Solving Large-Scale Sparse PCA to Certifiable (Near) Optimality

Dimitris Bertsimas · Ryan Cory-Wright · Jean Pauphilet

Abstract Sparse principal component analysis (PCA) is a popular dimensionality reduction technique for obtaining principal components which are linear combinations of a small subset of the original features. Existing approaches cannot supply certifiably optimal principal components with more than $p = 100s$ covariates. By reformulating sparse PCA as a convex mixed-integer semidefinite optimization problem, we design a cutting-plane method which solves the problem to certifiable optimality at the scale of selecting $k = 10$ covariates from $p = 300$ variables, and provides small bound gaps at a larger scale. We also propose two convex relaxations and randomized rounding schemes that provide certifiably near-exact solutions within minutes for $p = 100$s or hours for $p = 1,000$s. Using real-world financial and medical datasets, we illustrate our approach's ability to derive interpretable principal components tractably at scale.

Keywords Sparse principal component analysis · Mixed-integer optimization · Semidefinite optimization · Sparse eigenvalues

Mathematics Subject Classification (2010) 62H25 · 90C11 · 90C22

1 Introduction

In the era of big data, interpretable methods for compressing a high-dimensional dataset into a lower dimensional set which shares the same essential characteristics are imperative. Since the work of Hotelling [37], principal component analysis (PCA) has been one of the most popular approaches for completing this task. Formally, given centered data $\boldsymbol{A} \in \mathbb{R}^{n\times p}$ and its normalized empirical covariance matrix $\boldsymbol{\Sigma} := \frac{\boldsymbol{A}\boldsymbol{A}^\top}{n-1} \in \mathbb{R}^{p\times p}$, PCA selects one or more leading eigenvectors of $\boldsymbol{\Sigma}$ and subsequently projects $\boldsymbol{A}$ onto these eigenvectors. This can be achieved in $O(p^3)$ time by taking a singular value decomposition $\boldsymbol{\Sigma} = \boldsymbol{S}\boldsymbol{\Lambda}\boldsymbol{S}^\top$.

A common criticism of PCA is that the columns of $\boldsymbol{S}$ are not interpretable, since each eigenvector is a linear combination of all p original features. This causes difficulties because:

D. Bertsimas
Sloan School of Management, Massachusetts Institute of Technology, Cambridge, MA 02139
ORCID: 0000-0002-1985-1003 E-mail: dbertsim@mit.edu

R. Cory-Wright
Operations Research Center, Massachusetts Institute of Technology, Cambridge, MA 02139
ORCID: 0000-0002-4485-0619 E-mail: ryancw@mit.edu

J. Pauphilet
Operations Research Center, Massachusetts Institute of Technology, Cambridge, MA 02139
ORCID: 0000-0001-6352-0984 E-mail: jpauph@mit.edu

- In medical diagnostic applications such as cancer detection, downstream decisions taken using principal component analysis need to be interpretable.
- In scientific applications such as protein folding, each original co-ordinate axis has a physical interpretation, and the reduced set of co-ordinate axes should also posses this property.
- In financial applications such as investing capital across a set of index funds, each non-zero entry in each eigenvector used to reduce the feature space incurs a transaction cost.
- If $p \gg n$, PCA suffers from a curse of dimensionality and becomes physically meaningless [2].

One common method for obtaining interpretable principal components is to stipulate that they are sparse, i.e., maximize variance while containing at most k non-zero entries. This approach leads to the following non-convex mixed-integer quadratically constrained problem [see 25]:

$$\max_{\boldsymbol{x}\in\mathbb{R}^p} \ \boldsymbol{x}^\top \boldsymbol{\Sigma} \boldsymbol{x} \text{ s.t. } \boldsymbol{x}^\top \boldsymbol{x} = 1, \ ||\boldsymbol{x}||_0 \leq k, \tag{1}$$

where the sparsity constraint $||\boldsymbol{x}||_0 \leq k$ forces variance to be explained in a compelling fashion.

1.1 Background and Literature Review

Owing to sparse PCA's fundamental importance in a variety of applications including best subset selection [26], natural language processing [58], compressed sensing [20], and clustering [46], three distinct classes of methods for addressing Problem (1) have arisen. Namely, (a) heuristic methods which obtain high-quality sparse PCs in an efficient fashion but do not supply guarantees on the quality of the solution, (b) convex relaxations which obtain certifiably near-optimal solutions by solving a convex relaxation and rounding, and (c) exact methods which obtain certifiably optimal solutions, albeit possibly in exponential time in the worst case.

Heuristic Approaches: The importance of identifying a small number of interpretable principal components has been well-documented in the literature since the work of Hotelling [37] [see also 38], giving rise to many distinct heuristic approaches for obtaining high-quality solutions to Problem (1). Two interesting such approaches are to rotate dense principal components to promote sparsity [42, 52, 40], or apply an l_1 penalty term as a convex surrogate to the cardinality constraint [39, 59]. Unfortunately, the former approach does not provide performance guarantees, while the latter approach leads to a non-convex optimization problem.

More recently, motivated by the need to rapidly obtain high-quality sparse principal components at scale, a wide variety of first-order heuristic methods have emerged. The first such *modern* heuristic was developed by Journée et al. [41], and involves combining the power method with thresholding and re-normalization steps. By pursuing similar ideas, several related methods have since been developed [see 35, 53, 47, 56, among others]. Unfortunately, while these methods are often very effective in practice, they sometimes badly fail to recover an optimal sparse principal component, and a practitioner using a heuristic method typically has no way of knowing when this has occurred. Indeed, Berk and Bertsimas [8] recently compared 7 heuristic methods, including most of those reviewed here, on 14 instances of sparse PCA, and found that none of the heuristic methods successfully recovered an optimal solution in all 14 cases.

Convex Relaxations: Motivated by the shortcomings of heuristic approaches on high-dimensional datasets, and the successful application of semi-definite optimization in obtaining high-quality approximation bounds in other applications [see 34, 55], a variety of convex relaxations have been proposed for sparse PCA. The first such convex relaxation was proposed by d'Aspremont et al. [25], who reformulated sparse PCA as the rank-constrained mixed-integer semidefinite optimization problem (MISDO):

$$\max_{\boldsymbol{X} \succeq \mathbf{0}} \quad \langle \boldsymbol{\Sigma}, \boldsymbol{X} \rangle \text{ s.t. } \operatorname{tr}(\boldsymbol{X}) = 1,\ \|\boldsymbol{X}\|_0 \leq k^2,\ \operatorname{Rank}(\boldsymbol{X}) = 1, \tag{2}$$

where $\boldsymbol{X}$ models the outer product $\boldsymbol{x}\boldsymbol{x}^\top$. Problem (2) is as hard to solve as (1). Consequently, d'Aspremont et al. [25] relaxed both the cardinality and rank constraints and instead solved

$$\max_{\boldsymbol{X} \succeq \mathbf{0}} \quad \langle \boldsymbol{\Sigma}, \boldsymbol{X} \rangle \text{ s.t. } \operatorname{tr}(\boldsymbol{X}) = 1,\ \|\boldsymbol{X}\|_1 \leq k, \tag{3}$$

which supplies a valid upper bound on Problem (1)'s objective.

The semidefinite approach has since been refined in a number of follow-up works. Among others, d'Aspremont et al. [26], building upon the work of Ben-Tal and Nemirovski [7], proposed a different semidefinite relaxation which supplies a sufficient condition for optimality via the primal-dual KKT conditions, and d'Aspremont et al. [27] analyzed the quality of the semidefinite relaxation in order to obtain high-quality approximation bounds. A common theme in these approaches is that they require solving large-scale semidefinite optimization problems. This presents difficulties for practitioners because state-of-the-art implementations of interior point methods such as `Mosek` require $O(p^6)$ memory to solve Problem (3), and therefore currently cannot solve instances of Problem (2) with $p \geq 300$ [see 9, for a recent comparison].

More recently, by building on the work of Kim and Kojima [43], Ahmadi and Majumdar [1], Bertsimas and Cory-Wright [9] introduced a second-order cone relaxation of (2) which scales to $p = 1000s$, and matches the semidefinite bound after imposing a small number of cuts. Moreover, it typically supplies bound gaps of less than 5%. However, it does not supply an *exact* certificate of optimality, which is often desirable. Indeed, in financial and medical applications, a 0.1% improvement in solution quality often saves millions of dollars or tens of lives.

A fundamental drawback of existing convex relaxation techniques is that they are not coupled with rounding schemes for obtaining high-quality feasible solutions. This is problematic, because optimizers are typically interested in obtaining high-quality solutions, rather than certificates. In this paper, we take a step in this direction, by deriving new convex relaxations that naturally give rise to greedy and random rounding schemes. The fundamental point of difference between our relaxations and existing relaxations is that we derive our relaxations by rewriting sparse PCA as a MISDO and dropping an integrality constraint, rather than using more ad-hoc techniques.

Exact Methods: Motivated by the successful application of mixed-integer optimization for solving statistical learning problems such as best subset selection [10] and sparse classification [12], several exact methods for solving sparse PCA to certifiable optimality have been proposed. The first branch-and-bound algorithm for solving Problem (1) was proposed by Moghaddam et al. [48], by applying norm equivalence relations to obtain valid bounds. However, Moghaddam et al. [48]

did not couple their approach with high-quality initial solutions and tractable bounds to prune partial solutions. Consequently, they could not scale their approach beyond $p = 40$.

A more sophisticated branch-and-bound scheme was recently proposed by Berk and Bertsimas [8], which couples tighter Gershgorin Circle Theorem bounds [36, Chapter 6] with a fast heuristic due to [56] to solve problems up to $p = 250$. However, their method cannot scale beyond $p = 100$s, because the bounds obtained are too weak to avoid enumerating a sizeable portion of the tree.

Very recently, the authors developed a framework for reformulating convex mixed-integer optimization problems with logical constraints [see 13], and demonstrated that this framework allows a number of problems of practical relevance to be solved to certifiably optimality via a cutting-plane method. In this paper, we build upon this prior work by reformulating Problem (1) as a *convex* mixed-integer semidefinite optimization problem, and leverage this reformulation to design a cutting-plane method which solves sparse PCA to certifiable optimality. A key feature of our approach is that we need not solve any semidefinite subproblems. Rather, we use *ideas* from SDO to design a semidefinite-free approach which uses simple linear algebra techniques.

1.2 Contributions and Structure

The key contributions of the paper are twofold. First, we reformulate sparse PCA exactly as a mixed-integer semidefinite optimization problem; a reformulation which is, to the best of our knowledge, novel. Second, we propose a suite of techniques for solving non-convex mixed-integer quadratic optimization problems, such as sparse PCA, to certifiable optimality or near-optimality at a larger scale than existing state-of-the-art methods. The structure of the paper is as follows:

- In Section 2, we reformulate Problem (1) as a mixed-integer SDO, and propose a cutting-plane method which solves it to certifiable optimality. Moreover, we show that we need not solve any SDOs in our algorithmic strategy, by deriving a semidefinite free subproblem strategy.
- In Section 3, we analyze the semidefinite reformulation's convex relaxation, and introduce a greedy rounding scheme which supplies provably high-quality solutions to Problem (1) in polynomial time. We also propose a tighter doubly non-negative relaxation, and investigate its dual side, a Goemans-Williamson rounding scheme [34].
- In Section 4, we apply the cutting-plane and random rounding methods method to derive optimal and near optimal sparse principal components for problems in the UCI dataset. We also compare our method's performance against the method of Berk and Bertsimas [8], and find that our exact cutting-plane method performs comparably, while our relax+round approach successfully scales to problems an order of magnitude larger. A key feature of our numerical success is that we sidestep the computational difficulties in solving SDOs at scale by proposing semidefinite-free methods for solving the convex relaxations, i.e., solving second-order cone rather than semidefinite relaxations.

Notation: We let nonbold face characters such as b denote scalars, lowercase bold faced characters such as $\boldsymbol{x}$ denote vectors, uppercase bold faced characters such as $\boldsymbol{X}$ denote matrices, and calligraphic uppercase characters such as $\mathcal{Z}$ denote sets. We let $[p]$ denote the set of running indices $\{1, ..., p\}$. We let $\mathbf{e}$ denote a vector of all 1's, $\mathbf{0}$ denote a vector of all 0's, and $\mathbb{I}$ denote the identity matrix, with dimension implied by the context.

We also use an assortment of matrix operators. We let $\langle \cdot, \cdot \rangle$ denote the Euclidean inner product between two matrices, $\| \cdot \|_F$ denote the Frobenius norm of a matrix, $\| \cdot \|_\sigma$ denote the spectral norm of a matrix, $\| \cdot \|_*$ denote the nuclear norm of a matrix, $\boldsymbol{X}^\dagger$ denote the Moore-Penrose psuedoinverse of a matrix $\boldsymbol{X}$ and S^p_+ denote the $p \times p$ positive semidefinite cone; see Horn and Johnson [36] for a general theory of matrix operators.

2 An Exact Mixed-Integer Semidefinite Reformulation

In this section, we reformulate Problem (1) as a convex mixed-integer semidefinite convex optimization problem. In formulation (2), we introduce binary variables z_i to model whether $X_{i,j}$ is non-zero, via the logical constraint $X_{i,j} = 0$ if $z_i = 0$; note that we need not require that $X_{i,j} = 0$ if $z_j = 0$, since $\boldsymbol{X}$ is a symmetric matrix. By enforcing the logical constraint via $-M_{i,j} z_i \leq X_{i,j} \leq M_{i,j} z_i$ for sufficiently large $M_{i,j} > 0$, Problem (2) becomes

$$\max_{\boldsymbol{z} \in \{0,1\}^p : \boldsymbol{e}^\top \boldsymbol{z} \leq k} \; \max_{\boldsymbol{X} \in S^p_+} \; \langle \boldsymbol{\Sigma}, \boldsymbol{X} \rangle$$
$$\text{s.t.} \quad \text{tr}(\boldsymbol{X}) = 1, \; -M_{i,j} z_i \leq X_{i,j} \leq M_{i,j} z_i, \; \forall i, j \in [p], \; \text{Rank}(\boldsymbol{X}) = 1.$$

To obtain a MISDO reformulation, we omit the rank constraint. In general, omitting a rank constraint generates a relaxation and induces some loss of optimality. However, we can actually omit the constraint without loss of optimality! Indeed, the objective is convex and therefore some rank-one extreme matrices $\boldsymbol{X}$ is optimal. We formalize this observation in the following theorem; note that a similar result (although in the context of computing Restricted Isometry constants and with a different proof) exists [32, Theorem 3]:

Theorem 1 *Problem* (1) *attains the same optimal objective value as the problem:*

$$\begin{aligned} \max_{\boldsymbol{z} \in \{0,1\}^p : \boldsymbol{e}^\top \boldsymbol{z} \leq k} \; \max_{\boldsymbol{X} \in S^p_+} \; & \langle \boldsymbol{\Sigma}, \boldsymbol{X} \rangle & & \\ s.t. \quad & \text{tr}(\boldsymbol{X}) = 1 & & [\lambda], \\ & X_{i,j} \leq M_{i,j} z_i & & [\alpha^+_{i,j}], \; \forall i, j \in [p], \\ & -X_{i,j} \leq M_{i,j} z_i & & [\alpha^-_{i,j}], \; \forall i, j \in [p], \end{aligned} \tag{4}$$

where $M_{i,i} = 1$, $M_{i,j} = \frac{1}{2}$ *if* $j \neq i$ *and we associate a dual multiplier with each constraint in square brackets.*

Proof It suffices to demonstrate that for any feasible solution to (1) we can construct a feasible solution to (4) with an equal or greater payoff, and vice versa.

- Let $\boldsymbol{x} \in \mathbb{R}^p$ be a feasible solution to (1). Then, it is immediate that $(\boldsymbol{X} := \boldsymbol{x}\boldsymbol{x}^\top, \boldsymbol{z})$ forms a feasible solution to (4) with an equal cost, where $z_i = 1$ if $|x_i| > 0$ and $z_i = 0$ otherwise.
- Let $(\boldsymbol{X}, \boldsymbol{z})$ be a feasible solution to Problem (4), and let $\boldsymbol{X} = \sum_{i=1}^p \sigma_i \boldsymbol{x}_i \boldsymbol{x}_i^\top$ be a Cholesky decomposition of $\boldsymbol{X}$, where $\boldsymbol{e}^\top \boldsymbol{\sigma} = 1, \boldsymbol{\sigma} \geq \boldsymbol{0}$. Observe that $\|\boldsymbol{x}_i\|_0 \leq k, \forall i \in [p]$, since we can perform the Cholesky decomposition on the submatrix of $\boldsymbol{X}$ induced by $\boldsymbol{z}$, and "pad" out the remaining entries of each $\boldsymbol{x}_i$ with 0s to obtain the decomposition of $\boldsymbol{X}$. Therefore, let us set $\hat{\boldsymbol{x}} := \arg\max_i [\boldsymbol{x}_i^\top \boldsymbol{\Sigma} \boldsymbol{x}_i]$. Then, $\hat{\boldsymbol{x}}$ is a feasible solution to (1) with an equal or greater payoff.

Finally, we let $M_{i,i} = 1$, $M_{i,j} = \frac{1}{2}$ if $i \neq j$, as the 2×2 minors imply $X_{i,j}^2 \leq X_{i,i} X_{j,j} \leq \frac{1}{4}$ whenever $i \neq j$ [c.f. 32, Lemma 1]. □

Theorem 1 reformulates Problem (1) as a mixed-integer SDO. Therefore, we can solve Problem (4) using general branch-and-cut techniques for semidefinite optimization problems [see 33, 44, 23]. However, this approach is not scalable, as it comprises solving a large number of semidefinite subproblems and the community does not know how to efficiently warm-start IPMs for SDOs.

We now propose a saddle-point reformulation of Problem (4) which avoids the computational difficulty in solving a large number of SDOs by exploiting problem structure, as we will show in Section 2.2. Our reformulation allows us to propose a branch-and-cut method which solves each subproblem using linear algebra techniques.

We have the following result, essentially due to [13, Theorem 1]:

Theorem 2 *Problem* (4) *attains the same optimal value as the following problem:*

$$\max_{\boldsymbol{z} \in \{0,1\}^p:\ \boldsymbol{e}^\top \boldsymbol{z} \leq k} \quad f(\boldsymbol{z}) \tag{5}$$

$$\textit{where} \quad f(\boldsymbol{z}) := \min_{\lambda \in \mathbb{R}, \boldsymbol{\alpha} \in \mathbb{R}^{p \times p}} \quad \lambda + \sum_{i=1}^p z_i \left(|\alpha_{i,i}| + \frac{1}{2} \sum_{j=1, j \neq i}^p |\alpha_{i,j}| \right) \ \textit{s.t.}\ \lambda \mathbb{I} + \boldsymbol{\alpha} \succeq \boldsymbol{\Sigma} \tag{6}$$

Remark 1 The above theorem demonstrates that $f(\boldsymbol{z})$ is concave in $\boldsymbol{z}$, by rewriting it as the infimum of functions which are linear in $\boldsymbol{z}$ [16, Chapter 3.2.3].

Proof Let us rewrite the inner optimization problem as

$$\begin{aligned} f(\boldsymbol{z}) := \quad & \max_{\boldsymbol{X} \succeq \boldsymbol{0}} \langle \boldsymbol{\Sigma}, \boldsymbol{X} \rangle & \\ \text{s.t.} \quad & \operatorname{tr}(\boldsymbol{X}) = 1 & [\lambda], \\ & X_{i,j} \leq M_{i,j} z_i & [\alpha_{i,j}^+],\ \forall i, j \in [p], \\ & -X_{i,j} \leq M_{i,j} z_i & [\alpha_{i,j}^-],\ \forall i, j \in [p], \end{aligned}$$

The result then follows by invoking strong semidefinite duality, which holds as the optimization problem induced by $f(\boldsymbol{z})$ has non-empty relative interior. Observe that we replace $\boldsymbol{\alpha}^+ - \boldsymbol{\alpha}^-$ with $\boldsymbol{\alpha}$, $\boldsymbol{\alpha}^+ + \boldsymbol{\alpha}^-$ with the absolute value of $\boldsymbol{\alpha}$, and substitute $M_{i,i} = 1$, $M_{i,j} = \frac{1}{2}$ if $i \neq j$. □

2.1 A Cutting-Plane Method

Theorem 2 shows that evaluating $f(\hat{\boldsymbol{z}})$ yields the globally valid overestimator:

$$f(\boldsymbol{z}) \leq f(\hat{\boldsymbol{z}}) + \boldsymbol{g}_{\hat{\boldsymbol{z}}}^\top (\boldsymbol{z} - \hat{\boldsymbol{z}}),$$

where $\boldsymbol{g}_{\hat{\boldsymbol{z}}}$ is a supergradient of f at $\hat{\boldsymbol{z}}$, at no additional cost. In particular, we have

$$g_{\hat{\boldsymbol{z}},i} = \left(|\alpha_{i,i}^\star| + \frac{1}{2} \sum_{j=1, j \neq i}^p |\alpha_{i,j}^\star| \right),$$

where $\boldsymbol{\alpha}^\star$ is an optimal choice of $\boldsymbol{\alpha}$ for a fixed $\hat{\boldsymbol{z}}$. This observation leads to an efficient strategy for maximizing $f(\boldsymbol{z})$: iteratively maximizing and refining a piecewise linear upper estimator of $f(\boldsymbol{z})$. This strategy is called outer-approximation (OA), and was originally proposed by Duran and Grossmann [30]. OA works by iteratively constructing estimators of the following form at each t:

$$f^t(\boldsymbol{z}) = \min_{1\leq i\leq t} \left\{ f(\boldsymbol{z}_i) + \boldsymbol{g}_{\boldsymbol{z}_i}^\top (\boldsymbol{z} - \boldsymbol{z}_i) \right\}.$$

After constructing each overestimator, we maximize $f^t(\boldsymbol{z})$ over $\{0,1\}^p$ to obtain $\boldsymbol{z}_t$, and evaluate $f(\cdot)$ and its supergradient at $\boldsymbol{z}_t$. This procedure yields a non-increasing sequence of overestimators $\{f^t(\boldsymbol{z}_t)\}_{t=1}^T$ which converge to the optimal value of $f(\boldsymbol{z})$ within a finite number of iterations $T \leq \binom{p}{k}$, since $\{0,1\}^p$ is a finite set and OA never visits a point twice; see also [31, Theorem 2]. Additionally, we can avoid solving a different MILO at each OA iteration by integrating the entire algorithm within a single branch-and-bound tree, as proposed by [49, 6], using `lazy constraint callbacks`. Lazy constraint callbacks are now standard components of modern MILO solvers such as `Gurobi` or `CPLEX` and substantially speed-up OA. We formalize this procedure in Algorithm 1; note that $\partial f(\boldsymbol{z}_{t+1})$ denotes the set of supergradients of f at $\boldsymbol{z}_{t+1}$.

Algorithm 1 An outer-approximation method for Problem (1)

Require: Initial solution $\boldsymbol{z}_1$
  $t \leftarrow 1$
  repeat
    Compute $\boldsymbol{z}_{t+1}, \theta_{t+1}$ solution of

$$\max_{\boldsymbol{z}\in\{0,1\}^p : \boldsymbol{e}^\top \boldsymbol{z}\leq k, \theta} \theta \quad \text{s.t. } \theta \leq f(\boldsymbol{z}_i) + \boldsymbol{g}_{\boldsymbol{z}_i}^\top (\boldsymbol{z} - \boldsymbol{z}_i),\ \forall i \in [t],$$

    Compute $f(\boldsymbol{z}_{t+1})$ and $\boldsymbol{g}_{\boldsymbol{z}_{t+1}} \in \partial f(\boldsymbol{z}_{t+1})$
    $t \leftarrow t + 1$
  until $f(\boldsymbol{z}_t) - \theta_t \leq \varepsilon$
  return $\boldsymbol{z}_t$

2.2 A Computationally Efficient Subproblem Strategy

Our derivation and analysis of Algorithm 1 indicates that we can solve Problem (1) to certifiable optimality by solving a (potentially large) number of semidefinite subproblems. However, this is not a good idea in practice, because semidefinite optimization problems are expensive to solve. Therefore, we now derive a computationally efficient subproblem strategy which crucially does not require solving *any* semidefinite programs. Formally, we have the following result:

Theorem 3 *For any $\boldsymbol{z} \in \{0,1\}^p : \boldsymbol{e}^\top \boldsymbol{z} \leq k$, optimal dual variables in (5) are*

$$\lambda = \lambda_{\max}\left(\boldsymbol{\Sigma}_{1,1}\right),\ \boldsymbol{\alpha} = \begin{pmatrix} \boldsymbol{\alpha}_{1,1} & \boldsymbol{\alpha}_{1,2} \\ \boldsymbol{\alpha}_{1,2}^\top & \boldsymbol{\alpha}_{2,2} \end{pmatrix} = \begin{pmatrix} \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{\Sigma}_{2,2} - \lambda\mathbb{I} + \boldsymbol{\Sigma}_{1,2}^\top \left(\lambda\mathbb{I} - \boldsymbol{\Sigma}_{1,1}\right)^\dagger \boldsymbol{\Sigma}_{1,2} \end{pmatrix}, \tag{7}$$

where $\lambda_{\max}(\cdot)$ denotes the leading eigenvalue of a matrix, $\boldsymbol{\alpha} = \begin{pmatrix} \boldsymbol{\alpha}_{1,1} & \boldsymbol{\alpha}_{1,2} \\ \boldsymbol{\alpha}_{1,2}^\top & \boldsymbol{\alpha}_{2,2} \end{pmatrix}$ is a decomposition such that $\boldsymbol{\alpha}_{1,1}$ (resp. $\boldsymbol{\alpha}_{2,2}$) denotes the entries of $\boldsymbol{\alpha}$ where $z_i = z_j = 1$ ($z_i = z_j = 0$); $\boldsymbol{\Sigma}$ is similar.

Remark 2 By Theorem 3, we can solve a subproblem by computing the leading eigenvalue of $\boldsymbol{\Sigma}_{1,1}$ and solving a linear system. This justifies our claim that we need not solve any SDOs in our algorithmic strategy.

Proof We appeal to strong duality and complementary slackness. Observe that, for any $\boldsymbol{z} \in \{0,1\}^n$, $f(\boldsymbol{z})$ is the optimal value of a minimization problem over a closed convex compact set. Therefore, there exists some optimal primal solution $\boldsymbol{X}^\star$. Moreover, since the primal has non-empty relative interior, strong duality holds. Therefore, by complementary slackness, there must exist some dual-optimal solution $(\lambda, \boldsymbol{\alpha})$ which obeys complementarity with $\boldsymbol{X}^\star$. In particular, we have

$$(M_{i,j}z_i - X_{i,j})\alpha^+_{i,j} = 0, \quad \text{and} \quad (M_{i,j}z_i + X_{i,j})\alpha^-_{i,j} = 0, \quad \text{where } \boldsymbol{\alpha} = \boldsymbol{\alpha}^+ - \boldsymbol{\alpha}^-.$$

Moreover, $|X_{i,j}| \leq M_{i,j}$ is implied by $\text{tr}(\boldsymbol{X}) = 1, \boldsymbol{X} \succeq \boldsymbol{0}$. Therefore, by complementary slackness, we can take these constraints to be inactive when $z_i = 1$ without loss of generality, which implies that $\alpha^\star_{i,j} = 0$ if $z_i = 1$ in some dual-optimal solution. Moreover, we also have $\alpha^\star_{i,j} = 0$ if $z_j = 1$, since $\boldsymbol{\alpha}$ obeys the dual feasibility constraint $\lambda\mathbb{I} + \boldsymbol{\alpha} \succeq \boldsymbol{\Sigma}$, and therefore is itself symmetric.

Next, observe that, by strong duality, $\lambda = \lambda_{\max}(\boldsymbol{\Sigma}_{1,1})$ in this dual-optimal solution, since $\boldsymbol{\alpha}$ only takes non-zero values if $z_i = z_j = 0$ and therefore does not contribute to the objective.

To see that the result holds, observe that, by strong duality and complementary slackness, any dual feasible $(\lambda, \boldsymbol{\alpha})$ satisfying the above conditions is dual-optimal. Therefore, we need only find an $\boldsymbol{\alpha}_{2,2}$ such that

$$\begin{pmatrix} \lambda\mathbb{I} - \boldsymbol{\Sigma}_{1,1} & -\boldsymbol{\Sigma}_{1,2} \\ -\boldsymbol{\Sigma}_{2,1} & \lambda\mathbb{I} + \boldsymbol{\alpha}_{2,2} - \boldsymbol{\Sigma}_{2,2} \end{pmatrix} \succeq \boldsymbol{0}.$$

By the generalized Schur complement lemma [see 17, Equation 2.41], this is PSD if and only if

1. $\lambda\mathbb{I} - \boldsymbol{\Sigma}_{1,1} \succeq \boldsymbol{0}$,
2. $\left(\mathbb{I} - (\lambda\mathbb{I} - \boldsymbol{\Sigma}_{1,1})(\lambda\mathbb{I} - \boldsymbol{\Sigma}_{1,1})^\dagger\right)\boldsymbol{\Sigma}_{1,2} = \boldsymbol{0}$, and
3. $\lambda\mathbb{I} + \boldsymbol{\alpha}_{2,2} - \boldsymbol{\Sigma}_{2,2} \succeq \boldsymbol{\Sigma}_{1,2}^\top (\lambda\mathbb{I} - \boldsymbol{\Sigma}_{1,1})^\dagger \boldsymbol{\Sigma}_{1,2}$.

The first two conditions are independent of $\boldsymbol{\alpha}_{2,2}$, and therefore hold (otherwise strong duality and/or complementary slackness does not hold, a contradiction). Therefore, it suffices to pick $\boldsymbol{\alpha}_{2,2}$ in order that the third condition holds. We achieve this by setting $\boldsymbol{\alpha}_{2,2}$ so the PSD constraint in condition (3) holds with equality. □

2.3 Strengthening the Master Problem via the Gershgorin Circle Theorem

As Algorithm 1's rate of convergence rests heavily upon its implementation, we now propose a practical technique for accelerating Algorithm 1. Namely, inspired by [8]'s successful implementation of branch-and-bound using the Gershgorin Circle Theorem [see 36, Chapter 6] to generate bounds, we strengthen the master problem by imposing bounds from the circle theorem. Formally, we have the following result, which can be deduced from [36, Theorem 6.1.1]:

Theorem 4 *For any vector $\boldsymbol{z} \in \{0,1\}^p$ we have the following upper bound on $f(\boldsymbol{z})$*

$$f(\boldsymbol{z}) \leq \max_{i \in [p]: z_i = 1} \sum_{j \in [p]} z_j |\Sigma_{i,j}|.$$

Observe that this bound cannot be used to *directly* strengthen Algorithm 1's master problem, since the bound is not convex in $\boldsymbol{z}$. Nonetheless, it can be successfully applied if we (a) impose a big-M assumption on Problem (1)'s optimal objective and (b) introduce p additional binary variables $\boldsymbol{s} \in \{0,1\}^p : \boldsymbol{e}^\top \boldsymbol{s} = 1$. Formally, we impose the following valid inequalities in the master problem:

$$\exists \boldsymbol{s} \in \{0,1\}^p, t \in \mathbb{R} : \theta \leq t, t \geq \sum_{i \in [p]} z_i |\Sigma_{i,j}|, t \leq \sum_{i \in [p]} z_i |\Sigma_{i,j}| + M(1 - s_i), \boldsymbol{e}^\top \boldsymbol{s} = 1.$$

In the above inequalities, a valid M is given by any bound on the optimal objective. Since Theorem (4) supplies one such bound for any given $\boldsymbol{z}$, we compute our M by maximizing this bound over $\{\boldsymbol{z} \in \{0,1\}^p : \boldsymbol{e}^\top \boldsymbol{z} \leq k\}$.

We could further strengthen the master problem by imposing inequalities derived from *Brauer's Ovals of Cassini* [36, Theorem 6.4.7], which strictly improves upon Gershgorin's circle theorem. However, we do not consider this approach, as it introduces $O(p^2)$ additional binary variables in the master problem, which is less tractable, and Brauer's ovals require second-order cone, rather than linear, inequalities. Irregardless, an interesting extension would be to introduce the binary variables dynamically, via branch-and-cut-and-price [6].

To make clear the extent to which our numerical success depends upon the circle theorem, our results in Section 4 present implementations of Algorithm 1 both with and without this bound.

2.4 Frobenius Norm Regularization

In this section, we explore enforcing the logical relation $X_{i,j} = 0$ if $z_i = 0$ using Frobenius, rather than big-M regularization, as proposed in Bertsimas et al. [13]. By following their analysis, and also imposing the constraint $X_{i,j} = 0$ if $z_j = 0$ (unlike the big-M case, imposing both logical constraints is helpful for developing our subproblem strategy here) we obtain the following problem

$$\begin{aligned} \max_{\boldsymbol{z} \in \{0,1\}^p : \boldsymbol{e}^\top \boldsymbol{z} \leq k} \quad & \max_{\boldsymbol{X} \in S^p_+} \quad \langle \boldsymbol{\Sigma}, \boldsymbol{X} \rangle - \frac{1}{2\gamma} \|\boldsymbol{X}\|_F^2 \\ & \text{s.t.} \quad \operatorname{tr}(\boldsymbol{X}) = 1, X_{i,j} = 0 \text{ if } z_i = 0, \text{ or } z_j = 0 \; \forall i, j \in [p], \end{aligned} \tag{8}$$

which, by strong duality [13, Theorem 1], is equivalent to the saddle-point problem

$$\max_{\boldsymbol{z} \in \{0,1\}^p : \; \boldsymbol{e}^\top \boldsymbol{z} \leq k} \quad f(\boldsymbol{z}) \tag{9}$$

$$\text{where} \quad f(\boldsymbol{z}) := \min_{\lambda \in \mathbb{R}, \boldsymbol{\alpha} \in \mathbb{R}^{p \times p}, \boldsymbol{\beta} \in \mathbb{R}^{p \times p}} \quad \lambda + \frac{\gamma}{2} \sum_{i=1}^p z_i \sum_{j=1}^p (\alpha_{i,j} + \beta_{j,i})^2 \tag{10}$$

$$\text{s.t.} \quad \lambda \mathbb{I} + \boldsymbol{\alpha} + \boldsymbol{\beta} \succeq \boldsymbol{\Sigma},$$

and can be addressed by a cutting-plane method in much the same way.

It should be noted however that Problem (8) does not supply a rank-one matrix $\boldsymbol{X}^\star$, due to the ridge regularizer. Therefore, under Frobenius norm regularization, we first solve Problem (9) to obtain an optimal set of indices $\boldsymbol{z}$, and subsequently solve for an optimal $\boldsymbol{X}$ for this $\boldsymbol{z}$ in (5).

This perturbation strategy necessarily gives rise to some loss of optimality. However, this loss can be bounded. Indeed, the difference in optimal objectives between Problems (4) and (8) is at most $\frac{1}{2\gamma}\|\boldsymbol{X}^\star\|_F^2$, where $\boldsymbol{X}^\star$ is an optimal $\boldsymbol{X}$ in Problem (4). Moreover, since

$$\frac{1}{2\gamma}\|\boldsymbol{X}\|_F^2 = \frac{1}{2\gamma}\sum_i\sum_j X_{i,j}^2 \leq \frac{1}{2\gamma}\sum_i X_{i,i}\sum_j X_{j,j} = \frac{1}{2\gamma},$$

where the inequality follows from the 2×2 minors in $\boldsymbol{X}\succeq\mathbf{0}$ [c.f. 9, Proposition 3], the difference in objectives between Problems (4) and (8) is at most $\frac{1}{2\gamma}$ and becomes negligible as $\gamma\to\infty$.

We will make use of both types of regularization in our algorithmic results, and therefore derive an efficient subproblem strategy under ridge regularization as well:

Theorem 5 *For any fixed $\boldsymbol{z}\in\{0,1\}^p : \boldsymbol{e}^\top\boldsymbol{z}\leq k$, optimal dual variables in* (9) *are*

$$\lambda = \arg\min_{\lambda}\left\{\lambda + \frac{\gamma}{2}\|(\boldsymbol{\Sigma}_{1,1}-\lambda\mathbb{I})_+\|_F^2\right\}, \tag{11}$$
$$\boldsymbol{\alpha} = \begin{pmatrix}\boldsymbol{\alpha}_{1,1} & \boldsymbol{\alpha}_{1,2}\\ \boldsymbol{\alpha}_{2,1} & \boldsymbol{\alpha}_{2,2}\end{pmatrix} = \frac{1}{2}\begin{pmatrix}(\boldsymbol{\Sigma}_{1,1}-\lambda\mathbb{I})_+ & \mathbf{0}\\ 2\boldsymbol{\Sigma}_{2,1} & \boldsymbol{\Sigma}_{2,2}-\lambda\mathbb{I}\end{pmatrix}, \boldsymbol{\beta}=\boldsymbol{\alpha}^\top$$

where $(\boldsymbol{X})_+$ denotes the positive semidefinite component of $\boldsymbol{X}$, i.e., if $\boldsymbol{X}=\sum_{i=1}^p\sigma_i\boldsymbol{x}_i\boldsymbol{x}_i^\top$ is an eigendecomposition of $\boldsymbol{X}$ then $(\boldsymbol{X})_+=\sum_{i=1}^p\max(\sigma_i,0)\boldsymbol{x}_i\boldsymbol{x}_i^\top$, $\boldsymbol{\alpha}=\begin{pmatrix}\boldsymbol{\alpha}_{1,1} & \boldsymbol{\alpha}_{1,2}\\ \boldsymbol{\alpha}_{1,2}^\top & \boldsymbol{\alpha}_{2,2}\end{pmatrix}$ is a decomposition of $\boldsymbol{\alpha}$ such that $\boldsymbol{\alpha}_{1,1}$ (resp. $\boldsymbol{\alpha}_{2,2}$) denotes the entries of $\boldsymbol{\alpha}$ where $z_i=z_j=1$ (resp. $z_i=z_j=0$), and $\boldsymbol{\beta}$, $\boldsymbol{\Sigma}$ are similar.

Proof Observe that if $z_i=0$ then $\alpha_{i,j}$ does not contribute to the objective, while if $z_j=0$, $\beta_{i,j}$ does not contribute to the objective. Therefore, if $z_i=0$ and $z_j=1$ we can set $\beta_{i,j}=0$ and $\alpha_{i,j}$ to be any dual-feasible value, and vice versa. As a result, it suffices to solve $\boldsymbol{\alpha}_{1,1}$, $\boldsymbol{\beta}_{1,1},\lambda$, as we can subsequently pick the remaining components of $\boldsymbol{\alpha},\boldsymbol{\beta}$ in order that they are feasible and satisfy the aforementioned condition. Moreover, observe that we can set $\boldsymbol{\alpha}=\boldsymbol{\beta}^\top$ without loss of generality, since, in the derivation of the dual problem, $\boldsymbol{\alpha}$ is a matrix of dual variables associated with a constraint of the form $\boldsymbol{V}=\mathrm{Diag}(\boldsymbol{z})\boldsymbol{X}$, while $\boldsymbol{\beta}$ is a matrix of dual variables associated with a constraint of the form $\boldsymbol{V}=\boldsymbol{X}\mathrm{Diag}(\boldsymbol{z})$ [c.f. 13, Theorem 1].

Let us substitute $\hat{\boldsymbol{\alpha}}\leftarrow\boldsymbol{\alpha}_{1,1}+\boldsymbol{\beta}_{1,1}$ and consider the reduced inner dual problem

$$\min_{\hat{\boldsymbol{\alpha}},\lambda}\quad \lambda+\frac{\gamma}{2}\|\hat{\boldsymbol{\alpha}}\|_F^2 \text{ s.t. } \lambda\mathbb{I}+\hat{\boldsymbol{\alpha}}\succeq\boldsymbol{\Sigma}_{1,1}.$$

In this problem, for any λ, an optimal choice of $\hat{\boldsymbol{\alpha}}$ is given by projecting (with respect to the Frobenius distance) onto a positive semidefinite cone centered at $\boldsymbol{\Sigma}_{1,1}-\lambda\mathbb{I}$. Therefore, an optimal choice of $\hat{\boldsymbol{\alpha}}$ is given by $\hat{\boldsymbol{\alpha}}=(\boldsymbol{\Sigma}_{1,1}-\lambda\mathbb{I})_+$ [see 16, Chapter 8.1.1]. Moreover, we have verified that an optimal choice of λ is indeed given by solving (11), and therefore the result follows. □

We now derive an efficient technique for computing an optimal λ in (11):

Corollary 1 *Let $\boldsymbol{\Sigma}_{1,1}$ be a submatrix containing the entries of $\boldsymbol{\Sigma}$ where $z_i = z_j = 1$, and let $\sigma_1 \geq \ldots \geq \sigma_k$ denote the ordered eigenvalues of $\boldsymbol{\Sigma}_{1,1}$. Then, any λ which solves the following optimization problem is an optimal dual variable in* (11)*:*

$$\min_{\lambda \in \mathbb{R}, \boldsymbol{\theta} \in \mathbb{R}^k_+} \quad \lambda + \frac{\gamma}{2} \sum_{i=1}^{k} \theta_i^2 \ \textit{s.t.}\ \boldsymbol{\theta} \geq \boldsymbol{\sigma} - \lambda \boldsymbol{e}.$$

Moreover, suppose $\sigma_l \geq \lambda \geq \sigma_{l+1}$, where $\lambda := \frac{1}{\gamma l} + \frac{1}{l}\sum_{i=1}^{l} \sigma_i$. Then λ is optimal.

Proof Recall from Theorem 5 that any λ solving $\arg\min_\lambda \left\{\lambda + \frac{\gamma}{2}\|(\boldsymbol{\Sigma}_{1,1} - \lambda \mathbb{I})_+\|_F^2\right\}$ is optimal. Since $\| \left(\Sigma_{1,1} - \lambda \mathbb{I}\right)_+ \|_F^2 = \sum_{i=1}^{k} (\sigma_i - \lambda)_+^2$, this is equivalent to solving

$$\arg\min_{\lambda} \left\{ \lambda + \frac{\gamma}{2} \| \sum_{i=1}^{k} (\sigma_i - \lambda)_+^2 \right\}.$$

The result follows by solving the latter problem. □

As the quadratic optimization problem has a piecewise convex objective, some optimal choice of λ is either an endpoint of an interval $[\sigma_i, \sigma_{i+1}]$ or a solution of the form $\lambda := \frac{1}{\gamma l} + \frac{1}{l}\sum_{i=1}^{l} \sigma_i$ for some l. Therefore, we need only check at most $2k$ different values of λ. Moreover, since the objective function is convex in λ, we can check these points via bisection search, in $O(\log k)$ time.

Finally, observe that the value of the regularization term is always at least $\frac{1}{2\gamma k}$, since

$$\min_{\boldsymbol{X} \succeq \boldsymbol{0}} \quad \frac{1}{2\gamma} \|\boldsymbol{X}\|_F^2 \text{ s.t. } \quad \text{tr}(\boldsymbol{X}) = 1$$

is minimized by setting $\boldsymbol{X} = \frac{1}{n}\boldsymbol{e}\boldsymbol{e}^\top$, and we have the cardinality constraint $X_{i,j} = 0$ if $z_i = 0$, $\boldsymbol{e}^\top \boldsymbol{z} \leq k$. Therefore, we can subtract $\frac{1}{2\gamma k}$ from our circle theorem bound under ridge regularization.

3 Convex Relaxations and Rounding Methods

In this section, we explore two convex relaxations of Problem (1), and propose methods for rounding both relaxations to obtain near-optimal solutions which function as high-quality warm-starts. The first relaxation corresponds to relaxing $\boldsymbol{z} \in \{0,1\}^p$ to $\boldsymbol{z} \in [0,1]^p$ and naturally gives rise to a greedy rounding scheme where we set the largest k elements of $\boldsymbol{z}^\star$, an optimal solution to the convex relaxation, to 1. The second relaxation corresponds to relaxing $\boldsymbol{z} \in \{0,1\}^p$ to $\boldsymbol{z} \in [0,1]^p, \boldsymbol{Z} \succeq \boldsymbol{z}\boldsymbol{z}^\top, Z_{i,i} = z_i, \boldsymbol{Z} \geq \boldsymbol{0}$ and naturally gives rise to Goemans-Williamson rounding [34]. It should not be too surprising that tighter relaxations give rise to more powerful rounding schemes, as relaxations and rounding schemes are indeed two sides of the same coin [5].

3.1 A Boolean Relaxation and a Greedy Rounding Method

We now analyze a Boolean relaxation of (4), which we obtain by relaxing $\boldsymbol{z} \in \{0,1\}^p$ to $\boldsymbol{z} \in [0,1]^p$. This gives:

$$\max_{\boldsymbol{z} \in [0,1]^p} \max_{\boldsymbol{X} \succeq \boldsymbol{0}} \quad \langle \boldsymbol{\Sigma}, \boldsymbol{X} \rangle \text{ s.t. } \text{tr}(\boldsymbol{X}) = 1, |X_{i,j}| \leq M_{i,j} z_i, \forall i, j \in [p]. \tag{12}$$

Algorithm 2 A greedy rounding method for Problem (1)

Require: Covariance matrix $\boldsymbol{\Sigma}$, sparsity parameter k

Compute $\boldsymbol{z}^\star$ solution of

$$\max_{\boldsymbol{X}\in S^p_+, \boldsymbol{z}\in[0,1]^p: \boldsymbol{e}^\top \boldsymbol{z}\leq k} \langle \boldsymbol{\Sigma}, \boldsymbol{X}\rangle \text{ s.t. } \text{tr}(\boldsymbol{X}) = 1, |X_{i,j}| \leq M_{i,j} z_i, \forall i,j \in [p].$$

Construct $\boldsymbol{z} \in \{0,1\}^p : \boldsymbol{e}^\top \boldsymbol{z} = k$ such that $z_i \geq z_j$ if $z_i^\star \geq z_j^\star$.

Compute $\boldsymbol{X}$ solution of

$$\max_{\boldsymbol{X}\in S^p_+} \langle \boldsymbol{\Sigma}, \boldsymbol{X}\rangle \text{ s.t. } \text{tr}(\boldsymbol{X}) = 1, X_{i,j} = 0 \text{ if } z_i z_j = 0, \ \forall i,j \in [p].$$

return $\boldsymbol{z}, \boldsymbol{X}$.

A useful strategy for obtaining a high-quality feasible solution is to solve (12) and set $z_i = 1$ for k indices corresponding to the largest $\boldsymbol{z}_j$'s in (12). We formalize this strategy in Algorithm 2.

We now provide a theoretical guarantee on the quality of the solution returned by the greedy rounding strategy described in Algorithm 2. Formally:

Theorem 6 *Let $\boldsymbol{z}^\star$ denote an optimal solution to Problem* (12)*, $\boldsymbol{z}$ denote a greedily rounded from Algorithm 2, and $\lambda(\boldsymbol{z}^\star), \boldsymbol{\alpha}(\boldsymbol{z}^\star)$ denote an optimal choice of dual variables in* (5)*. Then,*

$$f(\boldsymbol{z}^\star) - f(\boldsymbol{z}) \leq \sum_{i=1}^p L(p-r)(z_i^\star - z_i),$$

where L bounds each $\boldsymbol{\alpha}_{i,j}$ in absolute value, and r denotes the number of indices where $z_i^\star = 1$.

Proof We have that

$$\begin{aligned} f(\boldsymbol{z}^\star) - f(\boldsymbol{z}) &\leq \left(\min_{\substack{\lambda\in\mathbb{R}, \boldsymbol{\alpha}\in\mathbb{R}^{p\times p}: \\ \lambda\mathbb{I}+\boldsymbol{\alpha}\succeq\boldsymbol{\Sigma}}} \lambda + \sum_{i=1}^p z_i^\star \sum_{j=1}^p |\alpha_{i,j}| \right) - \left(\min_{\substack{\lambda'\in\mathbb{R}, \boldsymbol{\alpha}'\in\mathbb{R}^{p\times p}: \\ \lambda'\mathbb{I}+\boldsymbol{\alpha}'\succeq\boldsymbol{\Sigma}}} \lambda + \sum_{i=1}^p z_i^\star \sum_{j=1}^p |\alpha'_{i,j}| \right) \\ &\leq \sum_{i=1}^p (z_i^\star - z_i) \sum_{j=1}^p |\alpha_{i,j}(\boldsymbol{z}^\star))| \leq \sum_{i=1}^p L(p-r)(z_i^\star - z_i), \end{aligned}$$

where the second to last inequality follows from the optimality conditions derived in Theorem 3, and the last inequality holds because $\alpha^\star(\boldsymbol{z}^\star)_{i,j} = 0$ if $z_i^\star = 1$ or $z_j^\star = 1$, by complementary slackness. Given $\boldsymbol{z}^\star$, $\boldsymbol{\alpha}^\star$ we can refine this bound into a considerably better *a posteriori* bound. □

Higher-quality solutions can be obtained by first strengthening the relaxation with second-order cone inequalities, as discussed in Section 3.3, and then performing an analogous greedy rounding strategy. While the worst-case performance bound from Theorem 6 does not apply, we empirically find in our numerical experiments that the strengthened continuous relaxation supplies substantially better rounded solutions than the naive version of Algorithm 2.

3.2 A Doubly Non-Negative Relaxation and Goemans-Williamson Rounding

We now derive a stronger relaxation than Problem (12). Observe that, from a modelling perspective, Problem (12) features products of the original vector $x_i\, x_j$ in both the objective and the constraints. Therefore, unlike several other problems involving cardinality constraints such as compressed sensing, relaxations of sparse PCA benefit from invoking an optimization hierachy [see 24, Section 2.4.1, for a counterexample specific to compressed sensing]. In particular, let us model the outer product $\boldsymbol{z}\boldsymbol{z}^\top$ by introducing a matrix $\boldsymbol{Z}$ and imposing the semidefinite constraint $\boldsymbol{Z} \succeq \boldsymbol{z}\boldsymbol{z}^\top$. We tighten the formulation by requiring that $Z_{i,i} = z_i$ and imposing the RLT inequalities $\max(z_i + z_j - 1, 0) \leq Z_{i,j} \leq \min(z_i, z_j)$. Hence, we obtain the following relaxation:

$$\begin{aligned} \max_{\boldsymbol{z}\in[0,1]^p, \boldsymbol{Z}\in\mathbb{R}^{p\times p}_+} \max_{\boldsymbol{X}\succeq\boldsymbol{0}} \quad & \langle \boldsymbol{\Sigma}, \boldsymbol{X}\rangle \\ \text{s.t.} \quad & \mathrm{tr}(\boldsymbol{X}) = 1, \boldsymbol{e}^\top \boldsymbol{z} \leq k, \langle \boldsymbol{E}, \boldsymbol{Z}\rangle \leq k^2,\ Z_{i,i} = z_i, |X_{i,j}| \leq M_{i,j} Z_{i,j}, \\ & Z_{i,j} \geq \max(z_i + z_j - 1, 0), Z_{i,j} \leq \min(z_i, z_j), \forall i, j \in [p],\ \begin{pmatrix} 1 & \boldsymbol{z}^\top \\ \boldsymbol{z} & \boldsymbol{Z} \end{pmatrix} \succeq \boldsymbol{0}. \end{aligned} \tag{13}$$

Problem (13) is a doubly non-negative relaxation [18, see], as we have intersected inequalities from the Shor and RLT relaxations. This is noteworthy, because doubly non-negative relaxations dominate most other popular relaxations with $O(p^2)$ variables [4, Theorem 1]. Moreover, this relaxation is amenable to a *Goemans-Williamson* rounding scheme [34]. Namely, let $(\boldsymbol{z}^\star, \boldsymbol{Z}^\star)$ denote optimal choices of $(\boldsymbol{z}, \boldsymbol{Z})$ in Problem (13), $\hat{\boldsymbol{z}}$ be normally distributed random vector such that $\hat{\boldsymbol{z}} \sim \mathcal{N}(\boldsymbol{z}^\star, \boldsymbol{Z}^\star - \boldsymbol{z}^\star \boldsymbol{z}^{\star\top})$, and $\bar{\boldsymbol{z}}$ be a rounding of the vector such that $\bar{z}_i = 1$ for the k largest entries of $\hat{z}_i$; this is, up to feasibility on $\hat{\boldsymbol{z}}$, equivalent to the hyperplane rounding scheme of Goemans and Williamson [34] [see 11, for a proof]. We formalize this procedure in Algorithm 3:

Algorithm 3 A Goemans-Williamson [34] rounding method for Problem (1)

Require: Covariance matrix $\boldsymbol{\Sigma}$, sparsity parameter k

Compute $\boldsymbol{z}^\star, \boldsymbol{Z}^\star$ solution of

$$\max_{\substack{\boldsymbol{X}\in S^p_+, \boldsymbol{z}\in[0,1]^p: \boldsymbol{e}^\top\boldsymbol{z}\leq k,\\ \boldsymbol{Z}\in S^p_+: \langle \boldsymbol{E},\boldsymbol{Z}\rangle \leq k^2}} \langle \boldsymbol{\Sigma}, \boldsymbol{X}\rangle \text{ s.t. } \mathrm{tr}(\boldsymbol{X}) = 1, |X_{i,j}| \leq M_{i,j} Z_{i,j}, \forall i, j \in [p], \begin{pmatrix} 1 & \boldsymbol{z}^\top \\ \boldsymbol{z} & \boldsymbol{Z} \end{pmatrix} \succeq \boldsymbol{0},$$
$$Z_{i,i} = z_i, Z_{i,j} \geq \max(z_i + z_j - 1, 0), Z_{i,j} \leq \min(z_i, z_j), \forall i, j \in [p].$$

Compute $\hat{\boldsymbol{z}} \sim \mathcal{N}(\boldsymbol{z}^\star, \boldsymbol{Z}^\star - \boldsymbol{z}^\star \boldsymbol{z}^{\star\top})$

Construct $\bar{\boldsymbol{z}} \in \{0,1\}^p : \boldsymbol{e}^\top \bar{\boldsymbol{z}} = k$ such that $\bar{z}_i \geq \bar{z}_j$ if $\hat{z}_i \geq \hat{z}_j$.

Compute $\boldsymbol{X}$ solution of

$$\max_{\boldsymbol{X}\in S^p_+} \langle \boldsymbol{\Sigma}, \boldsymbol{X}\rangle \text{ s.t. } \mathrm{tr}(\boldsymbol{X}) = 1, X_{i,j} = 0 \text{ if } \bar{z}_i \bar{z}_j = 0,\ \forall i, j \in [p].$$

return $\boldsymbol{z}, \boldsymbol{X}$.

Note that as Algorithm 3 returns one of multiple possible $\bar{\boldsymbol{z}}$'s, a computationally useful strategy is to run the random rounding component several times and return the best solution found.

A very interesting question is whether it is possible to produce a constant factor guarantee on the quality of Algorithm 3's rounding, as Goemans and Williamson [34] successfully did for binary quadratic optimization. Unfortunately, despite our best effort, this does not appear to be possible as the quality of the rounding depends on the value of the optimal dual variables, which are hard to control in this setting. This should not be too surprising for two distinct reasons. Namely, (a) sparse regression, which reduces to sparse PCA [see 26, Section 6.1] is strongly NP-hard [22], and (b) sparse PCA is hard to approximate within a constant factor under the Small Set Expansion (SSE) hypothesis [21], meaning that producing a constant factor guarantee would contradict the SSE hypothesis of Raghavendra and Steurer [50].

We close this section by noting that a similar in spirit (although different in both derivation and implementation) combination of taking a semidefinite relaxation of $\boldsymbol{z} \in \{0,1\}^p$ and rounding *à la* Goemans-Williamson has been proposed for sparse regression [29].

3.3 Valid Inequalities for Strengthening Convex Relaxations

We now propose valid inequalities which allow us to improve the quality of the convex relaxations discussed previously, essentially by borrowing inequalities derived by [25, 9]. Note that as convex relaxations and random rounding methods are two sides of the same coin [5], applying these valid inequalities also improves the quality of the randomly rounded solutions. For concreteness, we focus on improving Problem (12), similar results can be shown for Problem (13) *mutatis mutandis*.

Theorem 7 *Let $\mathcal{P}_{strong}$ denote the optimal objective value of the following problem:*

$$\max_{\boldsymbol{X}\in S^p_+, \boldsymbol{z}\in[0,1]^p:\boldsymbol{e}^\top\boldsymbol{z}\leq k} \langle \boldsymbol{\Sigma}, \boldsymbol{X}\rangle \ s.t. \quad \begin{array}{l} \mathrm{tr}(\boldsymbol{X}) = 1, \sum_{j\in[p]} X_{i,j}^2 \leq X_{i,i} z_i, \\ |X_{i,j}| \leq M_{i,j} z_i, \forall i,j \in [p], \|\boldsymbol{X}\|_1 \leq k. \end{array} \tag{14}$$

Then, the following inequalities hold:

$$\max_{\boldsymbol{z}\in[0,1]^p:\boldsymbol{e}^\top\boldsymbol{z}\leq k} f(\boldsymbol{z}) \geq \mathcal{P}_{strong} \geq \max_{\boldsymbol{z}\in\{0,1\}^p:\boldsymbol{e}^\top\boldsymbol{z}\leq k} f(\boldsymbol{z}).$$

Moreover, suppose that an optimal solution to $\mathcal{P}_{strong}$ is of rank one. Then,

$$\mathcal{P}_{strong} = \max_{\boldsymbol{z}\in\{0,1\}^p:\boldsymbol{e}^\top\boldsymbol{z}\leq k} f(\boldsymbol{z}).$$

Proof We prove the inequalities successively:

- $\max_{\boldsymbol{z}\in[0,1]^p:\boldsymbol{e}^\top\boldsymbol{z}\leq k} f(\boldsymbol{z}) \geq \mathcal{P}_{strong}$: this holds because $\max_{\boldsymbol{z}\in[0,1]^p:\boldsymbol{e}^\top\boldsymbol{z}\leq k} f(\boldsymbol{z})$ is certainly a relaxation of $\mathcal{P}_{strong}$.
- $\mathcal{P}_{strong} \geq \max_{\boldsymbol{z}\in\{0,1\}^p:\boldsymbol{e}^\top\boldsymbol{z}\leq k} f(\boldsymbol{z})$: this holds because $\mathcal{P}_{strong}$ is indeed a valid relaxation of Problem (1) [see 9, for a derivation of the SOCP inequalities].

Finally, suppose that an optimal solution to Problem (14) is of rank one, i.e., the optimal matrix $\boldsymbol{X}$ can be decomposed as $\boldsymbol{X} = \boldsymbol{x}\boldsymbol{x}^\top$. Then, the SOCP inequalities imply that $\sum_{j\in[p]} x_i^2 x_j^2 \leq x_i^2 z_i$. However, $\sum_{j\in[p]} x_j^2 = \mathrm{tr}(\boldsymbol{X}) = 1$, which implies that $x_i^2 \leq x_i^2 z_i$, i.e., $z_i = 1$ for any index i such that $|x_i| > 0$. Since $\boldsymbol{e}^\top\boldsymbol{z} \leq k$, this implies that $\|\boldsymbol{x}\|_0 \leq k$, i.e., $\boldsymbol{X}$ also solves Problem (2). □

Observe that imposing a rank constraint in (14) forces $\boldsymbol{z}$ to be binary. This is perhaps surprising, as the constraint does not explicitly control $\boldsymbol{z}$, and $\boldsymbol{X}$, $\boldsymbol{z}$ are linked via second-order cones. However, it should not be too surprising, as rank constraints offer substantially more modelling power than binary variables [c.f. 45].

4 Numerical Results

We now assess the numerical behavior of the algorithms proposed in Section 2 and 3.

4.1 Performance of Exact Methods

In this section, we apply Algorithm 1 to medium and large-scale sparse principal component analysis problems, with and without Gershgorin circle theorem bounds in the master problem. All experiments were implemented in `Julia` 1.2, using `CPLEX` 12.10 and `JuMP.jl` 0.18.6, and performed on a standard Macbook Pro laptop, with a 2.9GHz 6-Core Intel i9 CPU, using 16 GB DDR4 RAM. We compare our approach to the branch-and-bound algorithm developed by [8] on the UCI `pitprops`, `wine`, `miniboone`, `communities`, `arrythmia` and `micromass` datasets, both in terms of runtime and the number of nodes expanded; we refer to [8, 9] for descriptions of these datasets. Note that we normalized all datasets before running the method (i.e., we compute the leading sparse principal components of correlation matrices). Additionally, we supply all methods with the warm-start used by [8] (i.e., the method of [56]), to maintain a fair comparison, and under big-M regularization impose the valid inequality $\theta \leq \sum_{i=1}^{n} z_i \Sigma_{i,i}$ to accelerate Algorithm 1.

Tables 1–2 report the time for Algorithm 1 (with and without Gershgorin circle theorem bounds in the master problem and with both big-M and ridge regularization) and the method of [8] to identify the leading k-sparse principal component for $k \in \{5, 10\}$, along with the number of nodes expanded, and the number of outer approximation cuts generated. We impose a relative optimality tolerance of 10^{-3} for all approaches. Note that p denotes the dimensionality of the correlation matrix, and $k \leq p$ denotes the target sparsity.

Our main findings from these experiments are as follows:

- For smaller problem sizes, the strength of the cuts developed here allows Algorithm 1 to outperform state-of-the-art methods such as the method of [8].
- For larger problem sizes, the adaptive branching strategy developed outperforms Algorithm 1. This suggests that our method could benefit from using the branching rules developed by [8], rather than using default `CPLEX` branching, since the method of [8] typically expands fewer nodes (even upon including the circle theorem inequalities in the master problem).
- Generating outer-approximation cuts and valid ubber bounds from the Gershgorin circle theorem are both powerful ideas, but the greatest aggregate power appears to arise from intersecting these bounds, rather than using one bound alone.
- The aggregate time spent in user callbacks did not exceed 0.1 seconds in any problem instance considered here, which suggests that the subproblem strategy proposed here is very efficient.
- For the Wine dataset, if we override the warm-start by supplying the solution $\boldsymbol{x} = \boldsymbol{e}_1$, a vector with 1 in the first entry and 0 elsewhere, the method of [8] returns a solution which is about

Table 1 Runtime in seconds under big-M regularization. We run all approaches on one thread, and impose a time limit of 600s. If a solver fails to converge, we report the relative bound gap at termination in brackets, and the no. explored nodes and cuts at the time limit.

Dataset	p	k	Algorithm 1			Algorithm 1+ Circle Theorem			Method of [8]	
			Time(s)	Nodes	Cuts	Time(s)	Nodes	Cuts	Time(s)	Nodes
Pitprops	13	5	0.44	1,890	784	**0.09**	45	22	1.58	7
		10	0.09	438	255	0.08	223	223	0.07	6
Wine	13	5	0.63	2,130	1,138	**0.04**	143	69	0.05	34
		10	0.11	300	463	0.09	364	232	**0.08**	8
Miniboone	50	5	0.09	10	18	0.03	3	6	0.09	3
		10	**0.00**	0	2	0.04	4	6	0.07	3
Communities	101	5	(2.87%)	46,720	24,040	**0.15**	109	2	0.54	92
		10	(13.3%)	44,050	23,140	**0.44**	373	76	0.80	426
Arrhythmia	274	5	(18.1%)	42,470	13,590	5.27	1,080	192	**3.57**	512
		10	(32.6%)	27,860	12,670	(4.21%)	61,000	11,600	1.49	196
Micromass	1300	5	33.99	1,000	509	131.3	4,580	4	**21.94**	927
		10	(107%)	4,380	33,660	378.6	321	16,090	**216.2**	34,710

Table 2 Runtime under ridge regularization. We run all approaches on one thread, and impose a time limit of 600s. If a solver fails to converge, we report the relative bound gap at termination in brackets, and the no. explored nodes and cuts at the time limit. Without the circle theorem we set $\gamma = \frac{1}{k}$; otherwise we set $\gamma = \frac{100}{k}$ to take advantage of the circle theorem.

Dataset	p	k	Algorithm 1			Algorithm 1+ Circle Theorem		
			Time(s)	Nodes	Cuts	Time(s)	Nodes	Cuts
Pitprops	13	5	0.18	37	78	0.42	42	16
		10	**0.04**	7	13	0.68	615	244
Wine	13	5	0.17	259	60	0.10	73	36
		10	0.10	124	40	0.61	394	230
Miniboone	50	5	(80%)	52,200	26,000	**0.01**	0	2
		10	(99.97%)	137,800	13,700	0.07	10	13
Communities	101	5	(139.5%)	104,800	14,830	0.54	272	55
		10	(106.9%)	93,300	9,420	2.20	1,800	328
Arrhythmia	274	5	(147.6%)	68,400	10,840	6.75	1,242	282
		10	(92.2%)	73,600	6,650	(4.63%)	77,200	11,360
Micromass	1300	5	(129.9%)	22,740	5,900	163.2	4	3,809
		10	(131.7%)	43,200	3,060	510.3	21,700	566

1% suboptimal (even with an optimality tolerance of $1e-10$), while our approach returns the optimal solution. Similarly, for the Arrythmia dataset, the method of [8] returns a solution which is at least 0.1% suboptimal when $k = 5$ in the absence of a warm-start, while our approach returns a solution which is 0.1% better. This suggests our approach is numerically more stable.

4.2 Convex Relaxations and Randomized Rounding Methods

In this section, we apply Algorithms 2 and 3 to obtain high quality convex relaxations and feasible solutions for the datasets studied in the previous subsection. We report the quality of the relaxations and randomly rounded solutions both with and without the additional inequalities discussed in Section 3.3, in Tables 3-4 respectively. All experiments were implemented using the same specifications as the previous section. Note that for the Goemans-Williamson rounding method we report the best solution found after 100 iterations of the random rounding procedure, for rounding is cheap compared to solving the continuous relaxation.

Table 3 Quality of relaxation gap (upper bound vs. optimal solution), objective gap (rounded solution vs. optimal solution) and runtime in seconds per approach, without additional inequalities.

Dataset	p	k	Algorithm 2			Algorithm 3		
			Relax. gap (%)	Obj. gap (%)	Time(s)	Relax. gap (%)	Obj. gap (%)	Time(s)
Pitprops	13	5	23.8%	0.00%	0.02	23.8%	6.89%	0.26
		10	1.10%	0.30%	0.03	1.10%	0.00%	0.25
Wine	13	5	36.8%	0.00%	0.02	36.8%	12.5%	0.18
		10	2.43%	0.26%	0.03	2.43%	3.81%	0.15
Miniboone	50	5	781.3%	235.6%	7.37	215.4%	0.00%	359.3
		10	340.6%	117.6%	7.50	340.6%	0.01%	198.6
Communities	101	5	426.3%	17.3%	501.6	227.0%	20.1%	41,048
		10	189.9%	13.6%	475.4	189.9%	57.2%	30,378

Table 4 Quality of relaxation gap (upper bound vs. optimal solution), objective gap (rounded solution vs. optimal solution) and runtime in seconds per approach, with additional inequalities.

Dataset	p	k	Algorithm 2+Inequalities			Algorithm 3+Inequalities		
			Relax. gap (%)	Obj. gap (%)	Time(s)	Relax. gap (%)	Obj. gap (%)	Time(s)
Pitprops	13	5	0.71%	0.00%	0.17	0.00%	0.00%	0.55
		10	0.12%	0.00%	0.27	0.01%	0.00%	0.80
Wine	13	5	1.56%	0.00%	0.24	0.00%	0.00%	0.46
		10	0.40%	0.00%	0.22	0.18%	0.00%	0.45
Miniboone	50	5	0.00%	0.00%	163.3	0.00%	0.00%	656.07
		10	0.00%	0.00%	148.5	0.00%	0.00%	443.80
Communities	101	5	0.07%	0.00%	29,849.8	0.07%	0.00%	30,337.4
		10	0.51%	0.00%	31,902.6	0.51%	0.00%	33,781.7

Observe that applying Algorithms 2 or 3 without the additional inequalities (Table 3) yields rather poor relaxations and randomly rounded solutions. However, by intersecting our relaxations with the additional inequalities from Section 3.3 (Table 4), we obtain extremely high quality relaxations. Indeed, with the additional inequalities, Algorithm 3 identifies the optimal solution

in all instances, and always supplies a bound gap of less than 1%. Moreover, in terms of obtaining high-quality solutions, the new inequalites allow Algorithm 2 to perform as well as Algorithm 3, despite optimizing over a second-order cone relaxation on $\boldsymbol{z}$, rather than a less tractable semidefinite relaxation on $(\boldsymbol{z}, \boldsymbol{Z})$.

The key drawback of applying these methods is that, as implemented in this section, they do not scale to sizes beyond which Algorithm 1 successfully solves. This is a drawback because Algorithm 1 supplies an exact certificate of optimality, while these methods do not. Motivated by this observation, we now explore techniques for scaling Algorithm 2, as it is tangibly more tractable than Algorithm 3 and performs as well in the presence of Section 3.3's inequalities.

4.3 Scalable Dual Bounds and Random Rounding Methods

In this section, we demonstrate that Algorithms 2 can be successfully scaled to generate high-quality bounds for $1000s \times 1000s$ matrices, by borrowing ideas from our prior work [9]. The key difference between this section and our prior work is that here we are interested in scaling both a convex relaxation and a greedy rounding mechanism, rather than only a convex relaxation.

Note that we could employ similar ideas to scale Algorithm 3, but we have refrained from doing so, as Algorithms 2 supplies sufficiently high-quality solutions and is numerically cheaper.

As shown in our prior work, the key bottleneck in solving convex relaxations such as Problem (14) is the presence of the semidefinite constraint $\boldsymbol{X} \succeq \mathbf{0}$. Therefore, to scale Algorithm 2 we (a) replace the semidefinite constraint $\boldsymbol{X} \succeq \mathbf{0}$ with its second-order cone relaxation $X_{i,j}^2 \leq X_{i,i}X_{j,j}, \forall i,j \in [p]$ and (b) impose a small number of cuts of the form $\langle \boldsymbol{X}, \boldsymbol{x}_t\boldsymbol{x}_t^\top \rangle \geq 0$; we refer to [9, Section 3.1] for details on cut generation. Note that if we add sufficiently many cuts then this relaxation matches the SDO relaxation [9, Theorem 1].

Table 5 reports the performance of Algorithm 2 (with the additional inequalities discussed in Section 3.3) when we relax the PSD constraint to requiring that its 2×2 minors are non-negative, and impose either 0 or 20 cuts of the form $\langle \boldsymbol{X}, \boldsymbol{x}_t\boldsymbol{x}_t^\top \rangle \geq 0$.

Observe that if we do not impose any cuts then we obtain a solution within 1% of optimality and provably within 15% of optimality in seconds (resp. minutes) for $p = 100$s (resp. $p = 1000$s). Moreover, if we impose 20 cuts then we obtain a solution within 0.3% of optimality and provably within 2% of optimality in minutes (resp. hours) for $p = 100$s (resp. $p = 1000$s).

To conclude this section, we explore Algorithm 2's ability to scale to even higher dimensional datasets in a high performance setting, by running the method on one Intel Xeon E5–2690 v4 2.6GHz CPU core using 600 GB RAM. Table 6 reports the methods scalability and performance on the Wilshire 5000 [see 9, for a description] `Gisette`, and `Arcene` UCI datasets. For the `Gisette` dataset, we report on the methods performance when we include the first $3,000$ and $4,000$ rows/columns (as well as all $5,000$ rows/columns). Similarly, for the `Arcene` dataset we report on the method's performance when we include the first $6,000$, $7,000$ or $8,000$ rows/columns. We do not report the method's performance when we impose cutting-planes, as solving the relaxation without cuts is already rather time consuming. Moreover, we do not impose the 2×2 minor constraints to save memory, do not impose $|X_{i,j}| \leq M_{i,j}z_i$ for the Arcene dataset to save even

Table 5 Quality of relaxation gap (upper bound vs. optimal solution), objective gap (rounded solution vs. optimal solution) and runtime in seconds per approach, with additional inequalities.

Dataset	p	k	Algorithm 2+Inequalities			Algorithm 2+Inequalities+20 cuts		
			Relax. gap (%)	Obj. gap (%)	Time(s)	Relax. gap (%)	Obj. gap (%)	Time(s)
Pitprops	13	5	4.25%	0.00%	0.02	0.72%	0.00%	0.36
		10	13.55%	0.08%	0.02	0.81%	0.30%	0.36
Wine	13	5	3.24%	0.09%	0.02	1.59%	0.00%	0.38
		10	6.71%	4.22%	0.02	1.24%	0.26%	0.37
Miniboone	50	5	0.00%	0.00%	0.11	0.00%	0.00%	0.11
		10	0.00%	0.00%	0.12	0.00%	0.00%	0.12
Communities	101	5	0.21%	0.00%	0.67	0.07%	0.00%	14.8
		10	1.53%	0.00%	0.68	0.66%	0.00%	14.4
Arrhythmia	274	5	3.44%	0.93%	8.60	1.42%	0.00%	203.6
		10	3.68%	0.97%	8.16	1.33%	0.00%	184.0
Micromass	1300	5	0.04%	0.00%	239.4	0.01%	0.00%	4,639.4
		10	0.63%	0.00%	232.6	0.32%	0.00%	6,391.9

more memory, and report the overall bound gap, as improving upon the randomly rounded solution is challenging in a high-dimensional setting.

Table 6 Quality of bound gap (rounded solution vs. upper bound) and runtime in seconds.

Dataset	p	k	Algorithm 2 (SOC relax)+Inequalities	
			Bound gap (%)	Time(s)
Wilshire 5000	2130	5	0.38%	1,036
		10	0.24%	1,014
Gisette	3000	5	1.67%	2,249
		10	35.81%	2,562
Gisette	4000	5	1.65%	5,654
		10	54.49%	8,452
Gisette	5000	5	2.01%	14,447
		10	2.30%	13,873
Arcene	6000	5	0.01%	3,333
		10	0.06%	3,616
Arcene	7000	5	0.03%	4,160
		10	0.05%	4,594
Arcene	8000	5	0.02%	6,895
		10	0.17%	8,479

Note that we do not report results for the `Arcene` dataset for $p > 8,000$, as computing this requires more memory than was available in our computing environment (i.e. $>$ 600 GB RAM).

These results suggest that if we solve the SOC relaxation using a first-order method rather than an interior point method, our approach could successfully generate certifiably near-optimal PCs when $p = 10,000$s, particularly if combined with a feature screening technique [see 26, 3].

5 Three Extensions and their Mixed-Integer Conic Formulations

We conclude by discussing three extensions of sparse PCA where our methodology is applicable.

5.1 Non-Negative Sparse PCA

One potential extension to this paper would be to develop a certifiably optimal algorithm for non-negative sparse PCA [see 57, for a discussion], i.e., develop a tractable reformulation of

$$\max_{\boldsymbol{x}\in\mathbb{R}^p} \quad \langle \boldsymbol{x}\boldsymbol{x}^\top, \boldsymbol{\Sigma}\rangle \text{ s.t. } \boldsymbol{x}^\top\boldsymbol{x} = 1, \boldsymbol{x} \geq \boldsymbol{0}, \|\boldsymbol{x}\|_0 \leq k.$$

Unfortunately, we cannot develop a MISDO reformulation of non-negative sparse PCA *mutatis mutandis* Theorem 1. Indeed, while we can still set $\boldsymbol{X} = \boldsymbol{x}\boldsymbol{x}^\top$ and relax the rank-one constraint without loss of optimality, if we do so then, by the non-negativity of $\boldsymbol{x}$, lifting $\boldsymbol{x}$ yields:

$$\begin{aligned} \max_{\boldsymbol{z}\in\{0,1\}^p:\boldsymbol{e}^\top\boldsymbol{z}\leq k} \quad & \max_{\boldsymbol{X}\in\mathcal{C}_n} \quad \langle \boldsymbol{\Sigma}, \boldsymbol{X}\rangle \\ & \text{s.t.} \quad \operatorname{tr}(\boldsymbol{X}) = 1,\ X_{i,j} = 0 \text{ if } z_i = 0,\ X_{i,j} = 0 \text{ if } z_j = 0,\ \forall i,j \in [p]. \end{aligned} \tag{15}$$

where $\mathcal{C}_n := \{\boldsymbol{X} : \exists\, \boldsymbol{U} \geq \boldsymbol{0}, \boldsymbol{X} = \boldsymbol{U}^\top\boldsymbol{U}\}$ denotes the completely positive cone, which is NP-hard to separate over and cannot currently be optimized over tractably [28].

Nonetheless, we can develop relatively tractable mixed-integer conic upper and lower bounds for non-negative sparse PCA. Indeed, we can obtain a fairly tight upper bound by replacing the completely positive cone with the larger doubly non-negative cone $\mathcal{D}_n := \{\boldsymbol{X} \in S^p_+ : \boldsymbol{X} \geq \boldsymbol{0}\}$, which is a high-quality outer-approximation of $\mathcal{C}_n$, indeed exact when $k \leq 4$ [19].

Unfortunately, this relaxation is strictly different in general, since the extreme rays of the doubly non-negative cone are not necessarily rank-one when $k \geq 5$ [19]. Nonetheless, to obtain feasible solutions which supply lower bounds, we could inner approximate the completely positive cone with the cone of non-negative scaled diagonally dominant matrices [see 1, 15].

5.2 Sparse PCA on Rectangular Matrices

A second extension would be to extend our methodology to the non-square case:

$$\max_{\boldsymbol{x}\in\mathbb{R}^m, \boldsymbol{y}\in\mathbb{R}^n} \quad \boldsymbol{x}^\top\boldsymbol{A}\boldsymbol{y} \text{ s.t. } \|\boldsymbol{x}\|_2 = 1, \|\boldsymbol{y}\|_2 = 1, \|\boldsymbol{x}\|_0 \leq k, \|\boldsymbol{y}\|_0 \leq k.$$

Observe that computing the spectral norm of a matrix $\boldsymbol{A}$ is equivalent to:

$$\max_{\boldsymbol{X}\in\mathbb{R}^{n\times m}} \quad \langle \boldsymbol{A}, \boldsymbol{X}\rangle \text{ s.t. } \begin{pmatrix} \boldsymbol{U} & \boldsymbol{X} \\ \boldsymbol{X}^\top & \boldsymbol{V} \end{pmatrix} \succeq \boldsymbol{0}, \operatorname{tr}(\boldsymbol{U}) + \operatorname{tr}(\boldsymbol{V}) = 2,$$

where, in an optimal solution, $\boldsymbol{U}$ stands for $\boldsymbol{x}\boldsymbol{x}^\top$, $\boldsymbol{V}$ stands for $\boldsymbol{y}\boldsymbol{y}^\top$ and $\boldsymbol{X}$ stands for $\boldsymbol{x}\boldsymbol{y}^\top$ (this can be seen by taking the dual of [51, Equation 2.4]).

Therefore, by using the same argument as in the positive semidefinite case, we can rewrite sparse PCA on rectangular matrices as the following MISDO:

$$\begin{aligned}\max_{\boldsymbol{w}\in\{0,1\}^m,\boldsymbol{z}\in\{0,1\}^n}\ \max_{\boldsymbol{X}\in\mathbb{R}^{n\times m}}\quad & \langle \boldsymbol{A},\boldsymbol{X}\rangle\\ \text{s.t. } & \begin{pmatrix}\boldsymbol{U} & \boldsymbol{X}\\ \boldsymbol{X}^\top & \boldsymbol{V}\end{pmatrix}\succeq \boldsymbol{0}, \operatorname{tr}(\boldsymbol{U})+\operatorname{tr}(\boldsymbol{V})=2,\\ & U_{i,j}=0 \text{ if } w_i=0, \forall i,j\in[m],\\ & V_{i,j}=0 \text{ if } z_i=0, \forall i,j\in[n], \boldsymbol{e}^\top\boldsymbol{w}\leq k, \boldsymbol{e}^\top\boldsymbol{z}\leq k.\end{aligned}$$

5.3 Sparse PCA with Multiple Principal Components

A third extension where our methodology is applicable is the problem of obtaining multiple principal components simultaneously, rather than deflating $\boldsymbol{\Sigma}$ after obtaining each principal component. As there are three distinct definitions of this problem, which each give rise to different formulations, we now discuss the extent to which our framework encompasses each case.

Common Support: Perhaps the simplest extension of sparse PCA to a multi-component setting arises when all r principal components have common support. By retaining the vector of binary variables $\boldsymbol{z}$ and employing the Ky-Fan theorem [c.f. 55, Theorem 2.3.8] to cope with multiple principal components, we obtain the following formulation in much the same manner as previously:

$$\max_{\boldsymbol{z}\in\{0,1\}^p:\boldsymbol{e}^\top\boldsymbol{z}\leq k}\ \max_{\boldsymbol{X}\in S^p_+}\quad \langle \boldsymbol{X},\boldsymbol{\Sigma}\rangle \text{ s.t. } \boldsymbol{0}\preceq\boldsymbol{X}\preceq\mathbb{I},\ \operatorname{tr}(\boldsymbol{X})=r,\ X_{i,j}=0,\ \text{if } z_i=0,\forall i\in[p].$$

Notably, the logical constraint $X_{i,j}=0$ if $z_i=0$, which formed the basis of our subproblem strategy, still successfully models the sparsity constraint. This suggests that (a) it should be possible to derive an equivalent subproblem strategy under common support, and (b) a cutting-plane method for common support should scale equally well as with a single component.

Disjoint Support: In a sparse PCA problem with disjoint support [54] , simultaneously computing the first r principal components is equivalent to solving:

$$\begin{aligned}\max_{\substack{\boldsymbol{z}\in\{0,1\}^{p\times r}:\boldsymbol{e}^\top\boldsymbol{z}_t\leq k,\forall t\in[r],\\ \boldsymbol{z}\boldsymbol{e}\leq\boldsymbol{e}}}\ \max_{\boldsymbol{W}\in\mathbb{R}^{p\times r}}\quad & \langle \boldsymbol{W}\boldsymbol{W}^\top,\boldsymbol{\Sigma}\rangle\\ & \boldsymbol{W}^\top\boldsymbol{W}=\mathbb{I}_r,\ W_{i,j}=0,\ \text{if } z_{i,t}=0,\forall i\in[p],t\in[r],\end{aligned}$$

where $z_{i,t}$ is a binary variable denoting whether feature i is a member of the tth principal component. By applying the technique used to derive Theorem 1 *mutatis mutandis*, and invoking the Ky-Fan theorem [c.f. 55, Theorem 2.3.8] to cope with the rank-r constraint, we obtain:

$$\begin{aligned}\max_{\boldsymbol{z}\in\{0,1\}^p:\boldsymbol{e}^\top\boldsymbol{z}\leq k}\ \max_{\boldsymbol{X}\in S^p}\quad & \langle \boldsymbol{X},\boldsymbol{\Sigma}\rangle\\ & \boldsymbol{0}\preceq\boldsymbol{X}\preceq\mathbb{I},\ \operatorname{tr}(\boldsymbol{X})=r,\ X_{i,j}=0,\ \text{if } Y_{i,j}=0,\forall i\in[p],\end{aligned}$$

where $Y_{i,j} = \sum_{t=1}^{r} z_{i,t} z_{j,t}$ is a binary matrix denoting whether features i and j are members of the same principal component; this problem can be addressed by a cutting-plane method in much the same manner as when $r = 1$.

General case: In the most general formulation of sparse PCA with multiple principal components (PCs), we only require that all PCs are sparse and orthogonal. This gives rise to:

$$\max_{\boldsymbol{z}\in\{0,1\}^{p\times r}:\boldsymbol{e}^\top \boldsymbol{z}_t \leq k, \forall t\in[r]} \max_{\boldsymbol{W}\in\mathbb{R}^{p\times r}} \quad \langle \boldsymbol{W}\boldsymbol{W}^\top, \boldsymbol{\Sigma}\rangle$$
$$\boldsymbol{W}^\top \boldsymbol{W} = \mathbb{I}_r,\ W_{i,j} = 0,\ \text{if } z_{i,t} = 0, \forall i \in [p], t \in [r].$$

Unfortunately, this problem is far less tractable than the two aforementioned formulations. First, we cannot introduce a binary matrix $\boldsymbol{Y}$ to indicate whether points (i, j) are in the same component, since $\boldsymbol{Y}$ does not capture notions of partially disjoint support (for instance, suppose that three principal components enjoy support on $(1, 2), (2, 3), (1, 3)$. Then, introducing a binary matrix $\boldsymbol{Y}$ indicating whether $X_{i,j}$ may take non-zero values, as successfully done previously, would not render $\boldsymbol{X} = \frac{1}{3}\boldsymbol{e}\boldsymbol{e}^\top$ infeasible). Therefore, we need to relate $\boldsymbol{z}_{i,t}$ to the continuous directly, which necessarily involves optimizing over $\boldsymbol{W}$, rather than $\boldsymbol{X}$. Therefore, this problem is equivalent to a low-rank optimization problem; and we believe it is not mixed-integer semidefinite representable. While low-rank problems scale to lower dimensions than cardinality-constrained problems, they can nonetheless be solved to certifiable optimality or near optimality, as explored in [14].

References

1. Ahmadi AA, Majumdar A (2019) Dsos and sdsos optimization: more tractable alternatives to sum of squares and semidefinite optimization. SIAM J Appl Alg Geom **3**(2):193–230
2. Amini AA, Wainwright MJ (2008) High-dimensional analysis of semidefinite relaxations for sparse principal components. In: Information Theory, 2008. ISIT 2008. IEEE International Symposium on, IEEE, pp 2454–2458
3. Atamtürk A, Gomez A (2020) Safe screening rules for l0-regression. Opt Online
4. Bao X, Sahinidis NV, Tawarmalani M (2011) Semidefinite relaxations for quadratically constrained quadratic programming: A review and comparisons. Math Prog **129**(1):129
5. Barak B, Kelner JA, Steurer D (2014) Rounding sum-of-squares relaxations. In: Proceedings of the forty-sixth annual ACM symposium on Theory of computing, pp 31–40
6. Barnhart C, Johnson EL, Nemhauser GL, Savelsbergh MW, Vance PH (1998) Branch-and-price: Column generation for solving huge integer programs. Oper Res **46**(3):316–329
7. Ben-Tal A, Nemirovski A (2002) On tractable approximations of uncertain linear matrix inequalities affected by interval uncertainty. SIAM J Opt **12**(3):811–833
8. Berk L, Bertsimas D (2019) Certifiably optimal sparse principal component analysis. Math Prog Computation **11**:381–420
9. Bertsimas D, Cory-Wright R (2020) On polyhedral and second-order-cone decompositions of semidefinite optimization problems. Oper Res Letters **48**(1):78–85
10. Bertsimas D, Shioda R (2009) Algorithm for cardinality-constrained quadratic optimization. Comp Opt & Appl **43**(1):1–22

11. Bertsimas D, Ye Y (1998) Semidefinite relaxations, multivariate normal distributions, and order statistics. In: Handbook of Combinatorial Optimization, Springer, pp 1473–1491
12. Bertsimas D, Pauphilet J, Van Parys B (2017) Sparse classification and phase transitions: A discrete optimization perspective. arXiv:171001352
13. Bertsimas D, Cory-Wright R, Pauphilet J (2019) A unified approach to mixed-integer optimization: Nonlinear formulations and scalable algorithms. arXiv:190702109
14. Bertsimas D, Cory-Wright R, Pauphilet J (2020) Mixed projection-conic optimization: A certifiably optimal framework for rank constrained problems. In preparation
15. Bostanabad MS, Gouveia J, Pong TK (2018) Inner approximating the completely positive cone via the cone of scaled diagonally dominant matrices. arXiv:180700379
16. Boyd S, Vandenberghe L (2004) Convex optimization. Cambridge University Press, Cambridge
17. Boyd S, El Ghaoui L, Feron E, Balakrishnan V (1994) Linear matrix inequalities in system and control theory, vol 15. Studies in Applied Mathematics, Society for Industrial and Applied Mathematics, Philadelphia, PA
18. Burer S (2010) Optimizing a polyhedral-semidefinite relaxation of completely positive programs. Math Prog Computation **2**(1):1–19
19. Burer S, Anstreicher KM, Dür M (2009) The difference between 5×5 doubly nonnegative and completely positive matrices. Lin Alg Appl **431**(9):1539–1552
20. Candes E, Tao T (2007) The dantzig selector: Statistical estimation when p is much larger than n. Ann Stat **35**(6):2313–2351
21. Chan SO, Papailliopoulos D, Rubinstein A (2016) On the approximability of sparse pca. In: Conference on Learning Theory, pp 623–646
22. Chen Y, Ye Y, Wang M (2019) Approximation hardness for a class of sparse optimization problems. J Mach Learn Res **20**(38):1–27
23. Coey C, Lubin M, Vielma JP (2020) Outer approximation with conic certificates for mixed-integer convex problems. Math Prog Computation, to appear
24. d'Aspremont A, Boyd S (2003) Relaxations and randomized methods for nonconvex qcqps. EE392o Class Notes, Stanford University 1:1–16
25. d'Aspremont A, El Ghaoui L, Jordan M, Lanckriet G (2005) A direct formulation for sparse pca using semidefinite programming. In: NIPS, pp 41–48
26. d'Aspremont A, Bach F, El Ghaoui L (2008) Optimal solutions for sparse principal component analysis. J Mach Learn Res **9**:1269–1294
27. d'Aspremont A, Bach F, El Ghaoui L (2014) Approximation bounds for sparse principal component analysis. Math Prog **148**(1-2):89–110
28. Dong H, Anstreicher K (2013) Separating doubly nonnegative and completely positive matrices. Math Prog **137**(1-2):131–153
29. Dong H, Chen K, Linderoth J (2015) Regularization vs. relaxation: A conic optimization perspective of statistical variable selection. arXiv:151006083
30. Duran MA, Grossmann IE (1986) An outer-approximation algorithm for a class of mixed-integer nonlinear programs. Math Prog **36**(3):307–339
31. Fletcher R, Leyffer S (1994) Solving mixed integer nonlinear programs by outer approximation. Math Prog **66**(1):327–349

32. Gally T, Pfetsh ME (2016) Computing restricted isometry constants via mixed-integer semidefinite programming. Opt Online
33. Gally T, Pfetsch ME, Ulbrich S (2018) A framework for solving mixed-integer semidefinite programs. Opt Meth & Soft **33**(3):594–632
34. Goemans MX, Williamson DP (1995) Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. J ACM **42**(6):1115–1145
35. Hein M, Bühler T (2010) An inverse power method for nonlinear eigenproblems with applications in 1-spectral clustering and sparse pca. In: NIPS, pp 847–855
36. Horn RA, Johnson CR (1990) Matrix analysis. Cambridge university press
37. Hotelling H (1933) Analysis of a complex of statistical variables into principal components. J Educ Psych **24**(6):417
38. Jeffers JN (1967) Two case studies in the application of principal component analysis. Appl Stat **16**(3):225–236
39. Jolliffe I, Trendafilov NT, Uddin M (2003) A modified principal component technique based on the lasso. J Comp Graph Stat **12**(3):531–547
40. Jolliffe IT (1995) Rotation of principal components: choice of normalization constraints. J Appl Stat **22**(1):29–35
41. Journée M, Nesterov Y, Richtárik P, Sepulchre R (2010) Generalized power method for sparse principal component analysis. J Mach Learn Res **11**(Feb):517–553
42. Kaiser HF (1958) The varimax criterion for analytic rotation in factor analysis. Psychometrika **23**(3):187–200
43. Kim S, Kojima M (2001) Second order cone programming relaxation of nonconvex quadratic optimization problems. Opt Methods & Soft **15**(3-4):201–224
44. Kobayashi K, Takano Y (2020) A branch-and-cut algorithm for solving mixed-integer semidefinite optimization problems. Comp Opt Appl **75**(2):493–513
45. Lubin M (2017) Mixed-integer convex optimization: outer approximation algorithms and modeling power. PhD thesis, Massachusetts Institute of Technology
46. Luss R, d'Aspremont A (2010) Clustering and feature selection using sparse principal component analysis. Opt & Eng **11**(1):145–157
47. Luss R, Teboulle M (2013) Conditional gradient algorithmsfor rank-one matrix approximations with a sparsity constraint. SIAM Rev **55**(1):65–98
48. Moghaddam B, Weiss Y, Avidan S (2006) Spectral bounds for sparse pca: Exact and greedy algorithms. In: NIPS, pp 915–922
49. Quesada I, Grossmann IE (1992) An lp/nlp based branch and bound algorithm for convex minlp optimization problems. Comput & Chem Eng **16**(10-11):937–947
50. Raghavendra P, Steurer D (2010) Graph expansion and the unique games conjecture. In: Proc. ACM, ACM, pp 755–764
51. Recht B, Fazel M, Parrilo PA (2010) Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. SIAM Rev **52**(3):471–501
52. Richman MB (1986) Rotation of principal components. J Climatology **6**(3):293–335
53. Richtárik P, Takáč M, Ahipaşaoğlu SD (2012) Alternating maximization: Unifying framework for 8 sparse pca formulations and efficient parallel codes. arXiv:12124137

54. Vu V, Lei J (2012) Minimax rates of estimation for sparse pca in high dimensions. In: Artificial intelligence and statistics, pp 1278–1286
55. Wolkowicz H, Saigal R, Vandenberghe L (2000) Handbook of semidefinite programming: theory, algorithms, and applications. Springer Science & Business Media
56. Yuan X, Zhang T (2013) Truncated power method for sparse eigenvalue problems. J Mach Learn Res **14**:899–925
57. Zass R, Shashua A (2007) Nonnegative sparse pca. In: NIPS, pp 1561–1568
58. Zhang Y, d'Aspremont A, El Ghaoui L (2012) Sparse pca: Convex relaxations, algorithms and applications. In: Handbook on Semidefinite, Conic and Polynomial Optimization, Springer, pp 915–940
59. Zou H, Hastie T, Tibshirani R (2006) Sparse principal component analysis. J Comp Graph Stat **15**(2):265–286