Mathematical Programming (2020) 184:491–530
https://doi.org/10.1007/s10107-019-01419-7

FULL LENGTH PAPER

Series A

Certifiably optimal sparse inverse covariance estimation

Dimitris Bertsimas[1] · Jourdain Lamperski[1] · Jean Pauphilet[1]

Received: 6 March 2016 / Accepted: 1 August 2019 / Published online: 17 August 2019

Abstract

We consider the maximum likelihood estimation of sparse inverse covariance matrices. We demonstrate that current heuristic approaches primarily encourage robustness, instead of the desired sparsity. We give a novel approach that solves the cardinality constrained likelihood problem to certifiable optimality. The approach uses techniques from mixed-integer optimization and convex optimization, and provides a high-quality solution with a guarantee on its suboptimality, even if the algorithm is terminated early. Using a variety of synthetic and real datasets, we demonstrate that our approach can solve problems where the dimension of the inverse covariance matrix is up to 1000 s. We also demonstrate that our approach produces significantly sparser solutions than Glasso and other popular learning procedures, makes less false discoveries, while still maintaining state-of-the-art accuracy.

Keywords Sparse inverse covariance estimation · Semidefinite optimization · Mixed-integer optimization · Glasso · Maximum likelihood estimation

Mathematics Subject Classification 90C11 · 90C22 · 62H12

1 Introduction

Estimating inverse covariance (precision) matrices is a fundamental task in modern multivariate analysis. Applications include undirected Gaussian graphical models [41], high dimensional discriminant analysis [12], portfolio allocation [22,24], complex data visualization [62], amongst many others, see Fan et al. [25] for a review.

✉ Dimitris Bertsimas
dbertsim@mit.edu

Jourdain Lamperski
jourdain@mit.edu

Jean Pauphilet
jpauph@mit.edu

[1] Massachusetts Institute of Technology, Operations Research Center, Cambridge, MA, USA

For example, in the context of undirected Gaussian graphical models, estimating the precision matrix corresponds to inferring the conditional independence structure on the related graphical model; zero entries in the precision matrix indicate that variables are conditionally independent.

Sparsity of the true precision matrix is a prevailing assumption [10,20,40,54,67] for two reasons.

1. The covariance matrix is often estimated empirically using the maximum likelihood estimator:
$$\overline{\mathbf{\Sigma}} = \frac{1}{n}\sum_{i=1}^{n}(x^{(i)} - \bar{x})(x^{(i)} - \bar{x})^T, \tag{1}$$
where the number of samples n can be lower than the space dimension p. When this is the case, it is known that the empirical covariance matrix[1] $\overline{\mathbf{\Sigma}}$ is singular, and thus does not accurately model the true covariance matrix. Moreover, the empirical covariance matrix can not be inverted to obtain an estimate of the precision matrix. Assuming sparsity of the true precision matrix is required for the precision matrix estimation problem to be well-defined.
2. In many applications, we use models to improve our knowledge of a given phenomenon and it is fair to admit that humans are limited in their ability to understand complex models. As Rutherford D. Roger said ‘We are drowning in information but starving for knowledge’. Models which only involve a small number variables, i.e. sparse models, are inherently simple. Sparse models with high predictive power can thus be extremely valuable in practice. We refer skeptic readers to the first chapter of Hastie et al. [33], which makes a strong case for sparsity in statistical learning.

The most common method for encouraging sparsity in precision matrix estimation involves solving a ℓ_1-regularized maximum likelihood problem. The problem is convex and can be solved in high dimensions. Though this approach is tractable, solutions suffer from similar drawbacks as Lasso solutions in linear regression [9]. For example, one drawback is the ℓ_1-penalty introduces extra bias when estimating nonzero entries in the precision matrix with large absolute values [40].

In this paper, we seek to confront these drawbacks by solving the cardinality constrained optimization problem for which the ℓ_1-regularized problem is a convex surrogate. The cardinality constrained problem parallels the relation the best subset selection (or feature selection) problem plays in linear regression with Lasso. The main goal of this work is to solve the cardinality constrained problem for problem sizes of interest, and compare the solutions with current approaches. A summary of the contributions in this paper is given below.

1. Recent results in linear regression establish that Lasso can be viewed as a robust optimization problem for an appropriately chosen uncertainty set [8,64]. In a seminal paper on precision matrix estimation, [3] already uncovered a similar

[1] Note that $\overline{\mathbf{\Sigma}}$ is not the only estimate of the covariance matrix. In particular, $\frac{n}{n-1}\overline{\mathbf{\Sigma}}$ is a widely-used unbiased estimator of the covariance matrix. In this paper, we will only consider $\overline{\mathbf{\Sigma}}$, which we might refer to as the empirical or sample covariance matrix.

connection, suggesting that the ℓ_1-regularization approach is primarily encouraging robustness and that sparsity is a fortunate by-product. We generalize their result and show that a wide family of regularization can indeed be viewed as a robust version of the inverse covariance estimation problem.
2. We formulate the cardinality constrained maximum likelihood problem for the inverse covariance matrix as a binary optimization problem. We show that the resulting discrete optimization problem is non-smooth in general, but that adding some well-chosen regularization penalty leads to a smooth convex discrete optimization problem. In particular, we show that the well-known big-M formulation or the Ridge regularization term satisfy this property.
3. We propose a combination of outer-approximation algorithm and first-order methods to solve the mixed-integer convex problem. To our knowledge, this is the first time in which such a scheme is used to solve a mixed-integer nonlinear optimization problem with semidefinite constraints. It is well-known that problems of this type are notoriously hard to solve, and we observe that our approach significantly outperforms available mixed-integer nonlinear solvers. An advantage of our approach over existing approaches is that it provides near optimal solutions fast, and a guarantee on the solutions suboptimality if the method is terminated early.
4. We report computational results with both synthetic and real-world datasets that show that our proposed approach can deliver near optimal solutions in a matter of seconds, and provably optimal solutions in a matter of minutes for p in the 100 s and k in the 10 s. The algorithm also provides high-quality solutions to problems in the 1000 s, but a certificate of optimality is more computationally expensive for those sizes.
5. We investigate empirically statistical properties of solutions for the cardinality constrained problem. We compare solutions with ℓ_1-regularized estimates and other popular learning procedures, and observe that cardinality-constrained estimates recover the sparsity pattern of the true underlying precision matrix with comparable accuracy as state-of-the-art but significantly better false detection rate and predictive power.
6. Finally, we show the modeling power of our framework and illustrate how it can be easily adapted to estimate Gaussian graphical with more structural information.

The structure of the paper is as follows: in Sect. 2, we describe the problem of interest and provide a more detailed overview of relevant results from the literature. We generalize existing results about the equivalence between regularization and robustness. From this perspective, ℓ_1-regularized approaches primarily encourage robustness instead of sparsity, which could explain the known drawbacks of these techniques. In Sect. 3 (supplemented by "Appendix A"), we provide a mixed-integer formulation for the cardinality-constrained problem. Though non-smooth in general, we show that adding big-M constraints or a ridge penalty term turns the problem into a smooth convex integer optimization problem, for which we propose an efficient cutting-plane procedure. We also discuss practical implementation and parameter tuning in Sect. 3.4 and "Appendix B". In Sect. 4, we describe and numerically compare first-order and coordinate descent methods to solve variants of the covariance selection problem, used in our algorithm to provide valid cuts. We perform a variety of computational tests in Sect. 5 and “Appendix C”, and use synthetic and real datasets to assess the algorithmic and statistical performance of our approach. Section 6 illustrates the modeling power of our approach by discussing extensions to cases where structural information about the correlation structure is available. In Sect. 7, we provide concluding remarks.

2 Overview and preliminaries

In this section, we provide a description of the problem formulation and an overview of current approaches for inducing sparsity in inverse covariance estimation. Previous work [3] showed that the ℓ_1-regularization approach is equivalent to a robust optimization problem with an appropriately chosen uncertainty set. We generalize their result and discuss practical implications. In particular, this equivalence suggests that current approaches are primarily encouraging robustness, not sparsity.

2.1 Problem description

Let us consider a Gaussian random variable $X \sim N(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ with unknown mean $\boldsymbol{\mu} \in \mathbb{R}^p$ and covariance $\boldsymbol{\Sigma} \in S^p_{++}$, where S^p_{++} denotes the set of symmetric positive definite matrices in $\mathbb{R}^{p \times p}$. Given a random sample $x^{(1)}, \ldots, x^{(n)}$ of X, we seek to estimate the precision matrix $\boldsymbol{\Sigma}^{-1}$. Let $\overline{\boldsymbol{\Sigma}} \in \mathbb{R}^{p \times p}$ be the empirical covariance matrix corresponding to the n observations as defined in (1). The maximum likelihood estimate of $\boldsymbol{\Sigma}^{-1}$ is the solution of the optimization problem

$$\min_{\boldsymbol{\Theta} \succ \mathbf{0}} \quad \langle \overline{\boldsymbol{\Sigma}}, \boldsymbol{\Theta} \rangle - \log \det \boldsymbol{\Theta}, \tag{2}$$

where the expression $\langle \cdot, \cdot \rangle$ is the usual trace inner product $\langle \overline{\boldsymbol{\Sigma}}, \boldsymbol{\Theta} \rangle = \operatorname{tr}(\overline{\boldsymbol{\Sigma}}^\top \boldsymbol{\Theta})$ and the objective function in (2) is the negative Gaussian log-likelihood of the data [67].

As mentioned in introduction, a more interesting problem in practice is the cardinality-constrained version of (2)

$$\min_{\boldsymbol{\Theta} \succ \mathbf{0}} \quad \langle \overline{\boldsymbol{\Sigma}}, \boldsymbol{\Theta} \rangle - \log \det \boldsymbol{\Theta} \quad \text{s.t.} \quad \|\boldsymbol{\Theta}\|_0 \leqslant k, \tag{3}$$

where $k \in \mathbb{Z}_+$, and $\|\boldsymbol{\Theta}\|_0 := \sum_{i>j} 1_{\Theta_{ij} \neq 0}$ counts the number of nonzero entries in the strictly lower triangular part of $\boldsymbol{\Theta}$.

Problem (3) parallels the role best subset selection plays in the context of linear regression. Like best subset selection, the cardinality constraint makes it computationally challenging and indeed NP-hard [14]. There is also the extra difficulty that the problem is a minimization over positive definite matrices S^p_{++}. To our knowledge, the problem has yet to be considered in the literature as a discrete optimization problem over positive definite matrices. Thus, this paper provides the first provably exact optimization approach for solving Problem (3). Closest to our approach are recent works for approximately solving a variant of Problem (3) with an ℓ_0 penalty instead of a constraint. Marjanovic and Hero [46] propose a coordinate descent method to find good stationary solutions. Liu et al. [42] approximate the ℓ_0 pseudo-norm by a series of ridge penalties and implement a variant of the alternating direction method of multipliers.

At the core of our methodology is the exploitation of novel techniques in discrete optimization. Recently, best subset selection and other cardinality constrained problems have been solved in high dimensions, using discrete optimization [6,7,9]. These approaches exploit the significant progress in mixed-integer optimization in the past decades and motivate our approach.

2.2 Notations

In the remaining of the paper, we will use bold characters to denote matrices or matrix-valued functions. Unless otherwise stated, all norms on matrices are vector norms and matrices are $p \times p$ matrices.

Let us recall some linear algebra identities, which will be useful in Sect. 4.3. For any invertible matrix $\mathbf{A}$ and vectors u, v, we can compute the determinant of $\mathbf{A} + uv^T$ ([51], Eqn. 6.2.3)

$$\det(\mathbf{A} + uv^T) = \det(\mathbf{A})\,(1 + v^T \mathbf{A}^{-1} u),$$

and its inverse (Woodbury–Sherman–Morrison Formula in [51], Eqn. 3.8.2)

$$(\mathbf{A} + uv^T)^{-1} = \mathbf{A}^{-1} - \frac{1}{1 + v^T \mathbf{A}^{-1} u} \mathbf{A}^{-1} u v^T \mathbf{A}^{-1}.$$

By default, all vectors are p-dimensional vectors. We will denote by $e_i, i = 1, \ldots, p$ the unit vectors with 1 at the ith coordinate and zero elsewhere, and e the vector of all ones.

2.3 Current approaches

A variety of convex and nonlinear based optimization methods have been proposed to induce sparsity using the maximum likelihood problem [26]. Many of these methods can be interpreted as convex relaxation for Problem (3), the most common of which being the ℓ_1-regularized negative log-likelihood minimization

$$\min_{\mathbf{\Theta} \succ \mathbf{0}} \ \langle \overline{\mathbf{\Sigma}}, \mathbf{\Theta} \rangle - \log\det \mathbf{\Theta} + \lambda \|\mathbf{\Theta}\|_1, \tag{4}$$

where $\|\mathbf{\Theta}\|_1 := \sum_{i,j} |\Theta_{ij}|$ is the ℓ_1 vector norm. In practice, it has been observed that the penalty term shrinks the coefficients of $\mathbf{\Theta}$ towards zero, and produces a sparse solution by setting many coefficients equal to zero. Problem (4) was originally motivated by the development and successes of Lasso as a convex surrogate for the best subset selection problem [67]. The problem is well-studied in the literature [3,29,55,57,67] and solved efficiently with a block coordinate descent procedure. Banerjee et al. [3]

originally proposed the block coordinate descent schema and solved each sub-problem using Nesterov's first-order method. Friedman et al. [29] then suggested a modified version of the algorithm, commonly referred to as Graphical Lasso or Glasso for each sub-problem is reformulated as a Lasso regression problem and solved as such. Mazumder and Hastie [48,49] then further improved the Glasso algorithm through smart feature screening rules. More recently, Krishnamurthy et al. [39] used coordinate descent to solve each sub-problem and released an R package which can solve (4) for a whole regularization path in a short amount of time—within a minute for $p = 1000$. Coordinate descent [57], alternating linearization [58], quadratic approximation and Newton's method [36,37,52], and stochastic proximal methods [2] have also been explored.

In earlier work, Meinshausen et al. [50] proposed an efficient algorithm to discover the sparsity pattern of $\mathbf{\Sigma}^{-1}$ by fitting a Lasso model to each variable, using the others as predictors. It has later been shown [3,29] that their approach can be viewed as an approximation of Problem (4). More recently, Fattahi and Sojoudi [27] proposed a simple thresholding heuristic and explored its connection with the graphical lasso (4)

Though the problem is tractable, it shares in the statistical shortcomings of its motivator, Lasso. Problem (4) leads to biased estimates because the ℓ_1-norm penalty term penalizes large entries more than the smaller entries [40]. Accordingly, upon increasing the degree of regularization, (4) sets more entries of $\mathbf{\Theta}$ to zero but leaves true predictors outside of the support. Thus, as soon as certain regularity conditions on the data are violated, Problem (4) becomes suboptimal as a variable selector and in terms of delivering a model with good predictive performance. In contrast, Problem (3) chooses variables to enter the active set without shrinking the entries in $\mathbf{\Theta}$. Lam and Fan [40] discuss other statistical shortcomings of (4).

To address these shortcomings, other relaxation of (3) have been proposed using smooth nonconvex penalties such as smoothly clipped absolute deviation (SCAD) [21] and minimax concave penalty (MCP) [68], which are folded concave penalties that do not introduce extra bias for estimating nonzero entries with large absolute values. Theoretical properties of these methods are well studied [40,55]. However, these formulations are nonconvex and cannot provide a guarantee on how close their optimal solution is to the optimal solution of Problem (3).

Estimators and approaches other than using maximum likelihood have also been proposed for inducing sparsity. Two such estimators are the constrained ℓ_1-minimization for inverse matrix estimation (CLIME) estimator [12] and the graphical Dantzig selector [66]. Rank and factor based methods have also been proposed; for a more complete survey of the different methods, see Fan et al. [26].

From an optimization perspective, mixed-integer semi-definite optimization (MI-SDP) has received a lot of attention in recent years, for they naturally appear in robust optimization problems with ellipsoidal uncertainty sets [4] or as reformulations of combinatorial problems [60]. Problem-specific MI-SDP strategies have been developed for problems such as binary quadratic programming [34], robust truss topology [65] or the max-cut problem [53]. More recently, rounding and Gomory cuts [1,13], branch-and-bound [30] and outer-approximation schemes [44] have also been developed, in an attempt to provide the same level of general-purpose solvers for MI-SDP as there are for mixed-integer linear optimization. Our approach is similar to the outer-approximation procedure described by Lubin et al. [44] but leverages the specific dependency between the binary and continuous variables in our problem. It also disconnects the combinatorial aspect of the problem from its SDP component, allowing us to benefit both from advances in mixed-integer linear optimization and tailor-made semidefinite strategies.

2.4 Equivalence between regularization and robustness

As originally enunciated by Banerjee et al. [3], the ℓ_1-regularization in (4) is the aftermath of a robust optimization problem. Indeed, one can prove a clear equivalence between regularization and robustification in the case of sparse inverse covariance problems:

Theorem 1.A *For any vector norm* $\|\cdot\|$,

$$\min_{\mathbf{\Theta}\succ 0} \langle \overline{\mathbf{\Sigma}}, \mathbf{\Theta}\rangle - \log\det \mathbf{\Theta} + \lambda\|\mathbf{\Theta}\| = \min_{\mathbf{\Theta}\succ 0} \max_{\mathbf{U}:\|\mathbf{U}\|_\star \leqslant \lambda} \langle \overline{\mathbf{\Sigma}} + \mathbf{U}, \mathbf{\Theta}\rangle - \log\det \mathbf{\Theta},$$

where $\|\cdot\|_\star$ *denotes the dual norm of* $\|\cdot\|$.

Theorem 1.B *For any* (p,q)*-induced norm* $\|\cdot\|_{(p,q)}$,

$$\min_{\mathbf{\Theta}\succ 0} \langle \overline{\mathbf{\Sigma}}, \mathbf{\Theta}\rangle - \log\det \mathbf{\Theta} + \lambda\|\mathbf{\Theta}\|_{(p,q)} = \min_{\mathbf{\Theta}\succ 0} \max_{\mathbf{U}\in\mathcal{U}_{(p,q)}} \langle \overline{\mathbf{\Sigma}} + \lambda\mathbf{U}, \mathbf{\Theta}\rangle - \log\det \mathbf{\Theta},$$

with $\mathcal{U}_{(p,q)} := \left\{uv^T : \|u\|_p = 1,\ \|v\|_{q^\star} = 1\right\}$ *and* $q^\star$ *defined such that* ${}^1\!/_q + {}^1\!/_{q^\star} = 1$.

Let us recall that for any matrix $\mathbf{A}$ and $p, q \in \mathbb{Z}_+ \cup \{\infty\}$, the (p,q)-induced norm of $\mathbf{A}$ is defined as

$$\|\mathbf{A}\|_{(p,q)} := \max_{u:\|u\|_p=1} \|\mathbf{A}u\|_q.$$

In particular, the operator norm or the largest singular value of $\mathbf{A}$ is equal to its (2, 2)-induced norm.

Proof Theorem 1.A follows directly from the definition of the dual norm

$$\|\mathbf{\Theta}\| = \max_{\mathbf{U}:\|\mathbf{U}\|_\star\leqslant 1} \langle \mathbf{U}, \mathbf{\Theta}\rangle.$$

Theorem 1.B follows from the fact that the dual norm of the ℓ_q-norm is the $\ell_{q^\star}$-norm, so that:

$$\|\mathbf{A}\|_{(p,q)} = \max_{u:\|u\|_p=1} \|\mathbf{A}u\|_q = \max_{u:\|u\|_p=1} \max_{v:\|v\|_{q^\star}=1} v^T\mathbf{A}u.$$

□

In the result above, the matrix $\mathbf{U}$ should be interpreted as the amount of noise on the covariance matrix $\overline{\boldsymbol{\Sigma}}$ one wishes to be protected against. Similar equivalence results have been proved in a wide range of other statistical settings [5]. From a Bayesian perspective, regularization can also be derived by imposing some prior distribution on the entries of $\boldsymbol{\Theta}$ and there is a one-to-one correspondence between the class of prior distributions, the corresponding uncertainty set in the robust perspective and the resulting penalty.

In addition to this robustness property, the ℓ_1-norm is fortunately sparsity-inducing. Killing two birds with one stone, ℓ_1-regularization has naturally received a lot of attention from the statistical community. Yet, it is fair to admit that the robustness interpretation of the ℓ_1-norm has been neglected and that many variants of (4) use the ℓ_1-norm solely for sparsity, even though it makes little sense from a robust perspective. For instance, diagonal entries of $\boldsymbol{\Theta}$ should be nonzero—a consequence of Hadamard's inequality and the constraint $\boldsymbol{\Theta} \succ 0$. This motivates the fact that diagonal entries are excluded from the cardinality constraint in (3). Similarly, many derivatives of (4) exclude diagonal entries from the ℓ_1-penalty, which, from a robust point of view, is equivalent to considering that diagonal entries of $\overline{\boldsymbol{\Sigma}}$ are noiseless. To avoid such unrealistic assumptions, robustness and sparsity should, in our opinion, be considered as two distinct properties and be treated as such.

3 Integer optimization perspective

We first formulate Problem (3) as binary optimization problem in Sect. 3.1, and prove that it is non-smooth in general. In practice, introducing big-M constants is a simple way to linearize such mixed-integer bilinear problems. Yet, choosing the right big-M values is hard, making these reformulations not always amenable for computation. We show in Sect. 3.2 that big-M formulations can be viewed as a special case of regularization. With regularization as a unifying perspective, we prove that a certain class of penalty functions leads to smooth convex integer optimization problems and propose a general cutting-plane algorithm to solve them in Sect. 3.3. We believe our approach provides a novel perspective on the big-M paradigm. In particular, we regard big-M more as a smoothing technique than a simple modeling trick and reveal promising alternatives, such as ridge regularization.

3.1 Problem formulation

Let us introduce binary variables $\mathbf{Z}_{ij}$ to encode the support of the inverse covariance matrix $\boldsymbol{\Theta}$. The set of feasible supports is

$$\mathcal{S}_p^k = \left\{ \mathbf{Z} \in \{0,1\}^{p \times p} : \forall i, Z_{ii} = 1 \text{ and } \forall i > j, Z_{ij} = Z_{ji} \text{ and } \sum_{i, j > i} Z_{ij} \leqslant k \right\}.$$

The first set of constraints allows diagonal elements of $\boldsymbol{\Theta}$ to take nonzero values. The second set of constraints follows from the fact that $\boldsymbol{\Theta}$ is symmetric. With these notations, we formulate the cardinality constrained Problem (3) as the mixed-integer optimization problem

$$\min_{\mathbf{Z}\in\mathcal{S}_p^k, \boldsymbol{\Theta}\succ\mathbf{0}} \quad \langle\overline{\boldsymbol{\Sigma}}, \boldsymbol{\Theta}\rangle - \log\det\boldsymbol{\Theta} \quad \text{s.t. } \Theta_{ij}=0 \text{ if } Z_{ij}=0 \ \forall(i,j),$$

which can be considered as a binary-only optimization problem

$$\min_{\mathbf{Z}\in\mathcal{S}_p^k} \quad h(\mathbf{Z}), \tag{5}$$

with the objective function

$$h(\mathbf{Z}) := \min_{\boldsymbol{\Theta}\succ\mathbf{0}} \quad \langle\overline{\boldsymbol{\Sigma}}, \boldsymbol{\Theta}\rangle - \log\det\boldsymbol{\Theta} \quad \text{s.t. } \Theta_{ij}=0 \text{ if } Z_{ij}=0 \ \forall(i,j). \tag{6}$$

The inner-minimization problem defining $h(\mathbf{Z})$ is a so-called covariance selection problem [17], which is a well-studied problem in the literature, and can be efficiently solved. In Sect. 4, we discuss more details of how the problem can be solved using tailored first-order methods [16] or coordinate descent schemes [39,57]. Note that the problem is always feasible since the identity matrix satisfies all the constraints. Fortunately, as a function of $\mathbf{Z}$, $h(\mathbf{Z})$ is convex (see proof in "Appendix A"). However, $h(\mathbf{Z})$ is piece-wise constant and exhibits strong discontinuities. In the following subsection, we explore techniques to reformulate or approximate $h(\mathbf{Z})$ in a smooth convex way, through the unifying lens of regularization.

3.2 Smoothing through regularization

In this section, we explore a regularized version of (6),

$$\tilde{h}(\mathbf{Z}) := \min_{\boldsymbol{\Theta}\succ\mathbf{0}} \quad \langle\overline{\boldsymbol{\Sigma}}, \boldsymbol{\Theta}\rangle - \log\det\boldsymbol{\Theta} + \Omega(\boldsymbol{\Theta}) \quad \text{s.t. } \Theta_{ij}=0 \text{ if } Z_{ij}=0 \ \forall(i,j),$$

where Ω is regularizer, that is, a convex function of $\boldsymbol{\Theta}$. In particular, we are interested in two special cases:

Big-*M regularization:* A traditional way to express the dependency between $\mathbf{Z}$ and $\boldsymbol{\Theta}$ in (6) is to use big-M constraints

$$\tilde{h}(\mathbf{Z}) := \min_{\boldsymbol{\Theta}\succ\mathbf{0}} \quad \langle\overline{\boldsymbol{\Sigma}}, \boldsymbol{\Theta}\rangle - \log\det\boldsymbol{\Theta} \quad \text{s.t. } |\Theta_{ij}| \leqslant M_{ij}Z_{ij} \ \forall(i,j).$$

$M_{ij} \in \mathbb{R}_+$ are constants chosen sufficiently large such that if $\boldsymbol{\Theta}^*$ is a minimizer for Problem (3), then $|\Theta^*_{ij}| \leqslant M_{ij}z_{ij}$. In this case, $\min_{\mathbf{Z}} \tilde{h}(\mathbf{Z}) = \min_{\mathbf{Z}} h(\mathbf{Z})$, i.e., h and $\tilde{h}$ have the same minimum with

$$\Omega(\boldsymbol{\Theta}) = \begin{cases} 0 & \text{if } |\Theta_{ij}| \leqslant M_{ij}, \\ +\infty & \text{otherwise.} \end{cases}$$

Ridge (or *ℓ_2^2) regularization:* One can choose

$$\Omega(\boldsymbol{\Theta}) = \frac{1}{2\gamma}\|\boldsymbol{\Theta}\|_2^2 = \frac{1}{2\gamma}\sum_{i,j}\Theta_{ij}^2,$$

for some positive constant γ. Whatever $\gamma > 0$, $\Omega(\boldsymbol{\Theta}) > 0$, so $\tilde{h}$ is not a reformulation but an upper-approximation of h. Ideally, one would like to minimize $\tilde{h}$ for $1/\gamma \to 0$. However, as previously seen, regularization induces desirable robustness properties, so having $1/\gamma > 0$ may be beneficial from a statistical perspective.

Under some weak assumptions on Ω, which are satisfied in the special cases of big-M and ridge regularization, one can reformulate $\tilde{h}(\mathbf{Z})$ using strong duality:

Theorem 2 *For any $\mathbf{Z} \in \{0,1\}^{p\times p}$ such that $Z_{ii} = 1$ for all $i = 1, \ldots, p$,*

$$\begin{aligned}\tilde{h}(\mathbf{Z}) &:= \min_{\boldsymbol{\Theta}\succ\mathbf{0}} \ \langle\overline{\boldsymbol{\Sigma}}, \boldsymbol{\Theta}\rangle - \log\det\boldsymbol{\Theta} + \Omega(\boldsymbol{\Theta}) \quad \textit{s.t. } \Theta_{ij} = 0 \textit{ if } Z_{ij} = 0\ \forall(i,j), \\ &= \max_{\mathbf{R}:\overline{\boldsymbol{\Sigma}}+\mathbf{R}\succ\mathbf{0}} \ p + \log\det(\overline{\boldsymbol{\Sigma}} + \mathbf{R}) - \langle\mathbf{Z}, \boldsymbol{\Omega}^\star(\mathbf{R})\rangle,\end{aligned}$$

where $\boldsymbol{\Omega}^\star$ is some generalization of the Fenchel conjugate for Ω (see [11], chap. 3.3).

An explicit statement of the assumptions and proof of the theorem can be found in "Appendix A". Theorem 2 calls for a few observations:

1. $\tilde{h}(\mathbf{Z})$ is a point-wise maximum of linear, hence convex, functions of $\mathbf{Z}$. As a result, $\tilde{h}$ is a convex function.
2. With the dual reformulation, it is easy to see that $\tilde{h}(\mathbf{Z})$ remains bounded.
3. For the big-M regularization, Theorem 2 reduces to

$$\begin{aligned}\tilde{h}(\mathbf{Z}) &= \min_{\boldsymbol{\Theta}\succeq\mathbf{0}} \langle\overline{\boldsymbol{\Sigma}}, \boldsymbol{\Theta}\rangle - \log\det\boldsymbol{\Theta} \ \text{ s.t. } |\Theta_{ij}| \leqslant M_{ij}Z_{ij}, \\ &= \max_{\mathbf{R}:\overline{\boldsymbol{\Sigma}}+\mathbf{R}\succ\mathbf{0}} \ p + \log\det(\overline{\boldsymbol{\Sigma}} + \mathbf{R}) - \sum_{i,j} M_{ij}Z_{ij}|R_{ij}|.\end{aligned}$$

4. For the ℓ_2^2-regularization, Theorem 2 reduces to

$$\begin{aligned}\tilde{h}(\mathbf{Z}) &= \min_{\boldsymbol{\Theta}\succ\mathbf{0}} \ \langle\overline{\boldsymbol{\Sigma}}, \boldsymbol{\Theta}\rangle - \log\det\boldsymbol{\Theta} + \frac{1}{2\gamma}\|\boldsymbol{\Theta}\|_2^2 \quad \text{s.t. } \Theta_{ij} = 0 \text{ if } Z_{ij} = 0, \\ &= \max_{\mathbf{R}:\overline{\boldsymbol{\Sigma}}+\mathbf{R}\succ\mathbf{0}} \ p + \log\det(\overline{\boldsymbol{\Sigma}} + \mathbf{R}) - \frac{\gamma}{2}\sum_{i,j} Z_{ij}R_{ij}^2.\end{aligned}$$

5. Given a feasible support $\mathbf{Z}$, we denote by $\mathbf{R}^\star(\mathbf{Z})$ the associated dual variable, i.e., $\tilde{h}(\mathbf{Z}) = p + \log\det(\overline{\boldsymbol{\Sigma}} + \mathbf{R}^\star(\mathbf{Z})) - \langle \mathbf{Z}, \boldsymbol{\Omega}^\star(\mathbf{R}^\star(\mathbf{Z}))\rangle$. Then for any feasible $\mathbf{Z}'$, we have
$$\tilde{h}(\mathbf{Z}') \geqslant \tilde{h}(\mathbf{Z}) + \langle \mathbf{Z}' - \mathbf{Z}, \boldsymbol{\Omega}^\star(\mathbf{R}^\star(\mathbf{Z}))\rangle. \tag{7}$$
The inequality above provides a linear lower-approximation of $\tilde{h}$ which coincides with $\tilde{h}$ at $\mathbf{Z}$. In particular, it proves that $-\boldsymbol{\Omega}^\star(\mathbf{R}^\star(\mathbf{Z}))$ is a subgradient of $\tilde{h}$ at $\mathbf{Z}$. This observation plays a central role in devising a numerical strategy to solve (5).

3.3 Cutting-plane algorithm

Instead of solving the non-smooth integer optimization Problem (5), we consider its regularized proxy
$$\min_{\mathbf{Z}\in\mathcal{S}_p^k} \quad \tilde{h}(\mathbf{Z}), \tag{8}$$
with
$$\begin{aligned}\tilde{h}(\mathbf{Z}) &= \min_{\boldsymbol{\Theta}\succ\mathbf{0}} \quad \langle \overline{\boldsymbol{\Sigma}}, \boldsymbol{\Theta}\rangle - \log\det\boldsymbol{\Theta} + \Omega(\boldsymbol{\Theta}) \quad \text{s.t. } \Theta_{ij} = 0 \text{ if } Z_{ij} = 0\,\forall (i,j),\\ &= \max_{\mathbf{R}:\overline{\boldsymbol{\Sigma}}+\mathbf{R}\succ\mathbf{0}} \quad p + \log\det(\overline{\boldsymbol{\Sigma}} + \mathbf{R}) - \langle \mathbf{Z}, \boldsymbol{\Omega}^\star(\mathbf{R})\rangle,\end{aligned} \tag{9}$$
as studied in the previous section. Our numerical approach substitutes $\tilde{h}$ in (8) by a piece-wise linear lower-approximation and iteratively refines this approximation. This process is equivalent to constraint generation: Applying the inequality (7) at all feasible supports, $\tilde{h}$ can indeed be seen as a piece-wise linear convex function with an exponential number of pieces:
$$\tilde{h}(\mathbf{Z}') = \max\left\{\tilde{h}(\mathbf{Z}) + \langle \mathbf{Z}' - \mathbf{Z}, \boldsymbol{\Omega}^\star(\mathbf{R}^\star(\mathbf{Z}))\rangle \;:\; \mathbf{Z}\in\mathcal{S}_p^k\right\}, \quad \forall \mathbf{Z}'\in\mathcal{S}_p^k,$$
and the algorithm iteratively includes new pieces. The method is referred to in the literature as outer-approximation [19] or generalized Benders decomposition (GBD) and described in pseudo-code in Algorithm 3.1.

We summarize some important observations, properties, and connections to the literature for the above algorithm.

1. Generalized Benders decomposition is a method that can be used to solve convex mixed-integer optimization problems. In this context, Problem (10) is often referred to as the master problem, and Problem (9) is referred to as the (separation) subproblem. The GBD algorithm converges in this context in a finite number of steps because subproblems (9) are convex and satisfy Slater's condition, and the set $\mathcal{S}_p^k$ is finite (see Theorem 2.4 in [31]). Thus, the above algorithm converges to an optimal solution for the cardinality constrained Problem (8) in a finite number of steps.
2. Note that at each iteration the algorithm supplies a feasible solution $\mathbf{Z}_t$, an upper bound $\tilde{h}(\mathbf{Z}_t)$, and a lower bound η_t on the optimal solution. Current heuristic approaches do not offer such a certificate of suboptimality.

Algorithm 3.1 Cutting-plane algorithm

```
Require: Initial point $\mathbf{Z}^{(1)} \in \mathcal{S}_p^k$, sample covariance matrix $\overline{\mathbf{\Sigma}}$, sparsity parameter $k$, and tolerance $\epsilon$.
  $t \leftarrow 1$
  repeat
    Compute $\mathbf{Z}_{t+1}, \eta_{t+1}$ solution of
        $$\min_{\mathbf{Z} \in \mathcal{S}_k^p, \eta} \quad \eta \quad \text{s.t. } \eta \geqslant \tilde{h}(\mathbf{Z}_i) + \langle \mathbf{Z} - \mathbf{Z}_i, \mathbf{\Omega}^\star(\mathbf{R}^\star(\mathbf{Z}_i)) \rangle, \ \forall i = 1, \ldots, t. \qquad (10)$$
    Compute $\mathbf{R}^\star(\mathbf{Z}_{t+1}), \tilde{h}(\mathbf{Z}_{t+1})$ by solving (9).
    $t \leftarrow t + 1$
  until $\eta_t < \tilde{h}(\mathbf{Z}_t) - \varepsilon$
  return $\mathbf{Z}_t$
```

3. Algorithm 3.1 requires to solve a large mixed-integer linear optimization problem each time a new constraint is added. Thus, a branch and bound tree is built at each iteration of the algorithm. Lazy constraint callbacks provide an alternative to building a new branch and bound tree at each iteration of the algorithm. When a constraint is added, instead of resolving the problem, the constraint is added to all active nodes in the current branch-and-bound tree. This enables the same tree to be used for all iterations. This saves the rework of building a new tree every time a mixed-integer feasible solution is found. Lazy constraint callbacks are a relatively new type of callback. CPLEX 12.3 introduced lazy constraint callbacks in 2010 and Gurobi 5.0 introduced lazy constraint callbacks in 2012. To date, the only mixed-integer solvers which provide lazy callback functionality are CPLEX [38], Gurobi [32], and GLPK (see http://gnu.org/software/glpk/).
4. The algorithm can greatly benefit from the choice of a good initial solution $\mathbf{Z}^{(1)}$. In practice, we initialize the algorithm with the support returned by Glasso or Meinshausen and Bühlmann's [50] local neighborhood selection method.

3.4 Implementation considerations and cross-validation

In this section, we describe the grid-search procedure to tune the value of the sparsity level, k, and the regularization parameter, M or γ.

Two alternatives have been considered in the literature for parameter tuning. The first approach is cross-validation: Before any computation, the data is divided into a training and a validation set, typically with a ratio of 2:1. Inverse covariance matrices are computed using the training data only and evaluated out-of-sample on the validation data. We pick the parameter values that lead to the best out-of-sample performance in terms of negative log-likelihood. Though simple, cross-validation does not generally have consistency properties for model selection [59]. Its "leave-one-out" or "multi-fold" variants are computationally more expensive for they repeat this process on multiple training / validation splits. The second approach consists in using an in-sample information criterion, such as the extended information criterion from Foygel and Drton [28]

$$BIC_{1/2}(\mathbf{\Theta}) = n\left[\langle \overline{\mathbf{\Sigma}}, \mathbf{\Theta}\rangle - \log\det\mathbf{\Theta}\right] + \|\mathbf{\Theta}\|_0 \log n + 2\|\mathbf{\Theta}\|_0 \log p,$$

which balances goodness of fit and complexity of the model. This criterion is satisfying for it can be computed in-sample and is asymptotically consistent. Consistency results, however, only hold asymptotically and under some assumptions on the data. We will compare those two approaches numerically in Sect. 5.

We test different values of k in a grid search manner. Let us remark that the sparsity k only impacts the feasible set of Problem (8) and that all linear lower approximations of $\tilde{h}$ generated from solving a particular instance of Problem (8) are valid for any value of k. Practically speaking, we solve a series of problems (8) for decreasing values of k, where each new problem is constructed from the previous one by adding a tighter cardinality constraint. In such a way, each new problem benefits from the cuts generated for previous problems.

Regarding the regularization parameter, we inspect values which are uniformly log-distributed, starting from $M_0 = p/\|\overline{\mathbf{\Sigma}}\|_1$ for the big-M regularization and $\gamma_0 = 4p/\|\overline{\mathbf{\Sigma}}\|_2^2$ for the ridge regularization. Those values follow from bounds on the norm of $\mathbf{\Theta}^\star$, the optimal solution of Problem (8), which we prove in Appendix A.3. For the big-M formulation, we describe an optimization-based approach to find valid M values from any feasible solution in Appendix B.

4 Covariance selection problem

In this section, we investigate numerical strategies to efficiently solve separation subproblems of the form (9). We provided both primal and dual formulations for the separation Problem (9). In Sect. 4.1, we discuss the main advantages of solving the primal vs. the dual formulation. In Sects. 4.2 and 4.3 we describe two families of numerical algorithms. In Sect. 4.4, we compare empirically those algorithms.

4.1 Comparisons between primal and dual approaches

The overall cutting-plane Algorithm 3.1 requires at each iteration not only the optimal value $h(\mathbf{Z})$ but also the associated dual variables $\mathbf{R}^\star(\mathbf{Z})$, which are eventually needed to obtain the subgradients $-\mathbf{\Omega}^\star(\mathbf{R}^\star(\mathbf{Z}))$. For that matter, solving the dual formulation in (9) appears attractive.

In the end, the variables of interest are the primal ones, i.e., the sparse precision matrix. Optimal primal and dual variables satify the KKT conditions $\overline{\mathbf{\Sigma}} + \mathbf{R}^\star - (\mathbf{\Theta}^\star)^{-1} = \mathbf{0}$ (see proof of Theorem 2 in Appendix A.2). So, primal variables can be reconstructed from the dual variables at the cost of a $p \times p$ matrix inversion. Due to numerical errors however, inverting $\mathbf{R}^\star(\mathbf{Z})$ might not lead to a sparse matrix. To that extent, it might be favorable to solve the primal formulation in (9), and obtain dual variables by inverting $\mathbf{\Theta}^\star(\mathbf{Z})$. This computation might be computationally expensive $(O(p^3))$, but $\mathbf{\Theta}^\star$ is sparse, it involves at most $p + 2k$ nonzero coefficients, a pattern which numerical algorithms could exploit.

All in all, the primal and dual formulations seem equally attractive. Moreover, both objective functions involve the log-determinant. As a result, any gradient-based method will require updating the decision variable, as well as its inverse. Matrix inversion is thus the computational bottleneck for both primal and dual methods. Based on these observations, we identified two streams of relevant numerical strategies:

1. The first stream of algorithms implements standard first- or second-order methods to solve the primal problem, leveraging the structure of the sparsity pattern defined by $\mathbf{Z}$ to efficiently compute and update the inverse of $\mathbf{\Theta}$ [16].
2. The second stream consists in coordinate descent methods for either the primal [57] or the dual formulation [39], where each iteration leads to low-rank update of the matrix and its inverse.

4.2 Gradient-based methods for the primal formulation

Dahl et al. [16] proposed an efficient gradient-based algorithm for solving the unregularized covariance selection Problem (6). The gradient of the objective function is

$$\overline{\mathbf{\Sigma}} - \mathbf{\Theta}^{-1}.$$

However, thanks to the constraints that $\Theta_{ij} = 0$ if $Z_{ij} = 0$, only the $p + 2k$ coordinates Θ_{ij} with (i, j) such that $Z_{ij} = 1$ are to be updated. In this context, Dahl et al. [16] showed how a particular kind of sparsity patterns - patterns whose clique graph is chordal (see [16], Section 3 for a definition)—could enable smart block structure decomposition of both $\mathbf{\Theta}$ and its inverse and fast computations of Θ_{ij} and Θ_{ij}^{-1} for the coordinates (i, j) of interest. They also generalize their approach to sparsity patterns which are not chordal, through the use of so-called chordal embeddings. For large and sparse matrices, Dahl et al. [16] report speedups in runtime of two to three orders of magnitude for computing the inverse, and hence the gradient of the objective function. In a similar fashion, their method can accelerate Hessian updates as well. They publicly released `CHOMPACK`, a library which implements sparse matrix computations leveraging chordal sparsity patterns [63].

Lastly, Dahl et al. [16] report that a limited-memory Broyden-Fletcher-Goldfarb-Reeves (BFGS) method significantly outperforms other first order methods, such as conjugate gradient, for the covariance selection Problem (6). Surprisingly, the authors mention but do not numerically compare with coordinate descent methods, which will be the topic of the next section.

In the case of the regularized covariance selection Problem (9), their approach can easily be adapted:

- For big-M regularization, one simply needs to project the iterates to ensure the constraints $|\Theta_{ij}| \leqslant M_{ij}$ are satisfied throughout the algorithm.
- Ridge regularization adds a $\frac{1}{\gamma}\mathbf{\Theta}$ term to the gradient, which raises no additional computational difficulty.

4.3 Coordinate descent methods

Coordinate descent methods are one of the most widely used and highly scalable methods in statistical learning problems. Indeed, as previously mentioned, the most successful methods for ℓ_1-regularized inverse covariance estimation (4) all involve a block coordinate descent strategy for the dual formulation and differ only in the algorithm used to solve the subproblem associated with each block. The caveat in coordinate descent methods often resides in an efficient update step, combined with a good rule for picking the coordinate to update. As noted by many authors in similar contexts [16,39,57], the update step can be computed in closed-form in our case, which makes coordinate descent methods very attractive.

For clarity, we illustrate the main ingredients of these methods on the primal formulation with ℓ_2^2-regularization only, but the same ideas can be applied to the dual formulation and to big-M regularization as well. For a given feasible support $\mathbf{Z}$, we solve

$$\min_{\mathbf{\Theta} \succ \mathbf{0}} \quad \langle \overline{\mathbf{\Sigma}}, \mathbf{\Theta} \rangle - \log\det \mathbf{\Theta} + \frac{1}{2\gamma} \|\mathbf{\Theta}\|_2^2 \quad \text{s.t. } \Theta_{ij} = 0 \text{ if } Z_{ij} = 0.$$

4.3.1 Coefficient updates

Given $\mathbf{\Theta} \succ 0$, we first consider the update of the (i, j)th coefficient with $i \neq j$, that is, $\Theta_{ij} \leftarrow \Theta_{ij} + t$ for some $t \in \mathbb{R}$. In matrix form, this can be written as $\mathbf{\Theta} \leftarrow \mathbf{\Theta} + t(e_i e_j^T + e_j e_i^T)$. Denoting $\mathbf{W} := \mathbf{\Theta}^{-1}$ the inverse of $\mathbf{\Theta}$, we have

$$\log\det(\mathbf{\Theta} + t e_i e_j^T + t e_j e_i^T) = \log\det \mathbf{\Theta} + \log\left(1 + 2W_{ij}t + (W_{ij}^2 - W_{ii}W_{jj})t^2\right),$$

so that the best update is obtained by minimizing

$$2\overline{\Sigma}_{ij}t - \log\left(1 + 2W_{ij}t + (W_{ij}^2 - W_{ii}W_{jj})t^2\right) + \tfrac{1}{\gamma}(\Theta_{ij} + t)^2.$$

Setting the derivative to zero, we find the best update $t^\star$ as the unique solution of the equation

$$2\overline{\Sigma}_{ij} - \frac{2W_{ij} + 2(W_{ij}^2 - W_{ii}W_{jj})t}{1 + 2W_{ij}t + (W_{ij}^2 - W_{ii}W_{jj})t^2} + \tfrac{2}{\gamma}(\Theta_{ij} + t) = 0,$$

which satisfies $1 + 2W_{ij}t + (W_{ij}^2 - W_{ii}W_{jj})t^2 > 0$. The above equation can be reduced into a cubic equation in t.

Regarding diagonal coefficients, the best update for the (i, i)th coefficient, $\Theta_{ii} \leftarrow \Theta_{ii} + 2t$, can similarly be found by minimizing

$$2\overline{\Sigma}_{ii}t - \log(1 + 2W_{ii}t) + \tfrac{1}{2\gamma}(\Theta_{ii} + 2t)^2,$$

over t such that $1 + 2W_{ii}t > 0$, which boils down to solving a quadratic equation.

In both cases, the value $t^\star$ for the best update $\mathbf{\Theta} \leftarrow \mathbf{\Theta} + t^\star(e_i e_j^T + e_j e_i^T)$ can fortunately be computed in closed-form, i.e., constant time. After updating $\mathbf{\Theta}$, $\mathbf{W}$ can be update in $O(p^2)$ steps only, using Woodbury–Sherman–Morrison formula.

Observe that using these one-coordinate updates, the matrix $\mathbf{\Theta}$ remains positive definite throughout the algorithm. Indeed, using Shur complements [69], $\mathbf{\Theta}+t^\star(e_i e_j^T + e_j e_i^T) \succ 0$ if $\mathbf{\Theta} \succ 0$ and $1+2W_{ij}t^\star+(W_{ij}^2 - W_{ii}W_{jj}) > 0$. If the algorithm is properly initialized by a positive definite matrix, positive definiteness of the subsequent iterates then follows by induction.

4.3.2 Update rule and computational complexity

In the case of Glasso, Scheinberg and Rish [57] successfully suggested a greedy rule: at each iteration, the algorithm scans through all the coefficients of $\mathbf{\Theta}$ and compute the objective decrease resulting from their update. Then, only the coefficient leading to the largest improvement is updated, as described in Algorithm 4.1. All together, one iteration of the algorithm updates one coefficient and requires $O(p^2)$ operations, with the update of $\mathbf{W}$ as the computational bottleneck. Note that this strategy is particularly efficient on the primal formulation, since there are only $p + 2k$ potentially nonzero coefficients, compared with $p \times (p + 1)/2$ in the dual.

Algorithm 4.1 Greedy coordinate descent algorithm

```
Require: Support $\mathbf{Z} \in \mathcal{S}_p^k$, sample covariance matrix $\overline{\mathbf{\Sigma}}$, regularization parameter $\gamma$.
  repeat
    For all $(i, j)$ such that $Z_{ij} = 1$, compute the objective decrease resulting from the update of the $(i, j)$th coefficient.
    Update $\mathbf{\Theta} \leftarrow \mathbf{\Theta} + t^\star e_i e_j^T + t^\star e_j e_i^T$ for $(i, j)$ which leads to the biggest improvement.
    Update $\mathbf{W}$ accordingly
  until Stopping criterion
  return $\mathbf{\Theta}$
```

Since updating the inverse of $\mathbf{\Theta}$ remains the challenging part, Krishnamurthy et al. [39] suggested a block coordinate approach for solving the dual formulation of the Lasso estimator (4). We can adapt their approach to our regularized covariance selection problem, both in primal and dual formulation. From a high level perspective, at each iteration, a whole row is updated instead of a single coefficient. The computational cost remains $O(p^2)$ steps per iteration, but one might expect fewer iterations in total. We refer to Krishnamurthy et al. [39] for a detailed presentation of the updates and the overall algorithm.

We terminate the algorithm as soon as the duality gap or the objective decrease is sufficiently small.

4.4 Empirical performance and comparisons

In this section, we compare the computational time required to solve the covariance selection problem by each method and see how they scale with the problem size p and the sparsity k. We also investigated how the conditioning of the problem, through the number of samples n used to compute the empirical covariance matrix $\overline{\mathbf{\Sigma}}$ and the regularization parameter M or γ, impacted computational time. However, we observed little effect and decided not to report those experiments.

4.4.1 Instance generation

As in Yuan and Lin [67], Friedman et al. [29], we consider a full precision matrix $\mathbf{\Theta}_0$ with $\Theta_{ii} = 2$ and $\Theta_{ij} = 1$ for $i \neq j$, in short $\mathbf{\Theta}_0 = \mathbf{I}_p + ee^T$. We then generate n random samples from the normal distribution $\mathcal{N}(0, \mathbf{\Theta}_0^{-1})$ and compute the empirical covariance matrix $\overline{\mathbf{\Sigma}}$. We randomly sample a feasible support $\mathbf{Z}$ from $\mathcal{S}_p^k$ and solve Problem (9).

The degrees of freedom in our simulations are the dimension p and the sparsity level t. Based on those quantities, k and n are fixed to

$$\begin{aligned} k &= \left\lfloor t \, \frac{p(p-1)}{2} \right\rfloor, \\ n &= p. \end{aligned}$$

4.4.2 Methods implementation

For both the big-M and the ℓ_2^2 regularization problem, we implement and compare five methods:

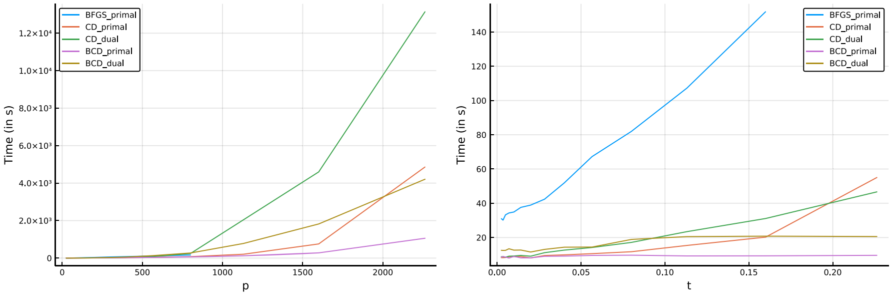
- a BFGS method on the primal formulation (`BFGS_primal`), using the library `CHOMPACK` for sparse matrix computations [63],
- four (block) coordinate descent strategies, denoted `CD_primal`, `CD_dual`, `BCD_primal`, and `BCD_dual`.

All code is written in `Julia 0.6.0` [43], with the exception of the BFGS algorithm, which is implemented in `Python 3.5.3` and integrated into the main `Julia` script using the `PyCall` package. We terminate the algorithms when the duality gap falls below 10^{-4} or the objective improvement after one iteration is less than 10^{-12}.

4.4.3 Empirical results

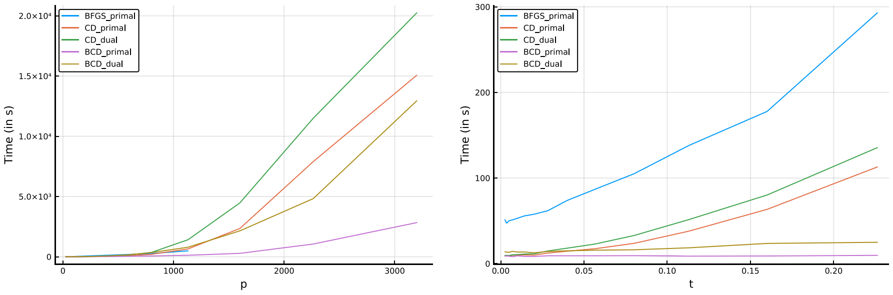
Figures 1 and 2 report computational time as a p and t increase for the big-M and ridge regularization respectively. From these experiments, we can make the following observations:

1. For (block) coordinate descent methods, solving the primal formulation is more effective than solving the dual problem.
2. Coordinate descent methods compete with block coordinate descent schemes when the sparsity level t is very low (less than 1%) but do not scale as well as t increases.
3. As a result, `BCD_primal` is often the best method for solving Problem (9).
4. The `BFGS_primal` algorithm generally takes 50–100 times longer than `BCD_primal`. For $p > 1000$, the algorithm did not terminate after a 12-h time limit.



(a) p, with $t = 1\%$. **(b)** t, with $p = 200$.

Fig. 1 Impact of dimension size p and sparsity level t on computational time, for the big-M regularization with $M = M_0 = p/\|\overline{\mathbf{\Sigma}}\|_1$



(a) p, with $t = 1\%$. **(b)** t, with $p = 200$.

Fig. 2 Impact of dimension size p and sparsity level t on computational time, for the ridge regularization with $\gamma = \gamma_0 = 4p/\|\overline{\mathbf{\Sigma}}\|_2^2$

5 Computational results

In this section, we present numerical results on both synthetic (Sect. 5.1) and real data (Sect. 5.2).

5.1 Synthetic experiments

We follow the methodology described in Banerjee et al. [3]. We sample precision matrices of the form $\mathbf{\Theta}_0 = \delta \mathbf{I}_p + 0.5\mathbf{Z}_0$, where $\mathbf{Z}_0 \in \mathcal{S}^p_{k_{true}}$ and δ is chosen so that the condition number is equal to p. We then randomly sample n vectors from a multivariate normal distribution $\mathcal{N}(0, \mathbf{\Theta}_0^{-1})$, compute the empirical covariance matrix $\overline{\mathbf{\Sigma}}$ and standardize it. To evaluate the output of the algorithms out-of-sample, we generate similarly $n/2$ (resp. $5n$) data points for the validation (resp. test) set.

In this setting, we can assess the feature selection ability of a method in terms of accuracy A, i.e., the fraction of the k_{true} nonzero upper-diagonal coefficients of $\mathbf{\Theta}_0$ correctly recovered, and false detection rate FDR, defined as the proportion of coefficients in the support of the solution which are not in the support of $\mathbf{\Theta}_0$. We also compute the negative log-likelihood ($-LL$) of the returned precision matrix on the test set.

All discrete optimization problems are terminated once the tolerance gap falls below 10^{-4}, where the tolerance gap is the percentage difference between the final lower and upper bounds, or after a 5-min time limit.

5.1.1 Impact of regularization and sparsity *k*

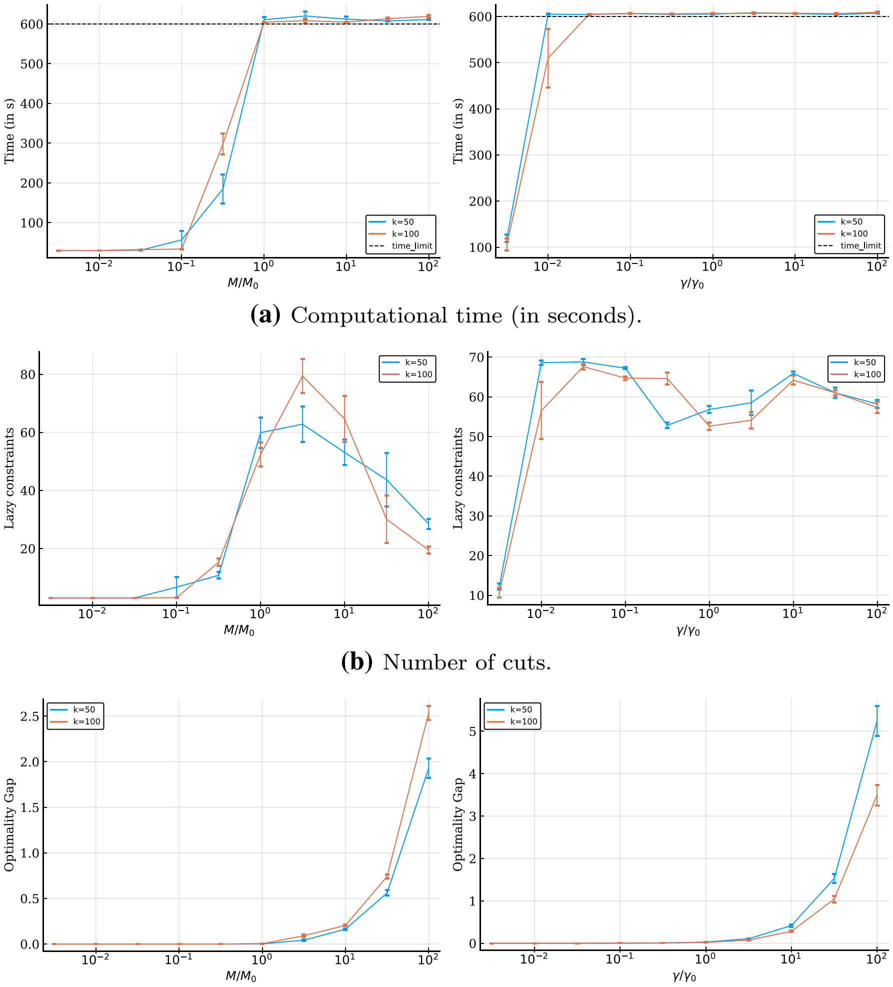
First, we consider one problem instance with $p = 200$, $n/p = 1$, and sparsity level $t_{true} = 1\%$. The discrete formulation (8) involves two hyper-parameters, the sparsity k and the regularization parameter M or γ, which needs to be tuned using grid-search as described in Sect. 3.4.

The value of the regularization parameter has a crucial impact on the overall computational time of the cutting-plane algorithm. Figure 3 shows a steep increase in computational time (top) and in the number of cuts (middle) as the regularization parameter, for both big-M and ridge regularization, increases. Unfortunately, for applications of interest in our experiments, we needed to use high values of M and γ and had to stop the algorithm after a 5-min time limit. Yet, this early stopping strategy did not harm the overall performance of our approach. Indeed, the algorithm is able to find optimal or near-optimal solutions in a short amount of time but spends most of the time proving optimality. For moderate values of M/γ, the optimality gap (Fig. 3c) after 5 min is indeed relatively small, and the algorithm spents a lot of time closing that gap. For large regularization parameter value, on the other hand, the gap increases significantly (over 100%) and becomes uninformative. This corresponds to the regime of most of our subsequent experiments for which we will not report optimality gaps. We provide extensive computational time experiments on smaller-size problems as n, p and k vary in Appendix C.

At the end of the grid search, we select the best pair of parameters and compare the quality of the solution in terms of sparsity, accuracy, false detection and out-of-sample log-likelihood with solutions returned by Glasso [29] and Meinshausen and Bühlmann's approximation scheme [50], implemented in the `R` package `glasso`.[2] We tuned the hyper-parameter ρ in those formulations through a grid search, testing values which led to similar sparsity level k as the discrete formulations. Table 1 (resp. Table 2) reports the results when the hyper-parameters are tuned using the negative log-likelihood on a validation set (resp. the information criterion from Foygel and Drton [28]).

In both cases, we observe that discrete formulations outperform the other two methods in terms of resulting sparsity (by at least 40%), false detection rate (by a factor 4–12) and out-of-sample likelihood (by 11–18%). On the other hand, Meinshausen and Bühlmann's approximation (MB in short) is always the fastest and most accurate method. Actually, we use its solution as a warm-start to our discrete optimization method. Let us remark that the big-M and the ridge formulation perform almost identically and that their performance is barely not impacted by the choice of the criterion. On the contrary, the model selected with Glasso and MB highly depends

[2] Available at https://cran.r-project.org/web/packages/glasso/.



(a) Computational time (in seconds).

(b) Number of cuts.

(c) Relative optimality gap.

Fig. 3 Impact of the regularization parameter M/M_0 for big-M (left), γ/γ_0 for ridge (right) on computational time (top), number of cuts (middle) and relative optimality gap (bottom). For the big-M regularization, $M_0 = p/\|\overline{\mathbf{\Sigma}}\|_1$. For ridge regularization, $\gamma_0 = 4p/\|\overline{\mathbf{\Sigma}}\|_2^2$

on the cross-validation criterion: with negative log-likelihood, both methods tend to select the less sparse model, whereas much sparser models are selected with $BIC_{1/2}$.

5.1.2 Impact of problem size

We now pursue the same comparison for problems with varying characteristics n/p, t and p.

Table 1 Average performance on synthetic data with $p = 200, n/p = 1, t = 1\%$ (leading to $k_{true} = 199$), where the hyper-parameters of each formulation is chosen using the best negative log-likelihood over a validation set

Method	Big-M	Ridge	MB	Glasso
$k^{\star}$	199 (0)	199 (0)	796 (0)	796 (0)
A	0.9508 (0.0080)	0.9508 (0.0080)	0.9960 (0.0020)	0.9945 (0.0023)
FDR	0.0492 (0.0080)	0.0492 (0.0080)	0.6791 (0.0030)	0.7514 (0.0006)
$-LL$	141.39 (3.05)	141.37 (3.05)	157.11 (2.47)	162.05 (1.89)
Time (in s)	352.87 (11.12)	203.36 (39.00)	1.10 (0.04)	3.97 (0.31)

We report the average performance over 10 instances (and their standard deviation)

Table 2 Average performance on synthetic data with $p = 200, n/p = 1, t = 1\%$ (leading to $k_{true} = 199$), where the hyper-parameters of each formulation are chosen using the best in-sample extended Bayesian information criterion $BIC_{1/2}$

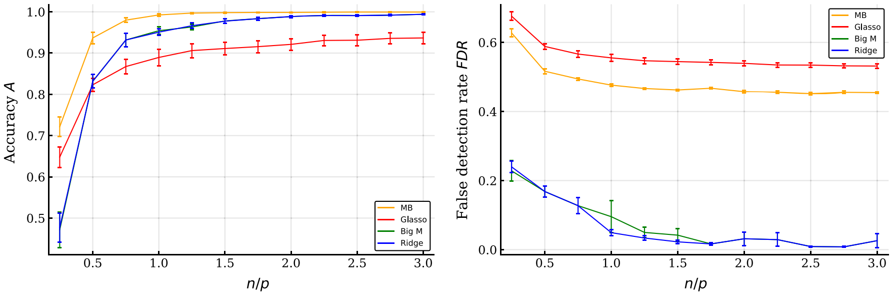
Method	Big-M	Ridge	MB	Glasso
$k^{\star}$	194 (5)	194 (5)	276 (8)	542 (26)
A	0.9317 (0.0081)	0.9317 (0.0081)	0.9890 (0.0037)	0.9814 (0.0047)
FDR	0.0444 (0.0062)	0.0444 (0.0062)	0.2634 (0.0213)	0.6329 (0.0167)
$-LL_{test}$	141.78 (3.24)	141.78 (3.24)	167.16 (2.48)	170.22 (2.42)
Time (in s)	349.5 (14.5)	225.2 (43.00)	0.90 (0.05)	2.77 (0.19)

We report the average performance over 10 instances (and their standard deviation)

Number of samples n. Information-theoretic intuition suggests that the problem becomes easier as n increases. For $n < p$, the empirical covariance matrix is always singular so its inverse cannot be properly defined without sparsity assumptions. On the other side of the spectrum, theoretical guarantees exists for many algorithms [50,56] in the limit $n \to \infty$. As shown on Fig. 4, this intuition is confirmed experimentally with accuracy (resp. false detection rate) increasing (resp. decreasing) as n/p increases. In addition, we observe that the conclusions drawn from the previous section hold consistently for various values of n: the discrete optimization formulations lead to reduced false detection rate, while being of comparable accuracy with the most accurate benchmark. They also demonstrate better out-of-sample negative log-likelihood (Fig. 6 in Appendix D) and their performance is robust to the cross-validation criterion used (Fig. 7 in Appendix D). Note that the other two methods, MB and Glasso, do not exhibit a decreasing false detection rate when cross-validated using the $BIC_{1/2}$ criterion.

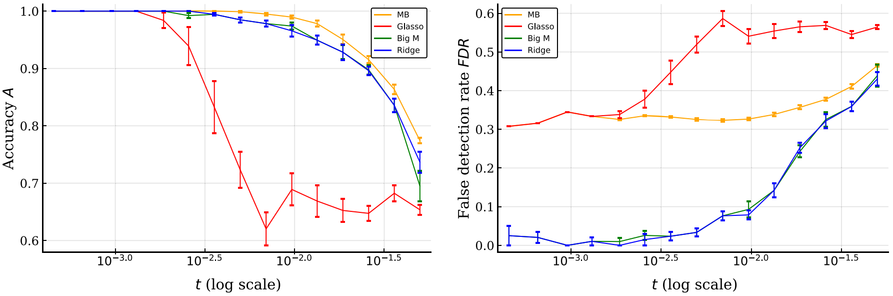
Sparsity level t. Recall that the sparsity level t relates to the number of nonzero upper-diagonal coefficients of $\mathbf{\Theta}_0$ through the relationship

$$k_{true} = \left\lfloor t \, \frac{p(p-1)}{2} \right\rfloor .$$



(a) Accuracy A vs. n/p. **(b)** False detection rate FDR vs. n/p.

Fig. 4 Impact of the number of samples n/p on support recovery. Results are averaged over 10 instances with $p = 200$, $t = 1\%$. Hyper-parameters are tuned using out-of-sample negative log-likelihood



(a) Accuracy A vs. t. **(b)** False detection rate FDR vs. t.

Fig. 5 Impact of the sparsity level t on support recovery. Results are averaged over 10 instances with $p = 200$, $n = p$. Hyper-parameters are tuned using the out-of-sample negative log-likelihood

From Sect. 4.4, we observed that the separation Problem (9) is increasingly harder to solve as t increases. In addition, the combinatorics of the master Problem (8) also increases with t, since the size of the feasible set $\mathcal{S}_p^{k_{true}}$ grows exponentially with k_{true} as long as $k_{true} \leqslant \frac{p(p-1)}{4}$ (i.e., $t \leqslant 0.5$). Figure 5 represents accuracy and false detection rate as t increases, for all methods, using negative log-likelihood as a cross-validation criterion. We report negative log-likelihood and results with $BIC_{1/2}$ as the cross-validation criterion in Appendix D (Figs. 8, 9 respectively).

Dimension p. For n/p and t fixed, the sparse precision matrix estimation problem should not be statistically more difficult as p increases, but computationally more expensive. We report results in Appendix D. Figures 10 and 11 report resulting accuracy and false detection rate as p increases, using negative log-likelihood and $BIC_{1/2}$ respectively as a cross-validation criterion. Figure 12 reports the impact of p on out-of-sample negative log-likelihood, Fig. 13 the impact on time. Interestingly, the big-M formulation is harder to scale than the ridge regularization, due to the additional constraints. As a result, fewer cuts were generated within the 5-min time limit and the resulting precision matrix shows a different accuracy/false detection trade-off with relatively poorer out-of-sample log-likelihood as p increases.

5.2 Analysis of a breast cancer dataset

We apply our method on a real breast cancer dataset analyzed in [35]. The dataset can be found at http://bioinformatics.mdanderson.org/. The dataset consists of 22,283 gene expression levels for 133 patients, including 34 with pathological complete response (pCR) and 99 with residual disease (RD). The pCR subjects are considered to have a high chance of cancer-free survival in the long term, and thus it is of interest to study the response states of the patients (pCR or RD) to preoperative chemotherapy. The main objective of this analysis is to estimate the inverse covariance matrix of the gene expression levels and then apply linear discriminant analysis (LDA) to predict whether or not a subject can achieve the pCR state.

The dataset has been studied in [23] using Glasso, revised Glasso, and SCAD. Later the same analysis was performed with the CLIME estimator [12]. For the sake of consistency, we perform the same analysis, but use our method to estimate inverse covariance matrices when needed. We first briefly describe how the data is prepared and analyzed. We then present our results and compare with known results in [12,23].

The data is first randomly divided into testing and training sets using stratified sampling. 5 pCR subjects and 16 RD subjects are randomly chosen to constitute the testing data. The remaining 112 subjects are chosen to constitute the training data. This process is repeated 100 times and the following data preparation techniques are used on each of the 100 instances of the training and testing data. A two-sample t test is performed between the two groups in the training dataset to determine the most significant genes; we retain the 113 genes with the smallest p values as the variables for prediction and the rest are discarded. The data for each variable (gene) is then standardized by dividing the data with the corresponding standard deviation, estimated from the training dataset.

We next perform the linear discriminant analysis. We assume the normalized gene expression data are normally distributed as $\mathcal{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma})$, where the two groups have the same covariance $\boldsymbol{\Sigma}$, but different means, $\boldsymbol{\mu}_k$ ($k = 1$ for pCR and $k = 2$ for RD). The linear discriminant scores are as follows:

$$\delta_k(\mathbf{x}) = \mathbf{x}^\top \hat{\boldsymbol{\Sigma}}^{-1} \hat{\boldsymbol{\mu}}_k - \frac{1}{2} \hat{\boldsymbol{\mu}}_k^\top \hat{\boldsymbol{\Sigma}}^{-1} \hat{\boldsymbol{\mu}}_k + \log \pi_k,$$

where $\pi_k = n_k/n$ is the proportion of the number of observations in the training data belonging to class k, and the classification rule is given by $\text{argmax}_k\, \delta_k(\mathbf{x})$. Based on each training dataset, we estimate the mean $\hat{\boldsymbol{\mu}}_k$ as,

$$\hat{\boldsymbol{\mu}}_k = \frac{1}{n_k} \sum_{i \in class-k} \mathbf{x}_i \quad \text{for } k = 1, 2,$$

and the precision matrix $\hat{\boldsymbol{\Sigma}}^{-1}$ using the cardinality constrained problem. Since the sample size is less than the dimension of the matrix, the empirical covariance is not invertible and can not be used in LDA.

Table 3 Metrics used for prediction performance comparison for the breast cancer dataset

Comparison metrics	Description
Specificity	$\frac{TN}{TN+FP}$
Sensitivity	$\frac{TP}{TP+FN}$
MCC	$\frac{TP\times TN-FP\times FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}}$

TP, TN, FP, and FN are the number of true positives, true negatives, false positives and false negatives, respectively. Positives correspond to pCR subjects and negatives correspond to RD subjects

Table 4 Comparison of estimators on the breast cancer dataset

Method	Specificity	Sensitivity	MCC	NNZ
Glasso	0.768 (0.009)	0.630 (0.021)	0.366 (0.018)	3923 (2)
Adaptive Lasso	0.787 (0.009)	0.622 (0.022)	0.381 (0.018)	1233 (1)
SCAD	0.794 (0.009)	0.634 (0.022)	0.402 (0.020)	674 (1)
CLIME	0.749 (0.009)	0.806 (0.017)	0.506 (0.020)	492 (7)
Big-M	0.779 (0.011)	0.717 (0.019)	0.460 (0.019)	436 (3)
Ridge	0.775 (0.011)	0.716 (0.020)	0.453 (0.021)	427 (3)

Data for Glasso, revised Glasso and SCAD is from [23] and data for CLIME is from [12]. Average performance is reported on 100 instances of training and testing data; standard deviations are included in parentheses. NNZ refers to the number of nonzero entries in the estimate

The classification performance of δ_k is clearly associated with the estimation performance of $\hat{\mathbf{\Sigma}}^{-1}$. Let true positive (TP) be the number of pCR subjects δ_k identifies as pCR subjects and let true negative (TN) be the number of RD subjects δ_k identifies as RD Subjects. To compare prediction performance, we use comparison metrics: specificity, sensitivity, and also Matthews Correlation Coefficient (MCC). They are each defined in Table 3. MCC is widely used in machine learning for assessing the quality of a binary classifier; it takes true and false, positives and negatives, into account and is generally regarded as a balanced measure. A larger MCC value indicates a better classifier [23].

We perform the LDA for each of the 100 instances and report a summary of average performance in Table 4. For each experiment, we calibrate the parameters k and M/γ using the extended Bayesian information criterion on the training data. We observe that our proposed methods outperform Lasso-based methods on all aspects. Our discrete optimization formulations are comparable to SCAD and Clime, yet not dominated nor dominating by either of the two. Big-M and ridge formulations improve over SCAD in terms of sensitivity and MCC, and over Clime in terms of specificity. On the contrary, SCAD ranks first on specificity and Clime on sensitivity and MCC. However, the biggest advantage of discrete formulations over the others is that they produce sparser estimates. This is especially desirable in the context of graphical models, when it is desirable to induce sparsity for explanatory and predictive power.

6 Extension to graphical model estimation with structural information

In this section, we illustrate the modeling power of our mixed-integer formulation. In graphical models estimation, it is not unusual to have some information or intuition about the correlation structure between variables [18], information which can easily be encoded in our framework by additional constraints on the binary variables $\mathbf{Z}$.

Sparsity In this paper, we focused on imposing sparsity on the precision matrix $\boldsymbol{\Theta}$. This requirement translates into the linear constraint

$$\sum_{i>j} Z_{ij} \leqslant k.$$

Partial knowledge of the support In some settings, the modeler has some partial knowledge of the correlation structure and can inform the optimization problem through the additional constraints

$$\begin{aligned} Z_{ij} &= 0, \text{ if } (i,j) \in \mathcal{S}_0, \\ Z_{ij} &= 1, \text{ if } (i,j) \in \mathcal{S}_1, \end{aligned}$$

where $\mathcal{S}_0$ (resp. $\mathcal{S}_1$) is a set of indices for which Θ_{ij}s are known to be 0 (resp. $\neq 0$).

Degree Information about the degree of each variable in the underlying structure (or graph) might also be relevant [45]. In a protein contact graph for example, the degree of each node is upper bounded by some constant. With our framework, the degree of any variable i is given by $d_i := \sum_{j \neq i} Z_{ij}$, so that adding the linear constraints

$$\ell_i \leqslant d_i \leqslant u_i, \; \forall i$$

would enforce lower (ℓ_i) and upper (u_i) bounds on the node degrees. In a more flexible fashion,

$$\left| \frac{1}{p} \sum_i d_i - \overline{d} \right| \leqslant \epsilon,$$

requires the average node degree to be within ϵ from a given target $\overline{d}$. Similarly, quadratic constraints could be added in order to match second moments. Finally, many real-world networks, including the network of webpages or some gene regulatory networks, involve nodes which have a lot more edges than the others [61]. Our framework can account for such hubs by introducing additional binary variables y_i, $i = 1, \ldots, p$ and adding the following constraints

$$d_i \leqslant d_{low} + (d_{high} - d_{low}) y_i, \ \forall i,$$
$$\sum_i y_i \leqslant m,$$

where d_{high} (resp. d_{low}) is the maximum degree of a hub (resp. non-hub) node and m is an upper-bound on the total number of hubs in the network.

Tree structure Finally, tree-structured graphical models have been extensively studied in the literature [15] for they are sparse and allow efficient inference. Introducing additional binary variables $y_{i,j}^k$ for all ordered triples (i, j, k) of pairwise different nodes, Martin [47] provided an extended formulation for a spanning tree:

$$\begin{aligned}
&\|Z\|_0 = p - 1, \\
&y_{ij}^k + y_{ji}^k = Z_{ij}, \qquad \forall i, j = 1, \ldots, p, \ i < p, \ \forall k = 1, \ldots, p, \\
&\sum_{j : j \notin \{i,k\}} y_{ij}^k = 1 - Z_{ik} \quad \forall i, k = 1, \ldots, p, \ i < k,
\end{aligned}$$

where $y_{ij}^k = 1$ if and only if the edge (i, j) is contained in the tree and k is in the component of j when removing (i, j) from the tree.

7 Summary

In this work, we use a variety of modern optimization methods to provide the first provably exact algorithm for solving the cardinality-constrained negative log-likelihood Problem (3). Through the unifying lens of regularization, we show that the well known big-M constraints are not only a formulation technique but more importantly a smoothing procedure. On that matter, ridge regularization can be considered as a fruitful alternative. Our cutting-plane approach has the additional benefit of treating separately the combinatorial aspect of the problem from the SDP component of it. The method provides provably optimal solutions, and delivers near optimal solutions in minutes for p in the 1000 s and sparsity level of the order of 1%. Computational experiments on both synthetic and real data show that such discrete formulations deliver solutions with increased out-of-sample predictive power and lower false detection rate than existing methods, while being as accurate.

A Proofs of Theorem 2 and corollaries

In this section, we detail the proof of Theorem 2. We first specify the assumptions required on the regularizer Ω, prove Theorem 2 and finally investigate some special cases of interest.

A.1 Assumptions

We first assume that the function Ω is decomposable, i.e., there exist scalar functions Ω_{ij} such that

$$\forall\, \mathbf{\Phi}, \quad \Omega(\mathbf{\Phi}) = \sum_{i,j} \Omega_{ij}(\Phi_{ij}). \tag{A1}$$

In addition, we assume that for all (i, j), Ω_{ij} is convex and tends to regularize towards zero. Formally,

$$\forall\, (i, j), \quad \min_x \ \Omega_{ij}(x) = \Omega_{ij}(0). \tag{A2}$$

Those first two assumptions are not highly restrictive and are satisfied by ℓ_∞-norm constraint (big-M), ℓ_1-norm regularization (LASSO) or $\|\cdot\|_2^2$-regularization, among others.

For any function f, we denote with a superscript $\star$ its Fenchel conjugate (see [11], chap. 3.3) defined as

$$f^\star(y) := \sup_x \langle x, y\rangle - f(x).$$

In particular, the Fenchel conjugate of any function f is convex. Given Assumption (A1),

$$\begin{aligned}\Omega^\star(\mathbf{R}) &= \sup_{\mathbf{\Phi}} \langle \mathbf{\Phi}, \mathbf{R}\rangle - \Omega(\mathbf{\Phi}),\\ &= \sum_{i,j} \sup_{\Phi_{ij}} \Phi_{ij} R_{ij} - \Omega_{ij}(\Phi_{ij}),\\ &= \sum_{i,j} \Omega_{ij}^\star(R_{ij}).\end{aligned}$$

As a result, it is easy to see that if Ω satisfies (A1) and (A2), so does its Fenchel conjugate.

Let us denote $\mathbf{A} \circ \mathbf{B}$ the Hadamard or component-wise product between matrices $\mathbf{A}$ and $\mathbf{B}$. Consider a matrix $\mathbf{R}$ and a support matrix $\mathbf{Z} \in \{0, 1\}^{p \times p}$. The function $\mathbf{Z} \mapsto \Omega^\star(\mathbf{Z} \circ \mathbf{R})$ is convex in $\mathbf{Z}$, by convexity of $\Omega^\star$. We now assume that it is linear in $\mathbf{Z}$, that is, there exists a function $\mathbf{\Omega}^\star : \mathbb{R}^{p\times p} \to \mathbb{R}^{p\times p}$ satisfying:

$$\forall\, \mathbf{Z} \in \{0, 1\}^{p\times p}, \forall\, \mathbf{R} \in \mathbb{R}^{p\times p}, \ \Omega^\star(\mathbf{Z} \circ \mathbf{R}) = \langle \mathbf{Z}, \mathbf{\Omega}^\star(\mathbf{R})\rangle. \tag{A3}$$

A.2 Proof of Theorem 2

Given $\mathbf{Z} \in \{0, 1\}^{p\times p}$ such that $Z_{ii} = 1$ for all $i = 1, \dots, p$, we first prove that under assumptions (A1) and (A2):

$$\begin{aligned}\tilde{h}(\mathbf{Z}) := \ &\min_{\mathbf{\Theta}\succ\mathbf{0}} \ \langle \overline{\mathbf{\Sigma}}, \mathbf{\Theta}\rangle - \log\det \mathbf{\Theta} + \Omega(\mathbf{\Theta}) \quad \text{s.t. } \Theta_{ij} = 0 \text{ if } Z_{ij} = 0\, \forall (i, j),\\ = \ &\max_{\mathbf{R}:\overline{\mathbf{\Sigma}}+\mathbf{R}\succ\mathbf{0}} \ p + \log\det(\overline{\mathbf{\Sigma}} + \mathbf{R}) - \Omega^\star(\mathbf{Z} \circ \mathbf{R}).\end{aligned}$$

Then, Assumption (A3) will conclude the proof.

Proof We decompose the minimization problem *à la Fenchel*.

$$\begin{aligned}\tilde{h}(\mathbf{Z}) &= \min_{\mathbf{\Theta}\succ\mathbf{0}} \langle\overline{\mathbf{\Sigma}}, \mathbf{\Theta}\rangle - \log\det\mathbf{\Theta} + \Omega(\mathbf{\Theta}) \text{ s.t. } \Theta_{ij} = 0 \text{ if } Z_{ij} = 0,\\ &= \min_{\mathbf{\Theta}\succ\mathbf{0},\mathbf{\Phi}} \langle\overline{\mathbf{\Sigma}}, \mathbf{\Theta}\rangle - \log\det\mathbf{\Theta} + \Omega(\mathbf{Z}\circ\mathbf{\Phi}) \text{ s.t. } \Theta_{ij} = Z_{ij}\Phi_{ij},\\ &= \min_{\mathbf{\Theta}\succeq\mathbf{0},\mathbf{\Phi}} \langle\overline{\mathbf{\Sigma}}, \mathbf{\Theta}\rangle - \log\det\mathbf{\Theta} + \Omega(\mathbf{Z}\circ\mathbf{\Phi}) \text{ s.t. } \mathbf{\Theta} = \mathbf{Z}\circ\mathbf{\Phi}.\end{aligned}$$

In the last equality, we omitted the constraint $\mathbf{\Theta} \succ \mathbf{0}$, which is implied by the domain of log det. Assuming (A1) and (A2) hold, the regularization term $\Omega(\mathbf{Z}\circ\mathbf{\Phi})$ can be replaced by $\Omega(\mathbf{\Phi})$ and

$$\tilde{h}(\mathbf{Z}) = \min_{\mathbf{\Theta}\succeq\mathbf{0},\mathbf{\Phi}} \langle\overline{\mathbf{\Sigma}}, \mathbf{\Theta}\rangle - \log\det\mathbf{\Theta} + \Omega(\mathbf{\Phi}) \text{ s.t. } \mathbf{\Theta} = \mathbf{Z}\circ\mathbf{\Phi}.$$

The above objective function is convex in $(\mathbf{\Theta}, \mathbf{\Phi})$, the feasible set is a non-empty - $\mathbf{\Theta} = \mathbf{\Phi} = \mathbf{I}_p$ is feasible - convex set, and Slater's conditions are satisfied. Hence, strong duality holds.

$$\begin{aligned}\tilde{h}(\mathbf{Z}) &= \min_{\mathbf{\Theta}\succeq\mathbf{0},\mathbf{\Phi}} \langle\overline{\mathbf{\Sigma}}, \mathbf{\Theta}\rangle - \log\det\mathbf{\Theta} + \Omega(\mathbf{\Phi}) \text{ s.t. } \mathbf{\Theta} = \mathbf{Z}\circ\mathbf{\Phi},\\ &= \min_{\mathbf{\Theta}\succeq\mathbf{0},\mathbf{\Phi}} \max_{\mathbf{R}} \langle\overline{\mathbf{\Sigma}}, \mathbf{\Theta}\rangle - \log\det\mathbf{\Theta} + \Omega(\mathbf{\Phi}) + \langle\mathbf{\Theta} - \mathbf{Z}\circ\mathbf{\Phi}, \mathbf{R}\rangle,\\ &= \max_{\mathbf{R}} \min_{\mathbf{\Theta}\succeq\mathbf{0}} \left[\langle\overline{\mathbf{\Sigma}} + \mathbf{R}, \mathbf{\Theta}\rangle - \log\det\mathbf{\Theta}\right] + \min_{\mathbf{\Phi}} \left[\Omega(\mathbf{\Phi}) - \langle\mathbf{Z}\circ\mathbf{\Phi}, \mathbf{R}\rangle\right].\end{aligned}$$

For the first inner-minimization problem, first-order conditions $\overline{\mathbf{\Sigma}} + \mathbf{R} - \mathbf{\Theta}^{-1} = \mathbf{0}$ lead to the constraint $\overline{\mathbf{\Sigma}} + \mathbf{R} \succ 0$ and the objective value is $p + \log\det(\overline{\mathbf{\Sigma}} + \mathbf{R})$. The second inner-minimization problem is almost the definition of the Fenchel conjugate:

$$\begin{aligned}\min_{\mathbf{\Phi}} \Omega(\mathbf{\Phi}) - \langle\mathbf{Z}\circ\mathbf{\Phi}, \mathbf{R}\rangle &= -\max_{\mathbf{\Phi}}\langle\mathbf{\Phi}, \mathbf{Z}\circ\mathbf{R}\rangle - \Omega(\mathbf{\Phi}),\\ &= -\Omega^\star(\mathbf{Z}\circ\mathbf{R})\end{aligned}$$

Hence,

$$h(\mathbf{Z}) = \max_{\mathbf{R}:\overline{\mathbf{\Sigma}}+\mathbf{R}\succ\mathbf{0}} p + \log\det(\overline{\mathbf{\Sigma}} + \mathbf{R}) - \Omega^\star(\mathbf{Z}\circ\mathbf{R}).$$

□

Remark Notice that we proved that $\tilde{h}(\mathbf{Z})$ could be written as point-wise maximum of *concave* functions of $\mathbf{Z}$. Assumption (A3) is needed to ensure that the function in the maximization is convex in $\mathbf{Z}$ at the same time.

A.3 Special cases and corollaries

A.3.1 No regularization

We first consider the unregularized case of (6) where $\forall\ \mathbf{\Phi},\ \Omega(\mathbf{\Phi}) = 0$. Assumptions (A1) and (A2) are obviously satisfied. Moreover, for any $\mathbf{R}$,

$$\Omega^\star(\mathbf{R}) = \sup_{\mathbf{\Phi}} \langle \mathbf{\Phi}, \mathbf{R} \rangle = \begin{cases} 0 & \text{if } \mathbf{R} = \mathbf{0}, \\ +\infty & \text{otherwise.} \end{cases}$$

With the convention that $0 \times \infty = 0$, Assumption (A3) is satisfied and Theorem 2 holds:

$$\begin{aligned} h(\mathbf{Z}) &= \max_{\mathbf{R}:\overline{\mathbf{\Sigma}}+\mathbf{R}\succ\mathbf{0}} p + \log\det(\overline{\mathbf{\Sigma}} + \mathbf{R}) - \langle \mathbf{Z}, \mathbf{\Omega}^\star(\mathbf{R}) \rangle, \\ &= \max_{\mathbf{R}:\overline{\mathbf{\Sigma}}+\mathbf{R}\succ\mathbf{0}} p + \log\det(\overline{\mathbf{\Sigma}} + \mathbf{R}) \quad \text{s.t. } Z_{ij} R_{ij} = 0,\ \forall (i, j). \end{aligned}$$

In particular, this reformulation proves that $h(\mathbf{Z})$ is convex,[3] but that the coordinates of its sub-gradient $-\mathbf{\Omega}^\star(\mathbf{R}^\star(\mathbf{Z}))$ are either 0 or $-\infty$, hence uninformative. Note that the same conclusion is true for ℓ_1-regularization.

From the proof of Theorem 2, one can derive a lower bound on $\|\mathbf{\Theta}^\star\|_\infty$ which will be useful for big-M regularization.

Theorem 3 *The solution of (8) satisfies* $\|\mathbf{\Theta}^\star\|_\infty \geqslant \frac{p}{\|\overline{\mathbf{\Sigma}}\|_1}$

Proof For a feasible support $\mathbf{Z}$, denote the optimal primal and dual variables $\mathbf{\Theta}^\star(\mathbf{Z})$ and $\mathbf{R}^\star(\mathbf{Z})$ respectively. There is no duality gap and KKT condition $\mathbf{\Theta}^\star(\mathbf{Z})^{-1} = \overline{\mathbf{\Sigma}}+\mathbf{R}^\star(\mathbf{Z})$ holds, so that $\langle \overline{\mathbf{\Sigma}}, \mathbf{\Theta}^\star(\mathbf{Z}) \rangle = p$. From Hölder's inequality, we obtain the desired lower bound. □

A.3.2 Big-*M* regularization

For the big-M regularization,

$$\Omega(\mathbf{\Theta}) = \begin{cases} 0 & \text{if } |\Theta_{ij}| \leqslant M_{ij}, \\ +\infty & \text{otherwise} \end{cases},$$

is decomposable with $\Omega_{i,j}(\Theta_{ij}) = 0$ if $|\Theta_{ij}| \leqslant M_{ij}$, $+\infty$ otherwise. Assumptions (A1) and (A2) are satisfied. Moreover, for any $\mathbf{R}$,

$$\Omega^\star(\mathbf{R}) = \sup_{\mathbf{\Phi}\,:\,\|\mathbf{\Phi}\|_\infty \leqslant \mathbf{M}} \langle \mathbf{\Phi}, \mathbf{R} \rangle = \|\mathbf{M} \circ \mathbf{R}\|_1.$$

[3] Convexity of $h(\mathbf{Z})$ can also be proved from the primal formulation (6) directly. Take two matrices $\mathbf{Z}_1$ and $\mathbf{Z}_2$, $\lambda \in (0, 1)$, $\mathbf{Z} := \lambda \mathbf{Z}_1 + (1 - \lambda)\mathbf{Z}_2$, then it follows from the definition (6) that $h(\mathbf{Z}) \leqslant \lambda h(\mathbf{Z}_1) + (1 - \lambda) h(\mathbf{Z}_2)$.

In particular, for any binary matrix $\mathbf{Z}$,

$$\Omega^\star(\mathbf{Z} \circ \mathbf{R}) = \sum_{i,j} |M_{ij} Z_{ij} R_{ij}| = \sum_{i,j} M_{ij} Z_{ij} |R_{ij}|,$$

so that Assumption (A3) is satisfied with $\mathbf{\Omega}^\star(\mathbf{R}) = \left(M_{ij}|R_{ij}|\right)_{ij}$.

A.3.3 Ridge regularization

For the ℓ_2^2-regularization,

$$\Omega(\mathbf{\Theta}) = \frac{1}{2\gamma}\|\mathbf{\Theta}\|_2^2,$$

is decomposable with $\Omega_{i,j}(\Theta_{ij}) = \frac{1}{2\gamma}\Theta_{ij}^2$. Assumptions (A1) and (A2) are satisfied. Moreover, for any $\mathbf{R}$,

$$\Omega^\star(\mathbf{R}) = \sup_{\mathbf{\Phi}} \langle \mathbf{\Phi}, \mathbf{R}\rangle - \frac{1}{2\gamma}\|\mathbf{\Phi}\|_2^2 = \frac{\gamma}{2}\|\mathbf{R}\|_2^2$$

In particular, for any binary matrix $\mathbf{Z}$,

$$\Omega^\star(\mathbf{Z} \circ \mathbf{R}) = \frac{\gamma}{2}\sum_{i,j}(Z_{ij}R_{ij})^2 = \frac{\gamma}{2}\sum_{i,j} Z_{ij} R_{ij}^2,$$

since $Z_{ij}^2 = Z_{ij}$, so that Assumption (A3) is satisfied with $\mathbf{\Omega}^\star(\mathbf{R}) = \left(\frac{\gamma}{2}R_{ij}^2\right)_{ij}$.

Moreover, from the proof of Theorem 2, one can connect the norm of $\mathbf{\Theta}^\star(\mathbf{Z})$ and γ.

Theorem 4 *For any support $\mathbf{Z}$, the norm of the optimal precision matrix $\mathbf{\Theta}^\star(\mathbf{Z})$ is bounded by*

$$\frac{\gamma}{2}\|\overline{\mathbf{\Sigma}}\|_2\left(\sqrt{1+\frac{4p}{\gamma\|\overline{\mathbf{\Sigma}}\|_2^2}}-1\right) \leqslant \|\mathbf{\Theta}^\star(\mathbf{Z})\|_2 \leqslant \sqrt{p\gamma}.$$

Proof There is no duality gap:

$$\begin{aligned}\langle \overline{\mathbf{\Sigma}}, \mathbf{\Theta}^\star(\mathbf{Z})\rangle &- \log\det \mathbf{\Theta}^\star(\mathbf{Z}) + \frac{1}{2\gamma}\|\mathbf{\Phi}^\star(\mathbf{Z})\|_2^2 \\ &= p + \log\det(\overline{\mathbf{\Sigma}} + \mathbf{R}^\star(\mathbf{Z})) + \frac{\gamma}{2}\|\mathbf{Z}\circ\mathbf{R}^\star(\mathbf{Z})\|_2^2.\end{aligned}$$

In addition, the following KKT conditions hold

$$\mathbf{\Theta}^\star(\mathbf{Z})^{-1} = \overline{\mathbf{\Sigma}} + \mathbf{R}^\star(\mathbf{Z}),$$

$$\mathbf{\Phi}^\star(\mathbf{Z}) = \gamma \mathbf{Z} \circ \mathbf{R}^\star(\mathbf{Z}),$$

where the second condition follows from the inner minimization problem defining $\Omega^\star$. All in all, we have

$$\langle \overline{\mathbf{\Sigma}}, \mathbf{\Theta}^\star(\mathbf{Z})\rangle + \frac{1}{\gamma}\|\mathbf{\Phi}^\star(\mathbf{Z})\|_2^2 = p.$$

Since $\mathbf{\Sigma}$ and $\mathbf{\Theta}^\star(\mathbf{Z})$ are semi-definite positive matrices, $\langle \overline{\mathbf{\Sigma}}, \mathbf{\Theta}^\star(\mathbf{Z})\rangle \geqslant 0$. Hence,

$$\|\mathbf{\Phi}^\star(\mathbf{Z})\|_2 \leqslant \sqrt{p\gamma}.$$

To obtain the lower bound, we apply Cauchy-Schwartz inequality $\langle \overline{\mathbf{\Sigma}}, \mathbf{\Theta}^\star(\mathbf{Z})\rangle \leqslant \|\overline{\mathbf{\Sigma}}\|_2\|\mathbf{\Theta}^\star(\mathbf{Z})\|_2$ and solve the quadratic equation

$$\frac{1}{\gamma}\|\mathbf{\Phi}^\star(\mathbf{Z})\|_2^2 + \|\overline{\mathbf{\Sigma}}\|_2\|\mathbf{\Theta}^\star(\mathbf{Z})\|_2 - p \geqslant 0.$$

□

In particular, the lower bound in Theorem 4 is controlled by the factor $\frac{4p}{\gamma\|\overline{\mathbf{\Sigma}}\|_2^2}$, suggesting an appropriate scaling of γ to start a grid search with.

B An optimization approach for finding big-*M* values

In this section, we present a method for obtaining suitable constants $\mathbf{M}$. The approach involves solving two optimization problems for each off-diagonal entry of the matrix being estimated. The problems provide lower and upper bounds for each entry of the optimal solution. First we present the problems, then we discuss how they are solved.

B.1 Bound optimization problems

Let $\hat{\mathbf{\Theta}}$ be a feasible solution for (3) and define,

$$u := \langle \hat{\mathbf{\Theta}}, \overline{\mathbf{\Sigma}}\rangle - \log\det\hat{\mathbf{\Theta}}.$$

A simple way to obtain lower bounds for the ijth entry of the optimal solution is to solve

$$\begin{aligned} \min_{\mathbf{\Theta}\succ\mathbf{0}} \quad & \Theta_{ij} \\ \text{s.t.} \quad & \langle \overline{\mathbf{\Sigma}}, \rangle - \log\det\mathbf{\Theta} \leqslant u. \end{aligned} \tag{11}$$

Likewise, to obtain upper bounds we solve

$$\begin{aligned} \max_{\boldsymbol{\Theta} \succ \mathbf{0}} \quad & \Theta_{ij} \\ \text{s.t.} \quad & \langle \overline{\boldsymbol{\Sigma}}, \rangle - \log\det \boldsymbol{\Theta} \leqslant u. \end{aligned} \tag{12}$$

Note that it is sufficient to find a feasible solution $\hat{\boldsymbol{\Theta}}$ to formulate (11) and (12), and a feasible solution with a smaller value leads to better bounds.

B.2 Solution approach

We describe the approach for the lower bound Problem (11) only, the upper bound Problem (12) being similar.

First, we make the additional assumption that $\overline{\boldsymbol{\Sigma}}$ is invertible. We know this assumption cannot hold in the high dimensional setting where $p > n$. Numerically, one can always argue that the lowest eigenvalues of $\overline{\boldsymbol{\Sigma}}$ are never exactly equal to zero but should be strictly positive. In this case however, these eigenvalues should be small and close to machine precision, making matrix inversion very unstable. Note that this extra assumption is required for problems (11) and (12) to be bounded.

Problem (11) is a semidefinite optimization problem and there are $^{p(p+1)}/_2$ entries to bound so it is necessary to efficiently solve (11) and avoid solving so many SDPs. Instead, one can solve the dual of (11) very efficiently. Note an advantage for considering the dual is we do not need to solve the problem to optimality to obtain a valid bound. Using basic arguments from convex duality theory similar to the ones invoked in Sect. A.2, the dual problem for (11) writes

$$\max_{\lambda > 0} \left\{ \lambda \left(p - u + \log\det\left(\frac{1}{2\lambda}(e_i e_j^T + e_j e_i^T) + \overline{\boldsymbol{\Sigma}} \right) \right) \right\} \tag{13}$$

Computationally, problem (13) is easier to solve because it is a convex optimization problem with a scalar decision variable λ.

Denote $g(\lambda)$ the objective function in the dual Problem (13). Algebraic manipulations yield

$$\begin{aligned} g(\lambda) &:= \lambda \left[p - u + \log\det\left(\frac{1}{2\lambda}(e_i e_j^T + e_j e_i^T) + \overline{\boldsymbol{\Sigma}} \right) \right], \\ &= \lambda \left[p - u + \log\det(\overline{\boldsymbol{\Sigma}}) + \log\left(1 + \frac{\Theta_{ij}}{\lambda} + \frac{\Theta_{ij}^2 - \Theta_{ii}\Theta_{jj}}{4\lambda^2} \right) \right], \end{aligned}$$

where $\boldsymbol{\Theta} = \overline{\boldsymbol{\Sigma}}^{-1}$. We can then easily derive the first and second derivatives of g and apply Newton's method to solve Problem (13).

C Additional material on computational performance of the cutting-plane algorithm

In this section, we consider the runtime of the cutting-plane algorithm on synthetic problems as in Sect. 5.1. In Sect. 5.1.1, we illustrated how the regularization parameter M or γ can impact the convergence of the cutting-plane algorithm, so we focus in this section on the impact of the problem sizes n, p and k.

In particular, we study the time needed by the algorithm to find the optimal solution (opt-time) and to verify the solution's optimality (ver-time), as well as the number of cuts required (laz-cons). We carry out all experiments by generating 10 instances of synthetic data[4] for $(p, k_{true}) \in \{30, 50, 80, 120, 200\} \times \{5, 10\}$ and different values of n. We solve each instance of (8) with big-M regularization for $k = k_{true}$, $M = 0.5$ and report average performance in Table 5. These computations are performed on 4 Intel E5-2690 v4 2.6 GHz CPUs (14 cores per CPU, no hyper threading) with 16GB of RAM in total. We chose to fix the value of $M = 0.5$ in order to isolate the impact of p, k and n on computational time, the specific value 0.5 being informed by the knowledge of the ground truth.

In general the algorithm provides an optimal solution in a matter of seconds, and a certificate of optimality in seconds or minutes even for p in the 100 s. Optimal verification occurs significantly quicker when the sample size n is larger because the sparsity pattern of the underlying matrix is easier to recover. However, we note that finding the optimal solution is not as affected by the sample size n. As p or k increase, optimal detection also does not significantly change, but optimal verification generally becomes significantly harder. Similar observations have been made for mixed-integer formulations of the best subset selection problem in linear and logistic regression [9]. We also observe that changes in k have a more substantial impact on the runtime than changes in n or p, especially when p is large. Finally, Meinshausen and Bühlmann's approximation is used as a warm-start and we observe that is often optimal, especially when n/p is large.

Thus, the cutting-plane algorithm in general provides an optimal or near-optimal solution fast, but optimal verification strongly depends on p, k, and n. Nonetheless, we observe that optimality of solutions can be verified for p in the 100 s and k in the 10 s in a matter of minutes.

[4] For each instance, we generate a sparse precision matrix $\boldsymbol{\Theta}_0$ as in Sect. 5.1 and n samples from the corresponding multivariate normal distribution.

Table 5 Average performance on instances of synthetic data with $k = k_{true}$

p	k_{true}	n	Ver-time	Opt-time	Cut-time	Laz-cons
30	5	200	2.37 (2.13)	0.0 (0.0)	1.95 (1.74)	28 (17.9)
		150	6.33 (7.34)	0.0 (0.0)	2.71 (3.14)	55 (55.8)
		100	30.7 (47.96)	0.0 (0.0)	14.46 (28.55)	258 (472.6)
30	10	300	31.11 (23.31)	5.05 (10.69)	14.32 (9.91)	265 (176.6)
		250	35.13 (28.89)	11.2 (13.13)	19.93 (14.91)	296 (204.8)
		200	33.7 (24.23)	7.75 (12.34)	15.35 (11.15)	290 (196.5)
50	5	200	9.59 (9.06)	0.0 (0.0)	5.23 (3.66)	42 (25.2)
		150	29.43 (20.28)	0.0 (0.0)	18.49 (12.98)	153 (107.0)
		100	183.7 (243.73)	0.0 (0.0)	99.36 (118.0)	788 (937.8)
50	10	300	24.19 (20.29)	0.0 (0.0)	12.57 (10.37)	98 (80.8)
		250	31.37 (18.48)	0.0 (0.0)	15.2 (9.46)	122 (77.8)
		200	40.38 (29.27)	0.55 (1.73)	26.14 (19.14)	210 (149.1)
80	5	200	70.12 (106.16)	0.0 (0.0)	51.56 (80.18)	154 (212.2)
		150	179.76 (175.22)	0.0 (0.0)	127.19 (110.85)	404 (348.3)
		100	988.9 (763.05)	0.0 (0.0)	482.83 (277.33)	1581 (990.9)
80	10	300	37.83 (9.17)	0.0 (0.0)	30.33 (10.11)	85 (25.2)
		250	71.4 (24.51)	0.0 (0.0)	47.06 (13.24)	139 (36.3)
		200	161.8 (74.35)	9.87 (31.2)	105.48 (41.14)	309 (121.6)
120	5	200	152.54 (113.42)	34.89 (110.34)	119.24 (99.43)	170 (108.9)
		150	713.45 (712.74)	251.25 (543.17)	480.18 (407.96)	740 (648.4)
		100	1793.67 (445.58)	646.84 (827.53)	1135.33 (320.83)	1671 (412.7)
120	10	300	238.7 (150.61)	0.0 (0.0)	172.75 (99.92)	224 (116.4)
		250	704.43 (568.93)	0.0 (0.0)	396.44 (238.16)	560 (348.5)
		200	1379.58 (666.52)	0.0 (0.0)	675.81 (248.96)	909 (393.1)
200	5	200	858.4 (770.03)	418.1 (496.15)	662.22 (567.77)	398 (335.0)
		150	1453.51 (614.68)	515.58 (548.82)	1023.24 (380.82)	723 (271.4)
		100	2000.28 (0.42)	917.42 (596.49)	1427.69 (139.69)	1024 (90.6)
200	10	300	934.55 (428.66)	337.16 (442.36)	646.12 (255.69)	368 (141.1)
		250	1792.1 (353.35)	354.84 (362.0)	1062.81 (205.64)	657 (167.6)
		200	2000.47 (0.9)	571.71 (571.04)	1198.26 (109.66)	763 (104.5)

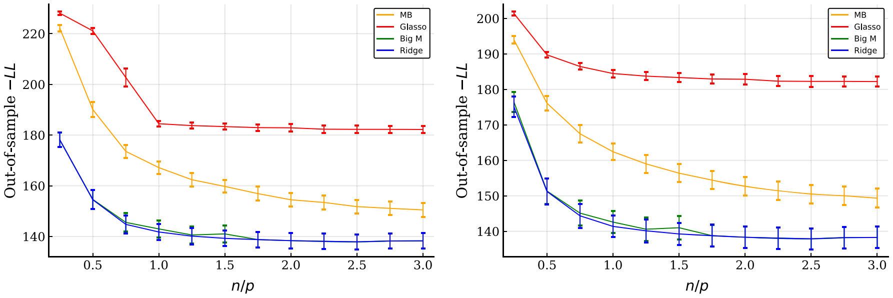
All problems are solved to a tolerance gap of 10^{-4}, where the tolerance gap is the percentage difference between the final lower and upper bounds. Title ver-time and opt-time refer to the time (in s) it takes to verify optimality and to find the optimal solution respectively, cut-time refers to the amount of time spent solving the separation problems, and laz-cons refers to the number of lazy constraints generated. We report average time over 10 random instances (and standard deviation)

D Additional comparisons on statistical performance

We report here additional results from the experiments conducted in Sect. 5.1.

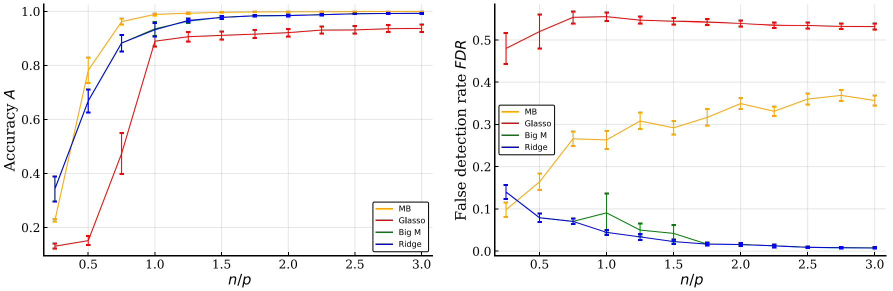
D.1 Comparisons for varying sample sizes *n*/*p*

See Figs. 6 and 7.



(a) $BIC_{1/2}$ as a CV criterion. **(b)** $-LL$ as a CV criterion.

Fig. 6 Impact of the number of samples n/p on out-of-sample negative log-likelihood. Results are averaged over 10 instances with $p = 200$, $t = 1\%$

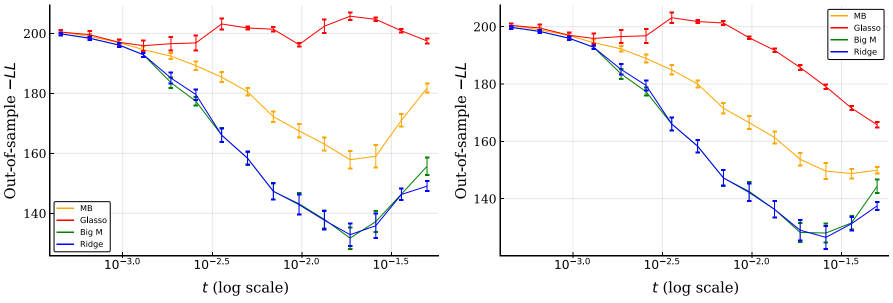


(a) Accuracy A vs. n/p. **(b)** False detection rate FDR vs. n/p.

Fig. 7 Impact of the number of samples n/p on support recovery. Results are averaged over 10 instances with $p = 200$, $t = 1\%$. Hyper-parameters are tuned using the $BIC_{1/2}$ criterion

D.2 Comparisons for varying sparsity levels *t*

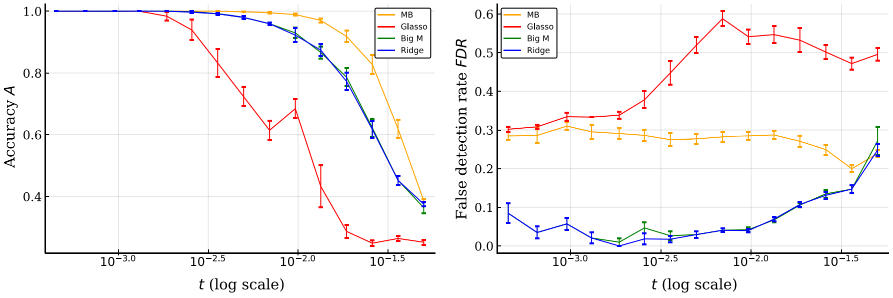
See Figs. 8 and 9.



(a) $BIC_{1/2}$ as a CV criterion.

(b) $-LL$ as a CV criterion.

Fig. 8 Impact of the sparsity level t on out-of-sample negative log-likelihood. Results are averaged over 10 instances with $p = 200$, $n = p$



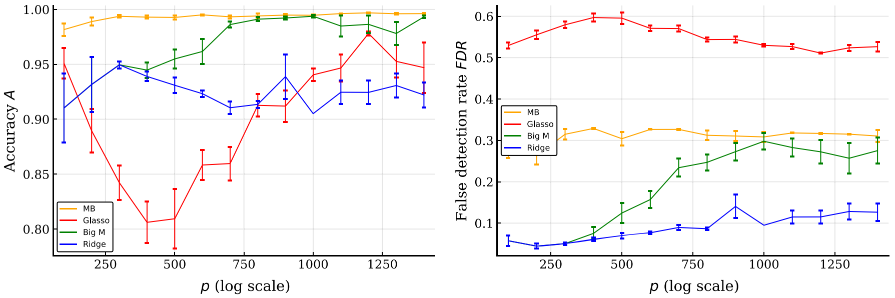
(a) Accuracy A vs. t.

(b) False detection rate FDR vs. t.

Fig. 9 Impact of the sparsity level t on support recovery. Results are averaged over 10 instances with $p = 200$, $n = p$. Hyper-parameters are tuned using the $BIC_{1/2}$ criterion

D.3 Comparisons for varying dimensions *p*

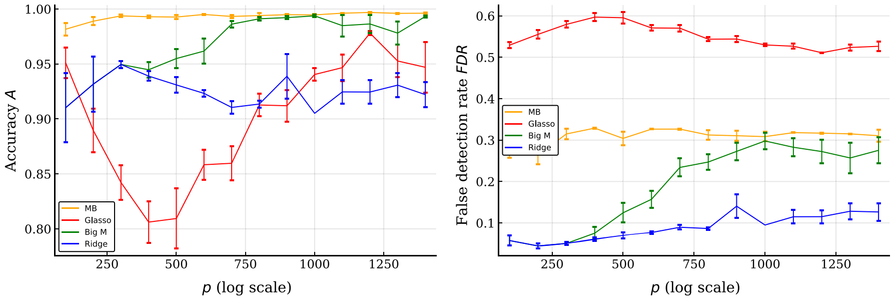
See Figs. 10, 11, 12 and 13.



(a) Accuracy A vs. p.

(b) False detection rate FDR vs. p.

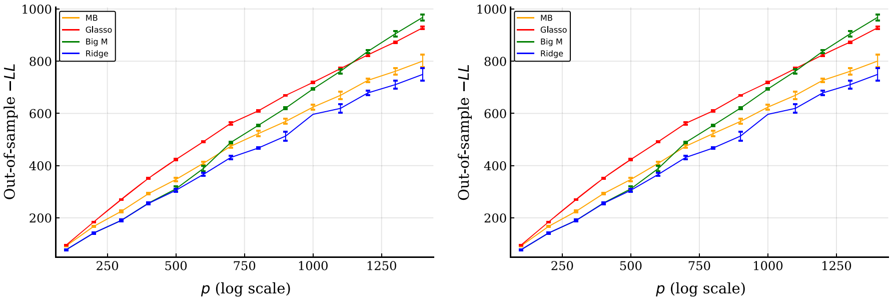
Fig. 10 Impact of the dimension p on support recovery. Results are averaged over 10 instances with $n = p$, $t = 1\%$. Hyper-parameters are tuned using $-LL$



(a) Accuracy A vs. p.

(b) False detection rate FDR vs. p.

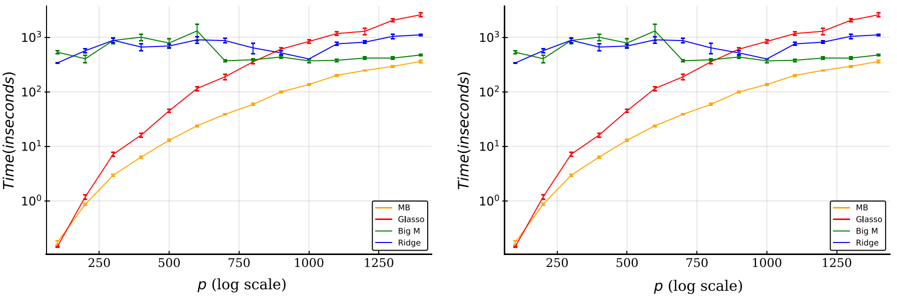
Fig. 11 Impact of the dimension p on support recovery. Results are averaged over 10 instances with $n = p$, $t = 1\%$. Hyper-parameters are tuned using the $BIC_{1/2}$ criterion



(a) $BIC_{1/2}$ as a CV criterion.

(b) $-LL$ as a CV criterion.

Fig. 12 Impact of the dimension p on out-of-sample negative log-likelihood. Results are averaged over 10 instances with $n = p$, $t = 1\%$



(a) $BIC_{1/2}$ as a CV criterion. **(b)** $-LL$ as a CV criterion.

Fig. 13 Impact of the dimension p on computational time. Results are averaged over 10 instances with $n = p$, $t = 1\%$. Recall that discrete formulations big-M and ridge are stopped after 5 min

References

1. Atamtürk, A., Narayanan, V.: Conic mixed-integer rounding cuts. Math. Program. **122**(1), 1–20 (2010)
2. Atchadé, Y.F., Mazumder, R., Chen, J.: Scalable computation of regularized precision matrices via stochastic optimization (2015). arXiv:1509.00426
3. Banerjee, O., El Ghaoui, L., d'Aspremont, A.: Model selection through sparse maximum likelihood estimation for multivariate gaussian or binary data. J. Mach. Learn. Res. **9**(Mar), 485–516 (2008)
4. Ben-Tal, A., El Ghaoui, L., Nemirovski, A.: Robust Optimization. Princeton University Press, Princeton (2009)
5. Bertsimas, D., Copenhaver, M.S.: Characterization of the equivalence of robustification and regularization in linear and matrix regression. Eur. J. Oper. Res. **270**, 93142 (2018)
6. Bertsimas, D., Mazumder, R.: Least quantile regression via modern optimization. Ann. Stat. **42**, 2494–2525 (2014)
7. Bertsimas, D., Van Parys, B.: Sparse high-dimensional regression: Exact scalable algorithms and phase transitions (2017). arXiv:1709.10029
8. Bertsimas, D., Brown, D.B., Caramanis, C.: Theory and applications of robust optimization. SIAM Rev. **53**(3), 464–501 (2011)
9. Bertsimas, D., King, A., Mazumder, R.: Best subset selection via a modern optimization lens. Ann. Stat. **44**(2), 813–852 (2016)
10. Bickel, P.J., Levina, E., et al.: Covariance regularization by thresholding. Ann. Stat. **36**(6), 2577–2604 (2008)
11. Boyd, S., Vandenberghe, L.: Convex Optimization. Cambridge University Press, Cambridge (2004)
12. Cai, T., Liu, W., Luo, X.: A constrained ℓ_1 minimization approach to sparse precision matrix estimation. J. Am. Stat. Assoc. **106**(494), 594–607 (2011)
13. Çezik, M.T., Iyengar, G.: Cuts for mixed 0–1 conic programming. Math. Program. **104**(1), 179–202 (2005)
14. Chickering D.M.:Learning Bayesian Networks is NP-Complete. In: Fisher D., Lenz HJ. (eds.) Learning from Data. Lecture Notes in Statistics, vol 112. Springer, New York, NY (1996)
15. Chow, C., Liu, C.: Approximating discrete probability distributions with dependence trees. IEEE Trans. Inf. Theory **14**(3), 462–467 (1968)
16. Dahl, J., Vandenberghe, L., Roychowdhury, V.: Covariance selection for nonchordal graphs via chordal embedding. Optim. Methods Softw. **23**(4), 501–520 (2008)
17. Dempster, A.P.: Covariance selection. Biometrics **28**, 157–175 (1972)
18. Drton, M., Maathuis, M.H.: Structure learning in graphical modeling. Ann. Rev. Stat. Appl. **4**, 365–393 (2017)
19. Duran, M.A., Grossmann, I.E.: An outer-approximation algorithm for a class of mixed-integer nonlinear programs. Math. Programm. **36**(3), 307–339 (1986)

20. El Karoui, N.: High-dimensionality effects in the markowitz problem and other quadratic programs with linear constraints: risk underestimation. Ann. Stat. **38**(6), 3487–3566 (2010)
21. Fan, J., Li, R.: Variable selection via nonconcave penalized likelihood and its oracle properties. J. Am. Stat. Assoc. **96**(456), 1348–1360 (2001)
22. Fan, J., Fan, Y., Lv, J.: High dimensional covariance matrix estimation using a factor model. J. Econom. **147**(1), 186–197 (2008)
23. Fan, J., Feng, Y., Yichao, W.: Network exploration via the adaptive lasso and scad penalties. Ann. Appl. Stat. **3**(2), 521 (2009)
24. Fan, J., Zhang, J., Ke, Y.: Vast portfolio selection with gross-exposure constraints. J. Am. Stat. Assoc. **107**(498), 592–606 (2012)
25. Fan, J., Han, F., Liu, H.: Challenges of big data analysis. Natl. Sci. Rev. **1**(2), 293–314 (2014)
26. Fan, J., Liao, Y., Liu, H.: An overview of the estimation of large covariance and precision matrices. Econom. J. **19**(1), C1–C32 (2016)
27. Fattahi, S., Sojoudi, S.: Graphical lasso and thresholding: equivalence and closed-form solutions (2017). arXiv:1708.09479
28. Foygel, R., Drton, M.: Extended Bayesian information criteria for Gaussian graphical models. In: Lafferty, J.D., Williams, C.K.I., Shawe-Taylor, J., Zemel, R.S., Culotta, A., (eds.) Advances in Neural Information Processing Systems 23, pp. 604–612. Curran Associates, Inc (2010)
29. Friedman, J., Hastie, T., Tibshirani, R.: Sparse inverse covariance estimation with the graphical lasso. Biostatistics **9**(3), 432–441 (2008)
30. Gally, T., Pfetsch, M.E., Ulbrich, S.: A framework for solving mixed-integer semidefinite programs. Optim. Methods Softw. **33**(3), 594–632 (2018)
31. Geoffrion, A.M.: Generalized benders decomposition. J. Optim. Theory Appl. **10**(4), 237–260 (1972)
32. Gurobi Optimization Inc. Gurobi Optimizer Reference Manual (2015). http://www.gurobi.com
33. Hastie, T., Tibshirani, R., Wainwright, M.: Statistical Learning with Sparsity: The Lasso and Generalizations. CRC Press, Boca Raton (2015)
34. Helmberg, C., Rendl, F.: Solving quadratic (0, 1)-problems by semidefinite programs and cutting planes. Math. Program. **82**(3), 291–315 (1998)
35. Hess, K.R., Keith Anderson, W., Symmans, F., Valero, V., Ibrahim, N., Mejia, J.A., Booser, D., Theriault, R.L., Buzdar, A.U., Dempsey, P.J., et al.: Pharmacogenomic predictor of sensitivity to preoperative chemotherapy with paclitaxel and fluorouracil, doxorubicin, and cyclophosphamide in breast cancer. J. Clin. Oncol. **24**(26), 4236–4244 (2006)
36. Hsieh, C.-J., Dhillon, I.S., Ravikumar, P.K., Sustik,M.A.: Sparse inverse covariance matrix estimation using quadratic approximation. In: Shawe-Taylor, J., Zemel, R. S., Bartlett, P. L., Pereira, F., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems 24, pp. 2330–2338. Curran Associates, Inc (2011)
37. Hsieh, C.-J., Sustik, M.A., Dhillon, I.S., Ravikumar, P.K., Poldrack, R.: Big & quic: sparse inverse covariance estimation for a million variables. In: Burges, C.J.C., Bottou, L., Welling, M., Ghahramani, Z., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems 26, pp. 3165–3173. Curran Associates, Inc (2013)
38. IBM ILOG. Cplex optimizer (2012). http://www-01.ibm.com/software/commerce/optimization/cplex-optimizer
39. Krishnamurthy, V., Ahipasaoglu, S.D., d'Aspremont, A.: A pathwise algorithm for covariance selection. Optim. Mach. Learn., p. 479 (2011). arXiv:0908.0143v2
40. Lam, C., Fan, J.: Sparsistency and rates of convergence in large covariance matrix estimation. Ann. Stat. **37**(6B), 4254 (2009)
41. Lauritzen, S.L.: Graphical Models, vol. 17. Clarendon Press, Oxford (1996)
42. Liu, Z., Lin, S., Deng, N., McGovern, D.P.B., Piantadosi, S.: Sparse inverse covariance estimation with ℓ_0 penalty for network construction with omics data. J. Comput. Biol. **23**(3), 192–202 (2016)
43. Lubin, M., Dunning, I.: Computing in operations research using julia. INFORMS J. Comput. **27**(2), 238–248 (2015)
44. Lubin, M., Yamangil, E., Bent, R., Vielma, J.P.: Polyhedral approximation in mixed-integer convex optimization. Math. Program. **172**(1–2), 139–168 (2018)
45. Ma, J., Zhao, F., Xu, J.: Structure learning constrained by node-specific degree distribution. In: Meila, M., Heskes, T. (eds.) Uncertainty in Artificial Intelligence 33, pp. 533–541. AUAI Press Corvalis, Oregon (2015)

46. Marjanovic, G., Hero, A.O.: ℓ_0 sparse inverse covariance estimation. IEEE Trans. Signal Process. **63**(12), 3218–3231 (2015)
47. Kipp Martin, R.: Using separation algorithms to generate mixed integer model reformulations. Oper. Res. Lett. **10**(3), 119–128 (1991)
48. Mazumder, R., Hastie, T.: Exact covariance thresholding into connected components for large-scale graphical lasso. J. Mach. Learn. Res. **13**(Mar), 781–794 (2012a)
49. Mazumder, R., Hastie, T.: The graphical lasso: new insights and alternatives. Electron. J. Stat. **6**, 2125 (2012b)
50. Meinshausen, N., Bühlmann, P., et al.: High-dimensional graphs and variable selection with the lasso. Ann. Stat. **34**(3), 1436–1462 (2006)
51. Meyer, C.D.: Matrix analysis and applied linear algebra, vol. 71. Siam (2000)
52. Oztoprak, F., Nocedal, J., Rennie, S., Olsen, P.A.: Newton-like methods for sparse inverse covariance estimation. In: Pereira, F., Burges, C.J.C., Bottou, L., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems 25, pp. 755–763. Curran Associates, Inc (2012)
53. Rendl, F., Rinaldi, G., Wiegele, A.: Solving max-cut to optimality by intersecting semidefinite and polyhedral relaxations. Math. Program. **121**(2), 307 (2010)
54. Rigollet, P., Tsybakov, A.: Estimation of covariance matrices under sparsity constraints (2012). arXiv:1205.1210
55. Rothman, A.J., Bickel, P.J., Levina, E., Zhu, J., et al.: Sparse permutation invariant covariance estimation. Electron. J. Stat. **2**, 494–515 (2008)
56. Santhanam, N.P., Wainwright, M.J.: Information-theoretic limits of selecting binary graphical models in high dimensions. IEEE Trans. Inf. Theory **58**(7), 4117–4134 (2012)
57. Scheinberg, K., Rish, I.: SINCO-a greedy coordinate ascent method for sparse inverse covariance selection problem. IBM T. J. Watson Research Center, Yorktown Heights, NY, 10598, July (2009)
58. Scheinberg, K., Ma, S., Goldfarb, D.: Sparse inverse covariance selection via alternating linearization methods. In: Advances in Neural Information Processing Systems, pp. 2101–2109 (2010)
59. Shao, J.: Linear model selection by cross-validation. J. Am. Stat. Assoc. **88**(422), 486–494 (1993)
60. Sotirov, R.: SDP relaxations for some combinatorial optimization problems. In: Handbook on Semidefinite, Conic and Polynomial Optimization, pp. 795–819. Springer (2012)
61. Tan, K.M., London, P., Mohan, K., Lee, S.-I., Fazel, M., Witten, D.: Learning graphical models with hubs. J. Mach. Learn. Res. **15**(1), 3297–3331 (2014)
62. Tokuda, T., Goodrich, B., Van Mechelen, I., Gelman, A., Tuerlinckx, F.: Visualizing distributions of covariance matrices. Columbia University, New York, USA, Technical Report, p. 18 (2011)
63. Vandenberghe, L., Andersen, M.S., et al.: Chordal graphs and semidefinite optimization. Found. Trends Optim. **1**(4), 241–433 (2015)
64. Xu, H., Caramanis, C., Mannor, S.: Robust regression and lasso. In: Advances in Neural Information Processing Systems, pp. 1801–1808 (2009)
65. Yonekura, K., Kanno, Y.: Global optimization of robust truss topology via mixed integer semidefinite programming. Optim. Eng. **11**(3), 355–379 (2010)
66. Yuan, M.: High dimensional inverse covariance matrix estimation via linear programming. J. Mach. Learn. Res. **11**(Aug), 2261–2286 (2010)
67. Yuan, M., Lin, Y.: Model selection and estimation in the gaussian graphical model. Biometrika **94**(1), 19–35 (2007)
68. Zhang, C.-H., et al.: Nearly unbiased variable selection under minimax concave penalty. Ann. Stat. **38**(2), 894–942 (2010)
69. Zhang, F.: The Schur Complement and Its Applications, vol. 4. Springer, Berlin (2006)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.