables by variance. This works intuitively because the diagonal is a rough proxy for the eigenvalues: the Schur-Horn theorem states that the diagonal of a matrix majorizes its eigenvalues (Horn and Johnson, 1985); sorting costs $O(n \log n)$. Another quick solution is to compute the leading eigenvector of Σ and form a sparse vector by thresholding to zero the coefficients whose magnitude is smaller than a certain level. This can be done with cost $O(n^2)$.

3.2 Full greedy solution

Following Moghaddam et al. (2006a), starting from an initial solution of cardinality one at $\rho = \Sigma_{11}$, we can update an increasing sequence of index sets $I_k \subseteq [1, n]$, scanning all the remaining variables to find the index with maximum variance contribution.

Greedy Search Algorithm.

- **Input**: $\Sigma \in \mathbf{R}^{n \times n}$
- **Algorithm**:
  1. Preprocessing: sort variables by decreasing diagonal elements and permute elements of Σ accordingly. Compute the Cholesky decomposition $\Sigma = A^T A$.
  2. Initialization: $I_1 = \{1\}$, $x_1 = a_1 / \|a_1\|$.
  3. Compute $i_k = \mathrm{argmax}_{i \notin I_k} \lambda_{\max} \left(\sum_{j \in I_k \cup \{i\}} a_j a_j^T \right)$.
  4. Set $I_{k+1} = I_k \cup \{i_k\}$ and compute x_{k+1} as the leading eigenvector of $\sum_{j \in I_{k+1}} a_j a_j^T$.
  5. Set $k = k + 1$. If $k < n$ go back to step 3.
- **Output**: sparsity patterns I_k.

At every step, I_k represents the set of nonzero elements (or sparsity pattern) of the current point and we can define z_k as the solution to problem (2) given I_k, which is:

$$z_k = \underset{\{z_{I_k^c} = 0,\ \|z\| = 1\}}{\mathrm{argmax}} \; z^T \Sigma z - \rho k,$$

which means that z_k is formed by padding zeros to the leading eigenvector of the submatrix Σ_{I_k,I_k}. Note that the entire algorithm can be written in terms of a factorization $\Sigma = A^TA$ of the matrix Σ, which means significant computational savings when Σ is given as a Gram matrix. The matrices Σ_{I_k,I_k} and $\sum_{i\in I_k} a_i a_i^T$ have the same eigenvalues and their eigenvectors are transformed of each other through the matrix A, i.e., if z is an eigenvector of Σ_{I_k,I_k}, then $A_{I_k}z/\|A_{I_k}z\|$ is an eigenvector of $A_{I_k}A_{I_k}^T$.

3.3 Approximate greedy solution

Computing $n-k$ eigenvalues at each iteration is costly and we can use the fact that uu^T is a subgradient of $\lambda_{\max}$ at X if u is a leading eigenvector of X (Boyd and Vandenberghe, 2004), to get:

$$\lambda_{\max}\left(\sum_{j\in I_k\cup\{i\}} a_j a_j^T\right) \geq \lambda_{\max}\left(\sum_{j\in I_k} a_j a_j^T\right) + (x_k^T a_i)^2, \tag{5}$$

which means that the variance is increasing by at least $(x_k^T a_i)^2$ when variable i is added to I_k. This provides a lower bound on the objective which does not require finding $n-k$ eigenvalues at each iteration. We then derive the following algorithm:

Approximate Greedy Search Algorithm.

- **Input**: $\Sigma \in \mathbf{R}^{n\times n}$
- **Algorithm**:
    1. Preprocessing. Sort variables by decreasing diagonal elements and permute elements of Σ accordingly. Compute the Cholesky decomposition $\Sigma = A^TA$.
    2. Initialization: $I_1 = \{1\}$, $x_1 = a_1/\|a_1\|$.
    3. Compute $i_k = \text{argmax}_{i\notin I_k}(x_k^T a_i)^2$
    4. Set $I_{k+1} = I_k\cup\{i_k\}$ and compute x_{k+1} as the leading eigenvector of $\sum_{j\in I_{k+1}} a_j a_j^T$.
    5. Set $k = k+1$. If $k < n$ go back to step 3.
- **Output**: sparsity patterns I_k.

Again, at every step, I_k represents the set of nonzero elements (or sparsity pattern) of the current point and we can define z_k as the solution to problem (2) given I_k, which is:

$$z_k = \underset{\{z_{I_k^c}=0,\ \|z\|=1\}}{\text{argmax}} \; z^T\Sigma z - \rho k,$$

which means that z_k is formed by padding zeros to the leading eigenvector of the submatrix Σ_{I_k,I_k}. Better points can be found by testing the variables corresponding to the p largest values of $(x_k^T a_i)^2$ instead of picking only the best one.

3.4 Computational Complexity

The complexity of computing a greedy regularization path using the classic greedy algorithm in Section 3.2 is $O(n^4)$: at each step k, it computes $(n-k)$ maximum eigenvalue of matrices with size k. The approximate algorithm in Section 3.3 computes a full path in $O(n^3)$: the first Cholesky decomposition is $O(n^3)$, while the complexity of the k-th iteration is $O(k^2)$ for the maximum eigenvalue problem and $O(n^2)$ for computing all products $(x^T a_j)$. Also, when the matrix Σ is directly given as a Gram matrix $A^T A$ with $A \in \mathbf{R}^{q\times n}$ with $q < n$, it is advantageous to use A directly as the square root of Σ and the total complexity of getting the path up to cardinality p is then reduced to $O(p^3 + p^2 n)$ (which is $O(p^3)$ for the eigenvalue problems and $O(p^2 n)$ for computing the vector products).

4. Convex Relaxation

In Section 2, we showed that the original sparse PCA problem (2) could also be written as in (4):

$$\phi(\rho) = \max_{\|x\|=1} \sum_{i=1}^{n} ((a_i^T x)^2 - \rho)_+.$$

Because the variable x appears solely through $X = xx^T$, we can reformulate the problem in terms of X only, using the fact that when $\|x\| = 1$, $X = xx^T$ is equivalent to $\mathbf{Tr}(X) = 1$, $X \succeq 0$ and $\mathbf{Rank}(X) = 1$. We thus rewrite (4) as:

$$\begin{array}{rll} \phi(\rho) = & \text{max.} & \sum_{i=1}^{n} (a_i^T X a_i - \rho)_+ \\ & \text{s.t.} & \mathbf{Tr}(X) = 1,\ \mathbf{Rank}(X) = 1 \\ & & X \succeq 0. \end{array}$$

Note that because we are maximizing a convex function over the convex set (spectahedron) $\Delta_n = \{X \in \mathbf{S}_n : \ \mathbf{Tr}(X) = 1,\ X \succeq 0\}$, the solution must be an extreme point of Δ_n (i.e. a rank one matrix), hence we can drop the rank constraint here. Unfortunately, $X \mapsto (a_i^T X a_i - \rho)_+$, the function we are *maximizing*, is convex in X and not concave, which means that the above problem is still hard. However, we show below that on rank one elements of Δ_n, it is also equal to a concave function of X, and we use this to produce a semidefinite relaxation of problem (2).

Proposition 1 *Let $A \in \mathbf{R}^{n\times n}$, $\rho \geq 0$ and denote by $a_1, \ldots, a_n \in \mathbf{R}^n$ the columns of A, an upper bound on:*

$$\begin{array}{rll} \phi(\rho) = & \textit{max.} & \sum_{i=1}^{n} (a_i^T X a_i - \rho)_+ \\ & \textit{s.t.} & \mathbf{Tr}(X) = 1,\ X \succeq 0,\ \mathbf{Rank}(X) = 1 \end{array} \tag{6}$$

can be computed by solving

$$\begin{array}{rll} \psi(\rho) = & \textit{max.} & \sum_{i=1}^{n} \mathbf{Tr}(X^{1/2} B_i X^{1/2})_+ \\ & \textit{s.t.} & \mathbf{Tr}(X) = 1,\ X \succeq 0. \end{array} \tag{7}$$

in the variables $X \in \mathbf{S}_n$, where $B_i = a_i a_i^T - \rho \mathbf{I}$, or also:

$$\begin{array}{rll} \psi(\rho) = & \textit{max.} & \sum_{i=1}^{n} \mathbf{Tr}(P_i B_i) \\ & \textit{s.t.} & \mathbf{Tr}(X) = 1,\ X \succeq 0,\ X \succeq P_i \succeq 0, \end{array} \tag{8}$$

which is a semidefinite program in the variables $X \in \mathbf{S}_n,\ P_i \in \mathbf{S}_n$.

Proof We let $X^{1/2}$ denote the positive square root (i.e. with nonnegative eigenvalues) of a symmetric positive semi-definite matrix X. In particular, if $X = xx^T$ with $\|x\| = 1$, then $X^{1/2} = X = xx^T$, and for all $\beta \in \mathbf{R}$, βxx^T has one eigenvalue equal to β and $n-1$ equal to 0, which implies $\mathbf{Tr}(\beta xx^T)_+ = \beta_+$. We thus get:

$$\begin{aligned} (a_i^T X a_i - \rho)_+ &= \mathbf{Tr}((a_i^T xx^T a_i - \rho)xx^T)_+ \\ &= \mathbf{Tr}(x(x^T a_i a_i^T x - \rho)x^T)_+ \\ &= \mathbf{Tr}(X^{1/2} a_i a_i^T X^{1/2} - \rho X)_+ = \mathbf{Tr}(X^{1/2}(a_i a_i^T - \rho \mathbf{I})X^{1/2})_+. \end{aligned}$$

For any symmetric matrix B, the function $X \mapsto \mathbf{Tr}(X^{1/2}BX^{1/2})_+$ is concave on the set of symmetric positive semidefinite matrices, because we can write it as:

$$\begin{aligned} \mathbf{Tr}(X^{1/2}BX^{1/2})_+ &= \max_{\{0 \preceq P \preceq X\}} \mathbf{Tr}(PB) \\ &= \min_{\{Y \succeq B,\ Y \succeq 0\}} \mathbf{Tr}(YX), \end{aligned}$$

where this last expression is a concave function of X as a pointwise minimum of affine functions. We can now relax the original problem into a convex optimization problem by simply dropping the rank constraint, to get:

$$\begin{array}{rll} \psi(\rho) \equiv & \text{max.} & \sum_{i=1}^n \mathbf{Tr}(X^{1/2} a_i a_i^T X^{1/2} - \rho X)_+ \\ & \text{s.t.} & \mathbf{Tr}(X) = 1,\ X \succeq 0, \end{array}$$

which is a convex program in $X \in \mathbf{S}_n$. Note that because B_i has at most one nonnegative eigenvalue, we can replace $\mathbf{Tr}(X^{1/2} a_i a_i^T X^{1/2} - \rho X)_+$ by $\lambda_{\max}(X^{1/2} a_i a_i^T X^{1/2} - \rho X)_+$ in the above program. Using the representation of $\mathbf{Tr}(X^{1/2}BX^{1/2})_+$ detailed above, problem (7) can be written as a semidefinite program:

$$\begin{array}{rll} \psi(\rho) = & \text{max.} & \sum_{i=1}^n \mathbf{Tr}(P_i B_i) \\ & \text{s.t.} & \mathbf{Tr}(X) = 1,\ X \succeq 0,\ X \succeq P_i \succeq 0, \end{array}$$

in the variables $X \in \mathbf{S}_n,\ P_i \in \mathbf{S}_n$, which is the desired result. ■

Note that we always have $\psi(\rho) \geq \phi(\rho)$ and when the solution to the above semidefinite program has rank one, $\psi(\rho) = \phi(\rho)$ and the semidefinite relaxation (8) is *tight*. This simple fact allows us to derive sufficient global optimality conditions for the original sparse PCA problem.

5. Optimality Conditions

In this section, we derive necessary and sufficient conditions to test the optimality of solutions to the relaxations obtained in Sections 3, as well as sufficient condition for the tightness of the semidefinite relaxation in (8).

5.1 Dual problem and optimality conditions

We first derive the dual problem to (8) as well as the Karush-Kuhn-Tucker (KKT) optimality conditions:

Lemma 2 *Let $A \in \mathbf{R}^{n\times n}$, $\rho \geq 0$ and denote by $a_1, \ldots, a_n \in \mathbf{R}^n$ the columns of A. The dual of problem (8):*

$$\begin{array}{rll} \psi(\rho) = & \textit{max.} & \sum_{i=1}^n \mathbf{Tr}(P_i B_i) \\ & \textit{s.t.} & \mathbf{Tr}(X) = 1,\ X \succeq 0,\ X \succeq P_i \succeq 0, \end{array}$$

in the variables $X \in \mathbf{S}_n$, $P_i \in \mathbf{S}_n$, is given by:

$$\begin{array}{ll} \textit{min.} & \lambda_{\max}\left(\sum_{i=1}^n Y_i\right) \\ \textit{s.t.} & Y_i \succeq B_i,\ Y_i \succeq 0, \quad i = 1, \ldots, n. \end{array} \tag{9}$$

in the variables $Y_i \in \mathbf{S}_n$. Furthermore, the KKT optimality conditions for this pair of semidefinite programs are given by:

$$\left\{ \begin{array}{l} \left(\sum_{i=1}^n Y_i\right) X = \lambda_{\max}\left(\sum_{i=1}^n Y_i\right) X \\ (X - P_i) Y_i = 0,\ P_i B_i = P_i Y_i \\ Y_i \succeq B_i,\ Y_i, X, P_i \succeq 0,\ X \succeq P_i,\ \mathbf{Tr}\, X = 1. \end{array} \right. \tag{10}$$

Proof Starting from:

$$\begin{array}{ll} \text{max.} & \sum_{i=1}^n \mathbf{Tr}(P_i B_i) \\ \text{s.t.} & 0 \preceq P_i \preceq X \\ & \mathbf{Tr}(X) = 1,\ X \succeq 0, \end{array}$$

we can form the Lagrangian as:

$$L(X, P_i, Y_i) = \sum_{i=1}^n \mathbf{Tr}(P_i B_i) + \mathbf{Tr}(Y_i(X - P_i))$$

in the variables $X, P_i, Y_i \in \mathbf{S}_n$, with $X, P_i, Y_i \succeq 0$ and $\mathbf{Tr}(X) = 1$. Maximizing $L(X, P_i, Y_i)$ in the primal variables X and P_i leads to problem (9). The KKT conditions for this primal-dual pair of SDP can be derived from Boyd and Vandenberghe (2004, p.267).

■

5.2 Optimality conditions for rank one solutions

We now derive the KKT conditions for problem (8) for the particular case where we are given a rank one candidate solution $X = xx^T$ and need to test its optimality. These necessary and sufficient conditions for the optimality of $X = xx^T$ for the convex relaxation then provide sufficient conditions for *global* optimality for the non-convex problem (2).

Lemma 3 *Let $A \in \mathbf{R}^{n\times n}$, $\rho \geq 0$ and denote by $a_1, \ldots, a_n \in \mathbf{R}^n$ the columns of A. The rank one matrix $X = xx^T$ is an optimal solution of (8) if and only if there are matrices $Y_i \in \mathbf{S}_n,\ i = 1, \ldots, n$ such that:*

$$\left\{\begin{array}{l} \lambda_{\max}\left(\sum_{i=1}^n Y_i\right) = \sum_{i\in I}((a_i^T x)^2 - \rho) \\ x^T Y_i x = \left\{\begin{array}{l} (a_i^T x)^2 - \rho \ \textit{if}\ i \in I \\ 0 \ \textit{if}\ i \in I^c \end{array}\right. \\ Y_i \succeq B_i,\ Y_i \succeq 0. \end{array}\right. \tag{11}$$

where $B_i = a_i a_i^T - \rho\mathbf{I},\ i = 1, \ldots, n$ and I^c is the complement of the set I defined by:

$$\max_{i\notin I}(a_i^T x)^2 \leq \rho \leq \min_{i\in I}(a_i^T x)^2.$$

Furthermore, x must be a leading eigenvector of both $\sum_{i\in I} a_i a_i^T$ and $\sum_{i=1}^n Y_i$.

Proof We apply Lemma 2 given $X = xx^T$. The condition $0 \preceq P_i \preceq xx^T$ is equivalent to $P_i = \alpha_i xx^T$ and $\alpha_i \in [0, 1]$. The equation $P_i B_i = XY_i$ is then equivalent to $\alpha_i(x^T B_i x - x^T Y_i x) = 0$, with $x^T B_i x = (a_i^T x)^2 - \rho$ and the condition $(X - P_i)Y_i = 0$ becomes $x^T Y_i x(1 - \alpha_i) = 0$. This means that $x^T Y_i x = ((a_i^T x)^2 - \rho)_+$ and the first-order condition in (10) becomes $\lambda_{\max}\left(\sum_{i=1}^n Y_i\right) = x^T\left(\sum_{i=1}^n Y_i\right)x$. Finally, we recall from Section 2 that:

$$\begin{aligned} \textstyle\sum_{i\in I}((a_i^T x)^2 - \rho) &= \max_{\|x\|=1} \max_{u\in\{0,1\}^n} \sum_{i=1}^n u_i((a_i^T x)^2 - \rho) \\ &= \max_{u\in\{0,1\}^n} \lambda_{\max}(A\,\mathbf{diag}(u)A^T) - \rho\mathbf{1}^T u \end{aligned}$$

hence x must also be a leading eigenvector of $\sum_{i\in I} a_i a_i^T$. ■

The previous lemma shows that given a candidate vector x, we can test the optimality of $X = xx^T$ for the semidefinite program (7) by solving a semidefinite feasibility problem in the variables $Y_i \in \mathbf{S}_n$. If this (rank one) solution xx^T is indeed optimal for the semidefinite relaxation, then x must also be *globally* optimal for the original nonconvex combinatorial problem in (2), so the above lemma provides sufficient global optimality conditions for the combinatorial problem (2) based on the (necessary and sufficient) optimality conditions for the convex relaxation (7) given in lemma 2. In practice, we are only given a sparsity pattern I (using the results of Section 3 for example) rather than the vector x, but Lemma 3 also shows that given I, we can get the vector x as the leading eigenvector of $\sum_{i\in I} a_i a_i^T$.

The next result provides more refined conditions under which such a pair (I, x) is optimal for some value of the penalty $\rho > 0$ based on a local optimality argument. In particular, they allow us to fully specify the dual variables Y_i for $i \in I$.

Proposition 4 *Let $A \in \mathbf{R}^{n\times n}$, $\rho \geq 0$ and denote by $a_1, \ldots, a_n \in \mathbf{R}^n$ the columns of A. Let x be the largest eigenvector of $\sum_{i\in I} a_i a_i^T$. Let I be such that:*

$$\max_{i\notin I}(a_i^T x)^2 < \rho < \min_{i\in I}(a_i^T x)^2, \tag{12}$$

the matrix $X = xx^T$ is optimal for problem (8) if and only if there are matrices $Y_i \in \mathbf{S}^n$ satisfying

$$\lambda_{\max}\left(\sum_{i\in I}\frac{B_i xx^T B_i}{x^T B_i x} + \sum_{i\in I^c} Y_i\right) \leq \sum_{i\in I}((a_i^T x)^2 - \rho), \tag{13}$$

with $Y_i \succeq B_i - \frac{B_i xx^T B_i}{x^T B_i x}$, $Y_i \succeq 0$, where $B_i = a_i a_i^T - \rho\mathbf{I}$, $i = 1, \ldots, n$.

Proof We first prove the necessary condition by computing a first order expansion of the functions $F_i : X \mapsto \mathbf{Tr}(X^{1/2}B_iX^{1/2})_+$ around $X = xx^T$. The expansion is based on the results in Appendix A which show how to compute derivatives of eigenvalues and projections on eigensubspaces. More precisely, Lemma 10 states that if $x^T Bx > 0$, then, for any $Y \succeq 0$:

$$F_i((1-t)xx^T + tY) = F_i(xx^T) + \frac{t}{x^T B_i x}\mathbf{Tr}\, B_i xx^T B_i(Y - xx^T) + O(t^{3/2}),$$

while if $x^T Bx < 0$, then, for any $Y \succeq 0$,:

$$F_i((1-t)xx^T + tY) = t_+\,\mathbf{Tr}\left(Y^{1/2}\left(B_i - \frac{B_i xx^T B_i}{x^T B_i x}\right)Y^{1/2}\right)_+ + O(t^{3/2}).$$

Thus if $X = xx^T$ is a global maximum of $\sum_i F_i(X)$, then this first order expansion must reflect the fact that it is also local maximum, i.e. for all $Y \in \mathbf{S}^n$ such that $Y \succeq 0$ and $\mathbf{Tr}\,Y = 1$, we must have:

$$\lim_{t\to 0_+}\frac{1}{t}\sum_{i=1}^{n}[F_i((1-t)xx^T + tY) - F_i(xx^T)] \leq 0,$$

which is equivalent to:

$$-\sum_{i\in I} x^T B_i x + \mathbf{Tr}\,Y\left(\sum_{i\in I}\frac{B_i xx^T B_i}{x^T B_i x}\right) + \sum_{i\in I^c}\mathbf{Tr}\left(Y^{1/2}\left(B_i - \frac{B_i xx^T B_i}{x^T B_i x}\right)Y^{1/2}\right)_+ \leq 0.$$

Thus if $X = xx^T$ is optimal, with $\sigma = \sum_{i\in I} x^T B_i x$, we get:

$$\max_{Y\succeq 0, \mathbf{Tr}\,Y=1}\mathbf{Tr}\,Y\left(\sum_{i\in I}\frac{B_i xx^T B_i}{x^T B_i x} - \sigma\mathbf{I}\right) + \sum_{i\in I^c}\mathbf{Tr}\left(Y^{1/2}\left(B_i - B_i x(x^T B_i x)^{\dagger} x^T B_i\right)Y^{1/2}\right)_+ \leq 0$$

which is also in dual form (using the same techniques as in the proof of Proposition 1):

$$\min_{\{Y_i\succeq B_i - \frac{B_i xx^T B_i}{x^T B_i x}, Y_i\succeq 0\}}\lambda_{\max}\left(\sum_{i\in I}\frac{B_i xx^T B_i}{x^T B_i x} + \sum_{i\in I^c} Y_i\right) \leq \sigma,$$

which leads to the necessary condition. In order to prove sufficiency, the only non trivial condition to check in Lemma 3 is that $x^T Y_i x = 0$ for $i \in I^c$, which is a consequence of the inequality:

$$x^T\left(\sum_{i\in I}\frac{B_i xx^T B_i}{x^T B_i x} + \sum_{i\in I^c} Y_i\right)x \leq \lambda_{\max}\left(\sum_{i\in I}\frac{B_i xx^T B_i}{x^T B_i x} + \sum_{i\in I^c} Y_i\right) \leq x^T\left(\sum_{i\in I}\frac{B_i xx^T B_i}{x^T B_i x}\right)x.$$

This concludes the proof. ■

The original optimality conditions in (3) are highly degenerate in Y_i and this result refines these optimality conditions by invoking the local structure. The local optimality analysis in proposition 4 gives more specific constraints on the dual variables Y_i. For $i \in I$, Y_i must be equal to $B_i x x^T B_i / x^T B_i x$, while if $i \in I^c$, we must have $Y_i \succeq B_i - B_i x x^T B_i / x^T B_i x$, which is a stricter condition than $Y_i \succeq B_i$ (because $x^T B_i x < 0$).

5.3 Efficient Optimality Conditions

The condition presented in Proposition 4 still requires solving a large semidefinite program. In practice, good candidates for $Y_i,\ i \in I^c$ can be found by solving for minimum trace matrices satisfying the feasibility conditions of proposition 4. As we will see below, this can be formulated as a semidefinite program which can be solved explicitly.

Lemma 5 *Let $A \in \mathbf{R}^{n\times n}$, $\rho \geq 0$, $x \in \mathbf{R}^n$ and $B_i = a_i a_i^T - \rho \mathbf{I}$ with $a_1, \ldots, a_n \in \mathbf{R}^n$ the columns of A. If $(a_i^T x)^2 < \rho$ and $\|x\| = 1$, an optimal solution of the semidefinite program:*

$$\begin{array}{ll} \textit{minimize} & \mathbf{Tr}\, Y_i \\ \textit{subject to} & Y_i \succeq B_i - \frac{B_i x x^T B_i}{x^T B_i x},\ x^T Y_i x = 0,\ Y_i \succeq 0, \end{array}$$

is given by:

$$Y_i = \max\left\{0, \rho \frac{(a_i^T a_i - \rho)}{(\rho - (a_i^T x)^2)}\right\} \frac{(\mathbf{I} - x x^T) a_i a_i^T (\mathbf{I} - x x^T)}{\|(\mathbf{I} - x x^T) a_i\|^2}. \tag{14}$$

Proof Let us write $M_i = B_i - \frac{B_i x x^T B_i}{x^T B_i x}$, we first compute:

$$\begin{aligned} a_i^T M_i a_i &= (a_i^T a_i - \rho) a_i^T a_i - \frac{(a_i^T a_i a_i^T x - \rho a_i^T x)^2}{(a_i^T x)^2 - \rho} \\ &= \frac{(a_i^T a_i - \rho)}{\rho - (a_i^T x)^2} \rho (a_i^T a_i - (a_i^T x)^2). \end{aligned}$$

When $a_i^T a_i \leq \rho$, the matrix M_i is negative semidefinite, because $\|x\| = 1$ means $a_i^T M a_i \leq 0$ and $x^T M x = a_i^T M x = 0$. The solution of the minimum trace problem is then simply $Y_i = 0$. We now assume that $a_i^T a_i > \rho$ and first check feasibility of the candidate solution Y_i in (14). By construction, we have $Y_i \succeq 0$ and $Y_i x = 0$, and a short calculation shows that:

$$\begin{aligned} a_i^T Y_i a_i &= \rho \frac{(a_i^T a_i - \rho)}{(\rho - (a_i^T x)^2)} (a_i^T a_i - (a_i^T x)^2) \\ &= a_i^T M_i a_i. \end{aligned}$$

We only need to check that $Y_i \succeq M_i$ on the subspace spanned by a_i and x, for which there is equality. This means that Y_i in (14) is feasible and we now check its optimality. The dual of the original semidefinite program can be written:

$$\begin{array}{ll} \text{maximize} & \mathbf{Tr}\, P_i M_i \\ \text{subject to} & \mathbf{I} - P_i + \nu x x^T \succeq 0 \\ & P_i \succeq 0, \end{array}$$

and the KKT optimality conditions for this problem are written:

$$\begin{cases} Y_i(\mathbf{I} - P_i + \nu xx^T) = 0, \; P_i(Y_i - M_i) = 0, \\ \mathbf{I} - P_i + \nu xx^T \succeq 0, \\ P_i \succeq 0, \; Y_i \succeq 0, \; Y_i \succeq M_i, \; Y_i xx^T = 0, \quad i \in I^c. \end{cases}$$

Setting $P_i = Y_i \mathbf{Tr}\, Y_i / \mathbf{Tr}\, Y_i^2$ and ν sufficiently large makes these variables dual feasible. Because all contributions of x are zero, $\mathbf{Tr}\, Y_i(Y_i - M_i)$ is proportional to $\mathbf{Tr}\, a_i a_i^T (Y_i - M_i)$ which is equal to zero and Y_i in (14) satisifies the KKT optimality conditions. ■

We summarize the results of this section in the theorem below, which provides sufficient optimality conditions on a sparsity pattern I.

Theorem 6 *Let $A \in \mathbf{R}^{n\times n}$, $\rho \geq 0$, $\Sigma = A^T A$ with $a_1, \ldots, a_n \in \mathbf{R}^n$ the columns of A. Given a sparsity pattern I, setting x to be the largest eigenvector of $\sum_{i\in I} a_i a_i^T$, if there is a $\rho^* \geq 0$ such that the following conditions hold:*

$$\max_{i\in I^c} (a_i^T x)^2 < \rho^* < \min_{i\in I} (a_i^T x)^2 \quad \text{and} \quad \lambda_{\max}\left(\sum_{i=1}^n Y_i\right) \leq \sum_{i\in I}((a_i^T x)^2 - \rho^*),$$

with the dual variables Y_i for $i \in I^c$ defined as in (14) and:

$$Y_i = \frac{B_i x x^T B_i}{x^T B_i x}, \quad \text{when } i \in I,$$

then the sparsity pattern I is globally optimal for the sparse PCA problem (2) with $\rho = \rho^*$ and we can form an optimal solution z by solving the maximum eigenvalue problem:

$$z = \underset{\{z_{I^c}=0, \; \|z\|=1\}}{\operatorname{argmax}} \quad z^T \Sigma z.$$

Proof Following proposition 4 and lemma 5, the matrices Y_i are dual optimal solutions corresponding to the primal optimal solution $X = xx^T$ in (7). Because the primal solution has rank one, the semidefinite relaxation (8) is tight so the pattern I is optimal for (2) and Section 2 shows that z is a globally optimal solution to (2) with $\rho = \rho^*$. ■

5.4 Gap minimization: finding the optimal ρ

All we need now is an efficient algorithm to find ρ^* in theorem 6. As we will show below, when the dual variables Y_i^c are defined as in (14), the duality gap in (2) is a convex function of ρ hence, given a sparsity pattern I, we can efficiently search for the best possible ρ (which must belong to an *interval*) by performing a few binary search iterations.

Lemma 7 *Let $A \in \mathbf{R}^{n\times n}$, $\rho \geq 0$, $\Sigma = A^T A$ with $a_1, \ldots, a_n \in \mathbf{R}^n$ the columns of A. Given a sparsity pattern I, setting x to be the largest eigenvector of $\sum_{i\in I} a_i a_i^T$, with the dual variables Y_i for $i \in I^c$ defined as in (14) and:*

$$Y_i = \frac{B_i x x^T B_i}{x^T B_i x}, \quad \text{when } i \in I.$$

The duality gap in (2) which is given by:

$$\mathrm{gap}(\rho) \equiv \lambda_{\max}\left(\sum_{i=1}^{n} Y_i\right) - \sum_{i\in I}((a_i^T x)^2 - \rho),$$

is a convex function of ρ when

$$\max_{i\notin I}(a_i^T x)^2 < \rho < \min_{i\in I}(a_i^T x)^2.$$

Proof For $i \in I$ and $u \in \mathbf{R}^n$, we have

$$u^T Y_i u = \frac{(u^T a_i a_i^T x - \rho u^T x)^2}{(a_i^T x)^2 - \rho},$$

which is a convex function of ρ (Boyd and Vandenberghe, 2004, p.73). For $i \in I^c$, we can write:

$$\frac{\rho(a_i^T a_i - \rho)}{\rho - (a_i^T x)^2} = -\rho + (a_i^T a_i - (a_i^T x)^2)\left(1 + \frac{(a_i^T x)^2}{\rho - (a_i^T x)^2}\right),$$

hence $\max\{0, \rho(a_i^T a_i - \rho)/(\rho - (a_i^T x)^2)\}$ is also a convex function of ρ. This means that:

$$u^T Y_i u = \max\left\{0, \rho\frac{(a_i^T a_i - \rho)}{(\rho - (a_i^T x)^2)}\right\} \frac{(u^T a_i - (x^T u)(x^T a_i))^2}{\|(\mathbf{I} - xx^T)a_i\|^2}$$

is convex in ρ when $i \in I^c$. We conclude that $\sum_{i=1}^n u^T Y_i u$ is convex, hence:

$$\mathrm{gap}(\rho) = \max_{\|u\|=1} \sum_{i=1}^{n} u^T Y_i u - \sum_{i\in I}((a_i^T x)^2 - \rho)$$

is also convex in ρ as a pointwise maximum of convex functions of ρ. ■

This result shows that the set of ρ for which the pattern I is optimal must be an interval. It also suggests an efficient procedure for testing the optimality of a given pattern I. We first compute x as a leading eigenvector $\sum_{i\in I} a_i a_i^T$. We then compute an interval in ρ for which x satisfies the basic consistency condition:

$$\max_{i\notin I}(a_i^T x)^2 \equiv \rho_{\min} \le \rho \le \rho_{\max} \equiv \min_{i\in I}(a_i^T x)^2.$$

Note that this interval could be empty, in which case I cannot be optimal. We then minimize $\mathrm{gap}(\rho)$ over the interval $[\rho_{\min}, \rho_{\max}]$. If the minimum is zero for some $\rho = \rho^*$, then the pattern I is optimal for the sparse PCA problem in (2) with $\rho = \rho^*$.

Minimizing the convex function $\mathrm{gap}(\rho)$ can be done very efficiently using binary search. The initial cost of forming the matrix $\sum_{i=1}^n Y_i$, which is a simple outer matrix product, is $O(n^3)$. At each iteration of the binary search, a subgradient of $\mathrm{gap}(\rho)$ can then be computed by solving a maximum eigenvalue problem, at a cost of $O(n^2)$. This means that the complexity of finding the optimal ρ over a given interval $[\rho_{\min}, \rho_{\max}]$ is $O(n^2 \log_2((\rho_{\max} - \rho_{\min})/\epsilon))$, where ϵ is the target precision. Overall then, the total cost of testing the optimality of a pattern I is $O(n^3 + n^2 \log_2((\rho_{\max} - \rho_{\min})/\epsilon))$.

Note that an additional benefit of deriving explicit dual feasible points Y_i is that plugging these solutions into the objective of problem (9):

$$\begin{array}{ll} \text{min.} & \lambda_{\max}\left(\sum_{i=1}^{n} Y_i\right) \\ \text{s.t.} & Y_i \succeq B_i,\ Y_i \succeq 0, \quad i = 1, \ldots, n. \end{array}$$

produces an *upper bound* on the optimum value of the original sparse PCA problem (2) even when the pattern I is not optimal (all we need is a ρ satisfying the consistency condition).

5.5 Solution improvements and randomization

When these conditions are not satisfied, the relaxation (8) has an optimal solution with rank strictly larger than one, hence is not tight. At such a point, we can use a different relaxation such as DSPCA by d'Aspremont et al. (2007b) to try to get a better solution. We can also apply randomization techniques to improve the quality of the solution of problem (8) (Ben-Tal and Nemirovski, 2002).

6. Applications

In this section, we discuss some applications of sparse PCA to subset selection and compressed sensing.

6.1 Subset selection

We consider p data points in $\mathbf{R}^n$, in a data matrix $X \in \mathbf{R}^{p \times n}$. We assume that we are given real numbers $y \in \mathbf{R}^p$ to predict from X using linear regression, estimated by least squares. We are thus looking for $w \in \mathbf{R}^n$ such that $\|y - Xw\|^2$ is minimum. In the subset selection problem, we are looking for sparse coefficients w, i.e., a vector w with many zeros. We thus consider the problem:

$$s(k) = \min_{w \in \mathbf{R}^n,\ \mathbf{Card}\, w \leq k} \|y - Xw\|^2. \tag{15}$$

Using the sparsity pattern $u \in \{0,1\}^n$, and optimizing with respect to w, we have

$$s(\rho) = \min_{u \in \{0,1\}^n,\ \mathbf{1}^T u \leq k} \|y\|^2 - y^T X(u)(X(u)^T X(u))^{-1} X(u)^T y, \tag{16}$$

where $X(u) = X\,\mathbf{diag}(u)$. We can rewrite $y^T X(u)(X(u)^T X(u))^{-1} X(u)^T y$ as the largest generalized eigenvalue of the pair $(X(u)^T yy^T X(u), X(u)^T X(u))$, i.e., as

$$y^T X(u)(X(u)^T X(u))^{-1} X(u)^T y = \max_{w \in \mathbf{R}^n} \frac{w^T X(u)^T yy^T X(u) w}{w^T X(u)^T X(u) w}.$$

We thus have:

$$s(k) = \|y\|^2 - \max_{u \in \{0,1\}^n, \mathbf{1}^T u \leq k} \max_{w \in \mathbf{R}^n} \frac{w^T \,\mathbf{diag}(u) X^T yy^T X\,\mathbf{diag}(u) w}{w^T \,\mathbf{diag}(u) X^T X\,\mathbf{diag}(u)) w}. \tag{17}$$

Given a pattern $u \in \{0,1\}^n$, let

$$s_0 = y^T X(u)(X(u)^T X(u))^{-1} X(u)^T y$$

be the largest generalized eigenvalue corresponding to the pattern u. The pattern is optimal if and only if the largest generalized eigenvalue of the pair $\{X(v)^T yy^T X(v), X(v)^T X(v)\}$ is less than s_0 for any $v \in \{0,1\}^n$ such that $v^T \mathbf{1} = u^T \mathbf{1}$. This is equivalent to the optimality of u for the sparse PCA problem with matrix $X^T yy^T X - s_0 X^T X$, which can be checked using the sparse PCA optimality conditions derived in the previous sections.

Note that unlike in the sparse PCA case, this convex relaxation does not immediately give a simple bound on the optimal value of the subset selection problem. However, we get a bound of the following form: when $v \in \{0,1\}^n$ and $w \in \mathbf{R}^n$ is such that $\mathbf{1}^T v = k$ with:

$$w^T \left(X(v)^T yy^T X(v) - s_0 X(v)^T X(v) \right) w \leq B,$$

where $B \geq 0$ (because s_0 is defined from u), we have:

$$\begin{aligned} \|y\|^2 - s_0 \geq s(k) \quad &\geq \quad \|y\|^2 - s_0 - B \left(\min_{v \in \{0,1\}^n, \mathbf{1}^T v = k} \lambda_{\min}(X(v)^T X(v)) \right)^{-1} \\ &\geq \quad \|y\|^2 - s_0 - B \left(\lambda_{\min}(X^T X) \right)^{-1}. \end{aligned}$$

This bound gives a sufficient condition for optimality in subset selection, for any problem instance and any given subset. This is to be contrasted with the sufficient conditions derived for particular algorithms, such as the LASSO (Yuan and Lin, 2007, Zhao and Yu, 2006) or backward greedy selection (Couvreur and Bresler, 2000). Note that some of these optimality conditions are often based on sparse eigenvalue problems (see Meinshausen and Yu (2006, §2)), hence our convex relaxations helps both in checking sufficient conditions for optimality (before the algorithm is run) and in testing a posteriori the optimality of a particular solution.

6.2 Sparse recovery

Following Candès and Tao (2005) (see also Donoho and Tanner (2005)), we seek to recover a signal $f \in \mathbf{R}^n$ from corrupted measurements $y = Af + e$, where $A \in \mathbf{R}^{m \times n}$ is a coding matrix and $e \in \mathbf{R}^m$ is an unknown vector of errors with low cardinality. This can be reformulated as the problem of finding the sparsest solution to an underdetermined linear system:

$$\begin{array}{ll} \text{minimize} & \|x\|_0 \\ \text{subject to} & Fx = Fy \end{array} \tag{18}$$

where $\|x\|_0 = \mathbf{Card}(x)$ and $F \in \mathbf{R}^{p \times m}$ is a matrix such that $FA = 0$. A classic trick to get good approximate solutions to problem (18) is to substitute the (convex) ℓ_1 norm to the (combinatorial) ℓ_0 objective above, and solve instead:

$$\begin{array}{ll} \text{minimize} & \|x\|_1 \\ \text{subject to} & Fx = Fy, \end{array} \tag{19}$$

which is equivalent to a linear program in $x \in \mathbf{R}^m$. Following Candès and Tao (2005), given a matrix $F \in \mathbf{R}^{p \times m}$ and an integer S such that $0 < S \leq m$, we define its *restricted isometry* constant δ_S as the smallest number such that for any subset $I \subset [1, m]$ of cardinality at most S we have:

$$(1 - \delta_S)\|c\|^2 \leq \|F_I c\|^2 \leq (1 + \delta_S)\|c\|^2, \tag{20}$$

for all $c \in \mathbf{R}^{|I|}$, where F_I is the submatrix of F formed by keeping only the columns of F in the set I. The following result then holds.

Proposition 8 ***(Candès and Tao (2005))***. *Suppose that the restricted isometry constants of a matrix $F \in \mathbf{R}^{p \times m}$ satisfy*

$$\delta_S + \delta_{2S} + \delta_{3S} < 1 \tag{21}$$

for some integer S such that $0 < S \leq m$, then if x is an optimal solution of the convex program:

$$\begin{array}{ll} \textit{minimize} & \|x\|_1 \\ \textit{subject to} & Fx = Fy \end{array}$$

such that $\mathbf{Card}\, x \leq S$ then x is also an optimal solution of the combinatorial problem:

$$\begin{array}{ll} \textit{minimize} & \|x\|_0 \\ \textit{subject to} & Fx = Fy. \end{array}$$

In other words, if condition (21) holds for some matrix F such that $FA = 0$, then perfect recovery of the signal f given $y = Af + e$ provided the error vector satisfies $\mathbf{Card}(e) \leq S$. Our key observation here is that the restricted isometry constant δ_S in condition (21) can be computed by solving the following sparse maximum eigenvalue problem:

$$\begin{array}{rll} (1 + \delta_S) \leq & \text{max.} & x^T(F^TF)x \\ & \text{s. t.} & \mathbf{Card}(x) \leq S \\ & & \|x\| = 1, \end{array}$$

in the variable $x \in \mathbf{R}^m$ and another sparse maximum eigenvalue problem on $\alpha\mathbf{I} - FF^T$ with α sufficiently large, with δ_S computed from the tightest one. In fact, (20) means that:

$$\begin{array}{rcl} (1 + \delta_S) & \leq & \displaystyle\max_{\{I \subset [1,m]:\ |I| \leq S\}} \ \max_{\|c\|=1} c^T F_I^T F_I c \\ & = & \displaystyle\max_{\{u \in \{0,1\}^n:\ \mathbf{1}^T u \leq S\}} \ \max_{\|x\|=1} x^T \,\mathbf{diag}(u) F^T F \,\mathbf{diag}(u) x \\ & = & \displaystyle\max_{\{\|x\|=1,\ \mathbf{Card}(x) \leq S\}} x^T F^T F x, \end{array}$$

hence we can compute an upper bound on δ_S by duality, with:

$$(1 + \delta_S) \leq \inf_{\rho \geq 0} \phi(\rho) + \rho S$$

where $\phi(\rho)$ is defined in (2). This means that while Candès and Tao (2005) obtained an asymptotic proof that some random matrices satisfied the restricted isometry condition (21) with overwhelming probability (i.e. exponentially small probability of failure), whenever they are satisfied, the *tractable* optimality conditions and upper bounds we obtain in

Section 5 for sparse PCA problems allow us to prove, *deterministically*, that a finite dimensional matrix satisfies the restricted isometry condition in (21). Note that Candès and Tao (2005) provide a slightly weaker condition than (21) based on restricted orthogonality conditions and extending the results on sparse PCA to these conditions would increase the maximum S for which perfect recovery holds. In practice however, we will see in Section 7.3 that the relaxations in (9) and d'Aspremont et al. (2007b) do provide very tight upper bounds on sparse eigenvalues of random matrices but solving these semidefinite programs for very large scale instances remains a significant challenge.

7. Numerical Results

In this section, we first compare the various methods detailed here on artificial examples, then test their performance on a biological data set. PathSPCA, a MATLAB code reproducing these results may be downloaded from the authors' web pages.

7.1 Artificial Data

We generate a matrix U of size 150 with uniformly distributed coefficients in $[0,1]$. We let $v \in \mathbf{R}^{150}$ be a sparse vector with:

$$v_i = \begin{cases} 1 & \text{if } i \leq 50 \\ 1/(i-50) & \text{if } 50 < i \leq 100 \\ 0 & \text{otherwise} \end{cases}$$

We form a test matrix $\Sigma = U^T U + \sigma v v^T$, where σ is the signal-to-noise ratio. We first compare the relative performance of the algorithms in Section 3 at identifying the correct sparsity pattern in v given the matrix Σ. The resulting ROC curves are plotted in figure 1 for $\sigma = 2$. On this example, the computing time for the approximate greedy algorithm in Section 3.3 was 3 seconds versus 37 seconds for the full greedy solution in Section 3.2. Both algorithms produce almost identical answers. We can also see that both sorting and thresholding ROC curves are dominated by the greedy algorithms.

We then plot the variance versus cardinality tradeoff curves for various values of the signal-to-noise ratio. In figure 2, We notice that the magnitude of the error (duality gap) decreases with the signal-to-noise ratio. Also, because of the structure of our problem, there is a kink in the variance at the (exact) cardinality 50 in each of these curves. Note that for each of these examples, the error (duality gap) is minimal precisely at the kink.

Next, we use the DSPCA algorithm of d'Aspremont et al. (2007b) to find better solutions where the greedy codes have failed to obtain globally optimal solutions. In d'Aspremont et al. (2007b), it was shown that an upper bound on (2) can be computed as:

$$\phi(\rho) \leq \min_{|U_{ij}| \leq \rho} \lambda_{\max}(\Sigma + U).$$

which is a convex problem in the matrix $U \in \mathbf{S}_n$. Note however that the cost of solving this relaxation for a *single* ρ is $O(n^4\sqrt{\log n})$ versus $O(n^3)$ for a full path of approximate solutions. Also, the results in d'Aspremont et al. (2007b) do not provide any hint on the value of ρ, but we can use the breakpoints coming from suboptimal points in the greedy

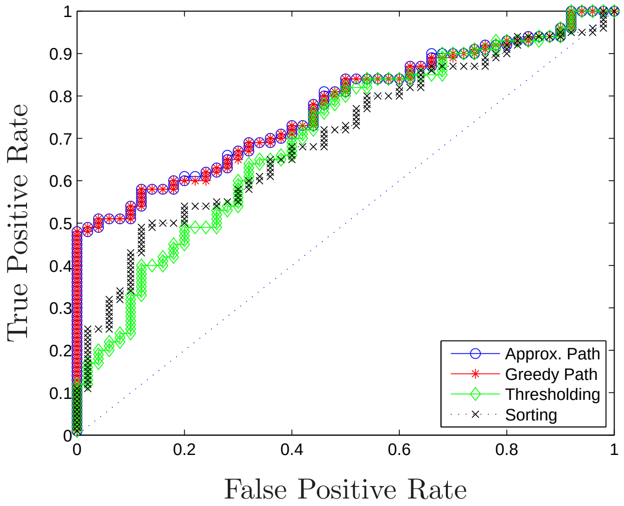

Figure 1: ROC curves for sorting, thresholding, fully greedy solutions (Section 3.2) and approximate greedy solutions (Section 3.3) for $\sigma = 2$.

search algorithms in Section 3.3 and the consistency intervals in Eq. (12). In figure 2 we plot the variance versus cardinality tradeoff curve for $\sigma = 10$. We plot greedy variances (solid line), dual upper bounds from Section 5.3 (dotted line) and upper bounds computed using DSPCA (dashed line).

7.2 Subset selection

We now present simulation experiments on synthetic datasets for the subset selection problem. We consider data sets generated from a sparse linear regression problem and study optimality for the subset selection problem, given the exact cardinality of the generating vector. In this setting, it is known that regularization by the ℓ_1-norm, a procedure also known as the Lasso (Tibshirani, 1996), will asymptotically lead to the correct solution if and only if a certain consistency condition is satisfied (Yuan and Lin, 2007, Zhao and Yu, 2006). Our results provide here a tractable test the optimality of solutions obtained from various algorithms such as the Lasso, forward greedy or backward greedy algorithms.

In Figure 3, we consider two pairs of randomly generated examples in dimension 16, one for which the lasso is provably consistent, one for which it isn't. We perform 50 simulations with 1000 samples and varying noise and compute the average frequency of optimal subset selection for Lasso and greedy backward algorithm together with the frequency of provable optimality (i.e., where our method did ensure a posteriori that the point was optimal). We can see that the backward greedy algorithm exhibits good performance (even in the Lasso-inconsistent case) and that our sufficient optimality condition is satisfied as long as there is not too much noise. In Figure 4, we plot the average mean squared error versus cardinality, over 100 replications, using forward (dotted line) and backward (circles) selection, the Lasso (large dots) and exhaustive search (solid line). The plot on the left shows the results when the Lasso consistency condition is satisfied, while the plot on the right shows the mean

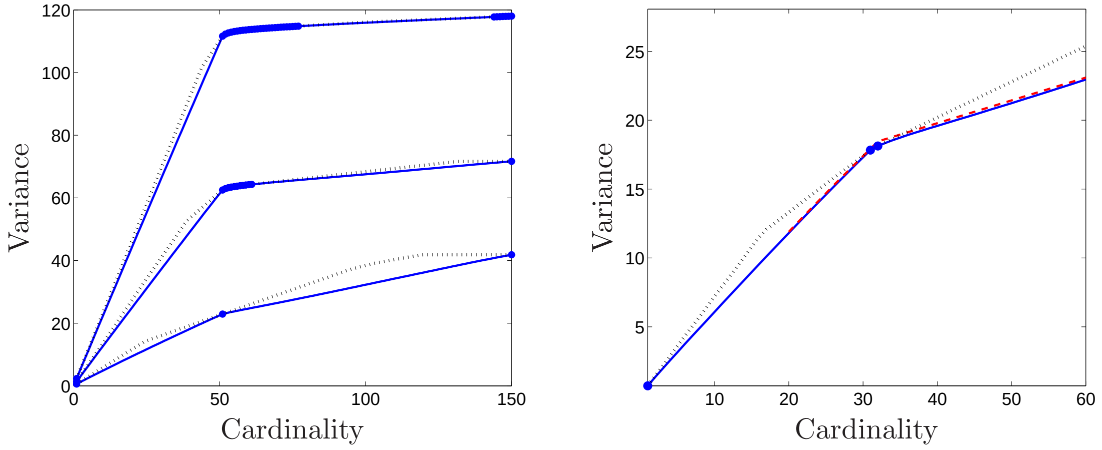

Figure 2: *Left:* variance versus cardinality tradeoff curves for $\sigma = 10$ (bottom), $\sigma = 50$ and $\sigma = 100$ (top). We plot the variance (solid line) and the dual upper bounds from Section 5.3 (dotted line) for each target cardinality. *Right:* variance versus cardinality tradeoff curve for $\sigma = 10$. We plot greedy variances (solid line), dual upper bounds from Section 5.3 (dotted line) and upper bounds computed using DSPCA (dashed line). Optimal points (for which the relative duality gap is less than 10^{-4}) are in bold.

squared errors when the consistency condition is not satisfied. The two sets of figures do show that the LASSO is consistent only when the consistency condition is satisfied, while the backward greedy algorithm finds the correct pattern if the noise is small enough (Couvreur and Bresler, 2000) even in the LASSO inconsistent case.

7.3 Sparse recovery

Following the results of Section 6.2, we compute the upper and lower bounds on sparse eigenvalues produced using various algorithms. We study the following problem:

$$\begin{array}{ll} \text{maximize} & x^T \Sigma x \\ \text{subject to} & \textbf{Card}(x) \leq S \\ & \|x\| = 1, \end{array}$$

where we pick F to be normally distributed and small enough so that computing sparse eigenvalues by exhaustive search is numerically feasible. We plot the maximum sparse eigenvalue versus cardinality, obtained using exhaustive search (solid line), the approximate greedy (dotted line) and fully greedy (dashed line) algorithms. We also plot the upper bounds obtained by minimizing the gap of a rank one solution (squares), by solving the semidefinite relaxation explicitly (stars) and by solving the DSPCA dual (diamonds). On the left, we use a matrix $\Sigma = F^T F$ with F Gaussian. On the right, $\Sigma = uu^T/\|u\|^2 + 2V$, where $u_i = 1/i,\ i = 1, \ldots, n$ and V is matrix with coefficients uniformly distributed in $[0, 1]$. Almost all algorithms are provably optimal in the noisy rank one case (as well as in many

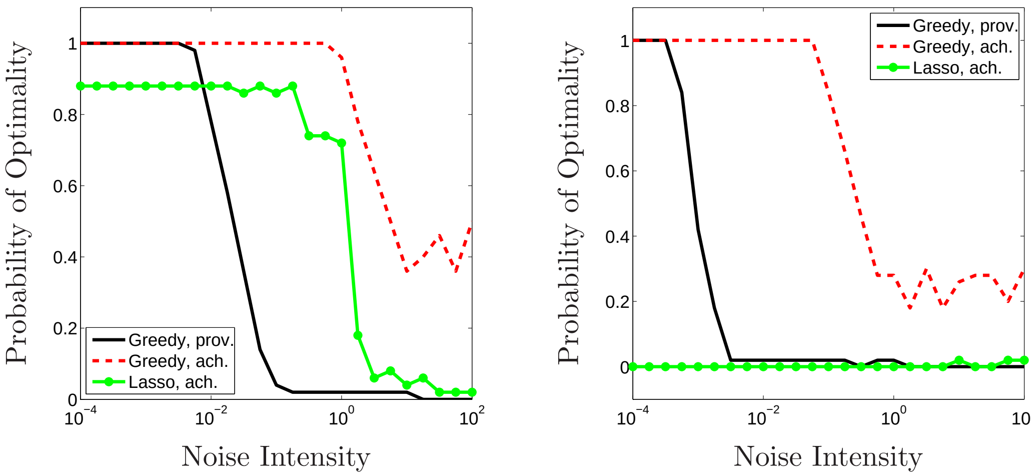

Figure 3: Backward greedy algorithm and Lasso. We plot the probability of achieved (dotted line) and provable (solid line) optimality versus noise for greedy selection against Lasso (large dots), for the subset selection problem on a noisy sparse vector. *Left:* Lasso consistency condition satisfied. *Right:* consistency condition not satisfied.

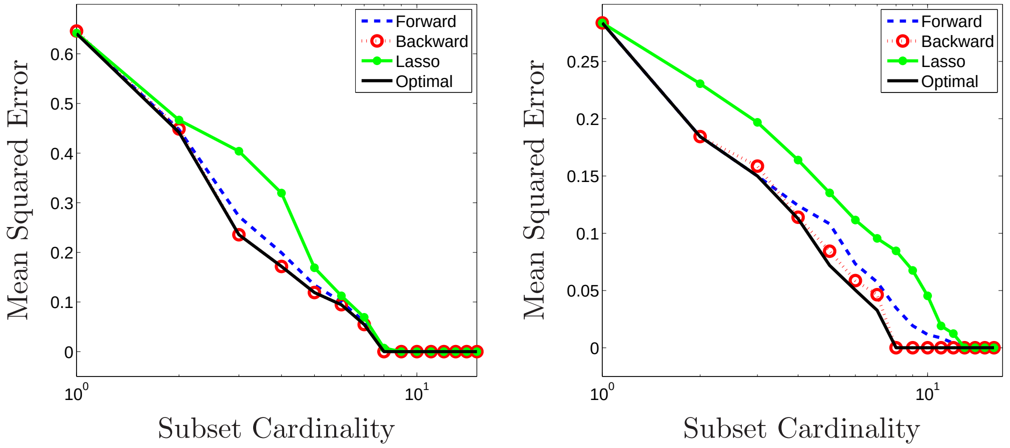

Figure 4: Greedy algorithm and Lasso. We plot the average mean squared error versus cardinality, over 100 replications, using forward (dotted line) and backward (circles) selection, the Lasso (large dots) and exhaustive search (solid line). *Left:* Lasso consistency condition satisfied. *Right:* consistency condition not satisfied.

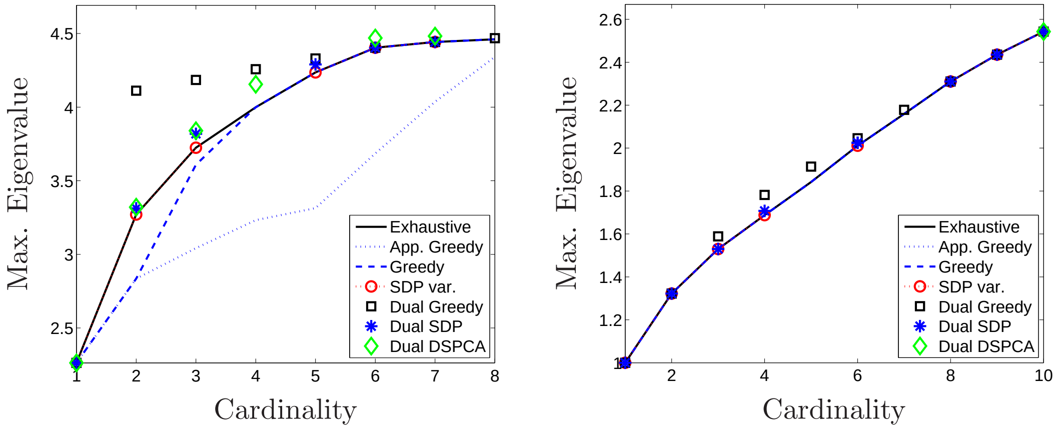

Figure 5: Upper and lower bound on sparse maximum eigenvalues. We plot the maximum sparse eigenvalue versus cardinality, obtained using exhaustive search (solid line), the approximate greedy (dotted line) and fully greedy (dashed line) algorithms. We also plot the upper bounds obtained by minimizing the gap of a rank one solution (squares), by solving the semidefinite relaxation explicitly (stars) and by solving the DSPCA dual (diamonds). *Left:* On a matrix $F^T F$ with F Gaussian. *Right:* On a sparse rank one plus noise matrix.

of the biological examples that follow), while Gaussian random matrices are harder. Note however, that the duality gap between the semidefinite relaxations and the optimal solution is very small in both cases, while our bounds based on greedy solutions are not as good. This means that solving the relaxations in (9) and d'Aspremont et al. (2007b) could provide very tight upper bounds on sparse eigenvalues of random matrices. However, solving these semidefinite programs for very large values of n remains a significant challenge.

7.4 Biological Data

We run the algorithm of Section 3.3 on two gene expression data sets, one on Colon cancer from Alon et al. (1999), the other on Lymphoma from Alizadeh et al. (2000). We plot the variance versus cardinality tradeoff curve in figure 6, together with the dual upper bounds from Section 5.3. In both cases, we consider the 500 genes with largest variance. Note that for many cardinalities, we have optimal or very close to optimal solutions. In Table 1, we also compare the 20 most important genes selected by the second sparse PCA factor on the colon cancer data set, with the top 10 genes selected by the RankGene software by Su et al. (2003). We observe that 6 genes (out of an original 4027 genes) were both in the top 20 sparse PCA genes and in the top 10 Rankgene genes.

8. Conclusion

We have presented a new convex relaxation of sparse principal component analysis, and derived tractable sufficient conditions for optimality. These conditions go together with

Rank	Rankgene	GAN	Description
3	8.6	J02854	Myosin regul.
6	18.9	T92451	Tropomyosin
7	31.5	T60155	Actin
8	25.1	H43887	Complement fact. D prec.
10	2.1	M63391	Human desmin
12	32.3	T47377	S-100P Prot.

Table 1: 6 genes (out of 4027) that were both in the top 20 sparse PCA genes and in the top 10 Rankgene genes.

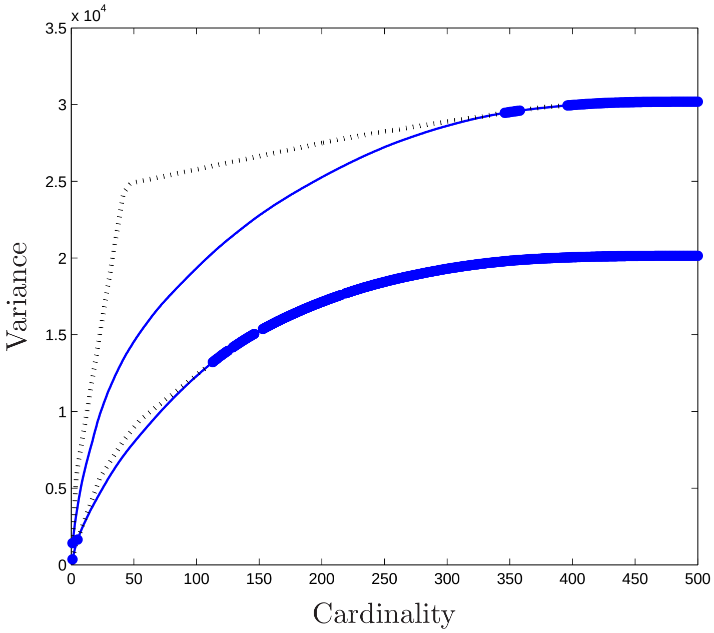

Figure 6: Variance (solid lines) versus cardinality tradeoff curve for two gene expression data sets, lymphoma (top) and colon cancer (bottom), together with dual upper bounds from Section 5.3 (dotted lines). Optimal points (for which the relative duality gap is less than 10^{-4}) are in bold.

efficient greedy algorithms that provide candidate solutions, many of which turn out to be optimal in practice. The resulting upper bounds also have direct applications to problems such as sparse recovery, subset selection or LASSO variable selection. Note that we extensively use this convex relaxation to test optimality and provide bounds on sparse extremal eigenvalues, but we almost never attempt to solve it numerically (except in some of the numerical experiments), which would provide optimal bounds. Having n matrix variables of dimension n, the problem is of course extremely large and finding numerical algorithms to directly optimize these relaxation bounds would be an important extension of this work.

Appendix A. Expansion of eigenvalues

In this appendix, we consider various results on expansions of eigenvalues we use in order to derive sufficient conditions. The following proposition derives a second order expansion of the set of eigenvectors corresponding to a single eigenvalue.

Proposition 9 *Let $N \in \mathbf{S}^n$. Let λ_0 be an eigenvalue of N, with multiplicity r and eigenvectors $U \in \mathbf{R}^{n\times r}$ (such that $U^TU = \mathbf{I}$). Let Δ be a matrix in $\mathbf{S}^n$. If $\|\Delta\|_F$ is small enough, the matrix $N + \Delta$ has exactly r (possibly equal) eigenvalues around λ_0 and if we denote by $(N + \Delta)_{\lambda_0}$ the projection of the matrix $N + \Delta$ onto that eigensubspace, we have:*

$$\begin{aligned}(N+\Delta)_{\lambda_0} = {} & \lambda_0 UU^T + UU^T\Delta UU^T + \lambda_0 UU^T\Delta(\lambda_0\mathbf{I}-N)^\dagger + \lambda_0(\lambda_0\mathbf{I}-N)^\dagger\Delta UU^T \\ & + UU^T\Delta UU^T\Delta(\lambda_0\mathbf{I}-N)^\dagger + (\lambda_0\mathbf{I}-N)^\dagger\Delta UU^T\Delta UU^T + UU^T\Delta(\lambda_0\mathbf{I}-N)^\dagger UU^T \\ & + \lambda_0 UU^T\Delta(\lambda_0\mathbf{I}-N)^\dagger\Delta(\lambda_0\mathbf{I}-N)^\dagger + \lambda_0(\lambda_0\mathbf{I}-N)^\dagger\Delta(\lambda_0\mathbf{I}-N)^\dagger\Delta UU^T \\ & + \lambda_0(\lambda_0\mathbf{I}-M)^\dagger\Delta UU^T\Delta(\lambda_0\mathbf{I}-M)^\dagger + O(\|\Delta\|_F^3)\end{aligned}$$

which implies the following expansion for the sum of the r eigenvalues in the neigborhood of λ_0:

$$\begin{aligned}\mathbf{Tr}(N+\Delta)_{\lambda_0} = {} & r\lambda_0 + \mathbf{Tr}\, U^T\Delta U + \mathbf{Tr}\, U^T\Delta(\lambda_0\mathbf{I}-N)^\dagger\Delta U \\ & + \lambda_0\,\mathbf{Tr}(\lambda_0\mathbf{I}-N)^\dagger\Delta UU^T\Delta(\lambda_0\mathbf{I}-N)^\dagger + O(\|\Delta\|_F^3).\end{aligned}$$

Proof We use the Cauchy residue formulation of projections on principal subspaces (Kato, 1966): given a symmetric matrix N, and a simple closed curve $\mathcal{C}$ in the complex plane that does not go through any of the eigenvalues of N, then

$$\Pi_{\mathcal{C}}(N) = \frac{1}{2i\pi}\oint_{\mathcal{C}} \frac{d\lambda}{\lambda\mathbf{I}-N}$$

is equal to the orthogonal projection onto the orthogonal sum of all eigensubspaces of N associated with eigenvalues in the interior of $\mathcal{C}$ (Kato, 1966). This is easily seen by writing down the eigenvalue decomposition $N = \sum_{i=1}^n \lambda_i u_i u_i^T$, and the Cauchy residue formula ($\frac{1}{2i\pi}\oint_{\mathcal{C}}\frac{d\lambda}{\lambda-\lambda_i} = 1$ if λ_i is in the interior $\mathrm{int}(\mathcal{C})$ of $\mathcal{C}$ and 0 otherwise), and:

$$\frac{1}{2i\pi}\oint_{\mathcal{C}}\frac{d\lambda}{\lambda\mathbf{I}-N} = \sum_{i=1}^n u_i u_i^T \times \frac{1}{2i\pi}\oint_{\mathcal{C}}\frac{d\lambda}{\lambda-\lambda_i} = \sum_{i,\ \lambda_i\in\mathrm{int}(\mathcal{C})} u_i u_i^T.$$

See Rudin (1987) for an introduction to complex analysis and Cauchy residue formula. Moreover, we can obtain the restriction of N onto a specific sum of eigensubspaces as:

$$N\Pi_{\mathcal{C}}(N) = \frac{1}{2i\pi}\oint_{\mathcal{C}}\frac{N d\lambda}{\lambda\mathbf{I}-N} = \frac{1}{2i\pi}\oint_{\mathcal{C}}\frac{\lambda d\lambda}{\lambda\mathbf{I}-N}.$$

From there we can easily compute expansions around a given N by using expansions of $(\lambda \mathbf{I} - N)^{-1}$. The proposition follows by considering a circle around λ_0 that is small enough to exclude other eigenvalues of N, and applying several times the Cauchy residue formula. ■

We can now apply the previous proposition to our particular case:

Lemma 10 *For any $a \in \mathbf{R}^n$, $\rho > 0$ and $B = aa^T - \rho\mathbf{I}$, we consider the function $F : X \mapsto \mathbf{Tr}(X^{1/2}BX^{1/2})_+$ from $\mathbf{S}^n_+$ to $\mathbf{R}$. let $x \in \mathbf{R}^n$ such that $\|x\| = 1$. Let $Y \succeq 0$. If $x^TBx > 0$, then*

$$F((1-t)xx^T + tY) = x^TBx + \frac{t}{x^TBx} \mathbf{Tr}\, Bxx^TB(Y - xx^T) + O(t^{3/2}),$$

while if $x^TBx < 0$, then

$$F((1-t)xx^T + tY) = \mathbf{Tr} \left(Y^{1/2} \left(B - \frac{Bxx^TB}{x^TBx} \right) Y^{1/2} \right)_+ + O(t^{3/2}).$$

Proof We consider $X(t) = (1-t)xx^T + tY$. We have $X(t) = U(t)U(t)^T$ with $U(t) = \begin{pmatrix} (1-t)^{1/2}x \\ t^{1/2}Y^{1/2} \end{pmatrix}$, which implies that the non zero eigenvalues of $X(t)^{1/2}BX(t)^{1/2}$ are the same as the non zero eigenvalues of $U(t)^TBU(t)$. We thus have

$$F(X(t)) = \mathbf{Tr}(M(t))_+,$$

with

$$\begin{aligned} M(t) &= \begin{pmatrix} (1-t)x^TBx & t^{1/2}(1-t)^{1/2}x^TBY^{1/2} \\ t^{1/2}(1-t)^{1/2}y^TBx & tY^{1/2}BY^{1/2} \end{pmatrix} \\ &= \begin{pmatrix} x^TBx & 0 \\ 0 & 0 \end{pmatrix} + t^{1/2} \begin{pmatrix} 0 & x^TBY^{1/2} \\ Y^{1/2}Bx & 0 \end{pmatrix} + t \begin{pmatrix} -x^TBx & 0 \\ 0 & Y^{1/2}BY^{1/2} \end{pmatrix} + O(t^{3/2}) \\ &= M(0) + t^{1/2}\Delta_1 + t\Delta_2 + O(t^{3/2}). \end{aligned}$$

The matrix $M(0)$ has a single (and simple) non zero eigenvalue which is equal to $\lambda_0 = x^TBx$ with eigenvector $U = (1,0)^T$. The only other eigenvalue of $M(0)$ is zero, with multiplicity n. Proposition 9 can be applied to the two eigenvalues of $M(0)$: there is one eigenvalue of $M(t)$ around x^TBx, while the n remaining ones are around zero. The eigenvalue close to λ_0 is equal to:

$$\begin{aligned} \mathbf{Tr}(M(t))_{\lambda_0} &= t\, \mathbf{Tr}\, U^\top \Delta_2 U + \lambda_0 + t\, \mathbf{Tr}\, U^T \Delta_1 (\lambda_0 \mathbf{I} - M(0))^\dagger \Delta_1 U \\ &\quad + \lambda_0\, \mathbf{Tr}(\lambda_0 \mathbf{I} - M(0))^\dagger \Delta_1 UU^T \Delta_1 (\lambda_0 \mathbf{I} - M(0))^\dagger + O(t^{3/2}) \\ &= x^TBx + \frac{t}{x^TBx} \mathbf{Tr}\, Bxx^TB(Y - xx^T) + O(t^{3/2}). \end{aligned}$$

For the remaining eigenvalues, we get that the projected matrix on the eigensubspace of $M(t)$ associated with eigenvalues around zero is equal to

$$\begin{aligned} (M(t))_0 &= t(\mathbf{I} - UU^T)\Delta_2(\mathbf{I} - UU^T) + t(\mathbf{I} - UU^T)\Delta_1(-M(0))^\dagger(\mathbf{I} - UU^T) + O(t^{3/2}) \\ &= \begin{pmatrix} 0 & 0 \\ 0 & tY^{1/2}(B - \frac{Bxx^TB}{x^TBx})Y^{1/2} \end{pmatrix}, \end{aligned}$$

which leads to a positive part equal to $t_+ \,\mathbf{Tr}\left[Y^{1/2}(B - \frac{Bxx^TB}{x^TBx})Y^{1/2}\right]_+$. When $x^TBx > 0$, then the matrix is negative definite (because $B = aa^T - \rho\mathbf{I}$), and thus the positive part is zero. By summing the two contributions, we obtain the desired result. ■

References

A. Alizadeh, M. Eisen, R. Davis, C. Ma, I. Lossos, and A. Rosenwald. Distinct types of diffuse large b-cell lymphoma identified by gene expression profiling. *Nature*, 403:503–511, 2000.

A. Alon, N. Barkai, D. A. Notterman, K. Gish, S. Ybarra, D. Mack, and A. J. Levine. Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *Cell Biology*, 96:6745–6750, 1999.

A. Ben-Tal and A. Nemirovski. On tractable approximations of uncertain linear matrix inequalities affected by interval uncertainty. *SIAM Journal on Optimization*, 12(3):811–833, 2002.

S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

J. Cadima and I. T. Jolliffe. Loadings and correlations in the interpretation of principal components. *Journal of Applied Statistics*, 22:203–214, 1995.

E. J. Candès and T. Tao. Decoding by linear programming. *Information Theory, IEEE Transactions on*, 51(12):4203–4215, 2005.

C. Couvreur and Y. Bresler. On the optimality of the backward greedy algorithm for the subset selection problem. *SIAM J. Matrix Anal. Appl.*, 21(3):797–808, 2000.

A. d'Aspremont, F. R. Bach, and L. El Ghaoui. Full regularization path for sparse principal component analysis. In *Proceedings of the Twenty-fourth International Conference on Machine Learning (ICML)*, 2007a.

A. d'Aspremont, L. El Ghaoui, M.I. Jordan, and G. R. G. Lanckriet. A direct formulation for sparse PCA using semidefinite programming. *SIAM Review*, 49(3):434–448, 2007b.

D. L. Donoho and J. Tanner. Sparse nonnegative solutions of underdetermined linear equations by linear programming. *Proceedings of the National Academy of Sciences*, 102(27): 9446–9451, 2005.

R.A. Horn and C.R. Johnson. *Matrix Analysis*. Cambridge University Press, 1985.

I. T. Jolliffe. Rotation of principal components: choice of normalization constraints. *Journal of Applied Statistics*, 22:29–35, 1995.

I. T. Jolliffe, N.T. Trendafilov, and M. Uddin. A modified principal component technique based on the LASSO. *Journal of Computational and Graphical Statistics*, 12:531–547, 2003.

T. Kato. *Perturbation Theory for Linear Operators.* Springer-Verlag, 1966.

N. Meinshausen and B. Yu. Lasso-type recovery of sparse representations for highdimensional data. Technical report, Technical Report, Statistics Department, UC Berkeley, 2006, 2006.

B. Moghaddam, Y. Weiss, and S. Avidan. Spectral bounds for sparse PCA: Exact and greedy algorithms. *Advances in Neural Information Processing Systems*, 18, 2006a.

B. Moghaddam, Y. Weiss, and S. Avidan. Generalized spectral bounds for sparse LDA. In *Proc. ICML*, 2006b.

B. K. Natarajan. Sparse approximate solutions to linear systems. *SIAM J. Comput.*, 24(2): 227–234, 1995.

W. Rudin. *Real and complex analysis, Third edition.* McGraw-Hill, Inc., New York, NY, USA, 1987. ISBN 0070542341.

B.K. Sriperumbudur, D.A. Torres, and G.R.G. Lanckriet. Sparse eigen methods by DC programming. *Proceedings of the 24th international conference on Machine learning*, pages 831–838, 2007.

Y. Su, T. M. Murali, V. Pavlovic, M. Schaffer, and S. Kasif. Rankgene: identification of diagnostic genes based on expression data. *Bioinformatics*, 19:1578–1579, 2003.

R. Tibshirani. Regression shrinkage and selection via the LASSO. *Journal of the Royal statistical society, series B*, 58(1):267–288, 1996.

M. Yuan and Y. Lin. On the non-negative garrotte estimator. *Journal of The Royal Statistical Society Series B*, 69(2):143–161, 2007.

Z. Zhang, H. Zha, and H. Simon. Low rank approximations with sparse factors I: basic algorithms and error analysis. *SIAM journal on matrix analysis and its applications*, 23 (3):706–727, 2002.

P. Zhao and B. Yu. On model selection consistency of lasso. *Journal of Machine Learning Research*, 7:2541–2563, 2006.

H. Zou, T. Hastie, and R. Tibshirani. Sparse Principal Component Analysis. *Journal of Computational & Graphical Statistics*, 15(2):265–286, 2006.