Learning High-dimensional Gaussian Graphical Models under Total Positivity without Adjustment of Tuning Parameters

Yuhao Wang
University of Cambridge
yw505@cam.ac.uk

Uma Roy
Google Research
uma.roy.us@gmail.com

Caroline Uhler
Massachusetts Institute of Technology
cuhler@mit.edu

Abstract

We consider the problem of estimating an undirected Gaussian graphical model when the underlying distribution is multivariate totally positive of order 2 (MTP$_2$), a strong form of positive dependence. Such distributions are relevant for example for portfolio selection, since assets are usually positively dependent. A large body of methods have been proposed for learning undirected graphical models without the MTP$_2$ constraint. A major limitation of these methods is that their structure recovery guarantees in the high-dimensional setting usually require a particular choice of a tuning parameter, which is unknown a priori in real world applications. We here propose a new method to estimate the underlying undirected graphical model under MTP$_2$ and show that it is provably consistent in structure recovery without adjusting the tuning parameters. This is achieved by a constraint-based estimator that infers the structure of the underlying graphical model by testing the signs of the empirical partial correlation coefficients. We evaluate the performance of our estimator in simulations and on financial data.

1 Introduction

Gaining insights into complex phenomena often requires characterizing the relationships among a large number of variables. Gaussian graphical models offer a powerful framework for representing high-dimensional distributions by capturing the conditional dependencies between the variables of interest in the form of a network. These models have been extensively used in a wide variety of domains ranging from speech recognition (Johnson et al., 2012b) to genomics (Kishino and Waddell, 2000) and finance (Wang et al., 2011).

In this paper we consider the problem of learning a Gaussian graphical model under the constraint that the distribution is multivariate totally positive of order 2 (MTP$_2$), or equivalently, that all partial correlations are non-negative. Such models are also known as attractive Gaussian random fields. MTP$_2$ was first studied in (Bølviken, 1982; Fortuin et al., 1971; Karlin and Rinott, 1980, 1983) and later also in the context of graphical models (Fallat et al., 2017; Lauritzen et al., 2019). MTP$_2$ is a strong form of positive dependence, which is relevant for modeling in various applications including phylogenetics or portfolio selection, where the shared ancestry or latent global market variable often lead to positive dependence among the observed variables (Müller and Scarsini, 2005; Zwiernik, 2015).

Due to the explosion of data where the number of variables p is comparable to or larger than the number of samples N, the problem of learning undirected Gaussian graphical models in the high-dimensional setting has been a central topic in machine learning, statistics and optimization. There are two main classes of algorithms for structure estimation for Gaussian graphical models in the high-dimensional setting. A first class of algorithms attempts to explicitly recover which edges exist in the graphical model, for example using conditional independence tests (Anandkumar et al., 2012; Soh and Tatikonda, 2018) or neighborhood selection (Meinshausen and Bühlmann, 2006). A second class of algorithms instead focuses on estimating the precision matrix. The most prominent of these algorithms is *graphical lasso* (Banerjee et al., 2008; Friedman et al., 2008; Ravikumar et al., 2011; Yuan and Lin, 2007), which applies an ℓ_1 penalty to the log-likelihood function to estimate the precision matrix. Other algorithms include linear programming based approaches such as graphical Dantzig (Yuan, 2010) and CLIME (Cai et al., 2011, 2016); optimiza-

Proceedings of the 23$^{\text{rd}}$International Conference on Artificial Intelligence and Statistics (AISTATS) 2020, Palermo, Italy. PMLR: Volume 108. Copyright 2020 by the author(s).

tion with non-convex penalties like (Fan et al., 2009; Lam and Fan, 2009; Loh and Wainwright, 2017); as well as greedy methods like (Johnson et al., 2012a; Shen et al., 2012).

The main limitation of all aforementioned approaches is the requirement of a specific tuning parameter to obtain consistency guarantees in estimating the edges of the underlying graphical model. In most real-world applications, the correct tuning parameter is unknown and difficult to discover. To make the estimate less sensitive to misspecification of tuning parameters, Liu and Wang (2017) and Sun and Zhang (2013) proposed estimating high-dimensional precision matrices using square-root lasso (Belloni et al., 2011) and scaled lasso (Sun and Zhang, 2012) respectively. These estimators have the advantage that their theoretical guarantees do not rely on an unknown tuning parameter, thereby allowing them to consistently estimate precision matrices without tuning parameter adjustment. While the estimated precision matrices from these methods are guaranteed to converge to the true precision matrix, the zero patterns of the estimated matrices are not guaranteed to recover the underlying graph.

The algorithms described above are for learning the underlying undirected graph in *general* Gaussian models. In this paper, we consider the special setting of MTP$_2$ Gaussian models. Several algorithms have been proposed that are able to exploit the additional structure imposed by MTP$_2$ with the goal of obtaining stronger results than for general Gaussian graphical models. In particular, Lauritzen et al. (2019) showed that the MLE exists whenever the sample size $N > 2$ (independent of the number of variables p), which is striking given that $N > p$ is required for the MLE to exist in general Gaussian graphical models. Since the MLE under MTP$_2$ is not a consistent estimator for the structure of the graph (Slawski and Hein, 2015), Slawski and Hein (2015) considered applying thresholding to entries in the MLE, but this procedure requires a tuning parameter and does not have consistency guarantees.

The three main contributions of this paper are:

1) we provide a new algorithm for learning Gaussian graphical models under MTP$_2$ that is based on conditional independence testing;

2) we prove that this algorithm does not require adjusting any tuning parameters for the theoretical consistency guarantees in structure recovery;

3) we show that our algorithm compares favorably to other methods for learning graphical models on both simulated data and financial data.

2 Preliminaries and Related Work

Gaussian graphical models: Given a graph $G = ([p], \mathcal{E})$ with vertex set $[p] = \{1, \cdots, p\}$ and edge set $\mathcal{E}$ we associate to each node i in G a random variable X_i. A distribution $\mathbf{P}$ on the nodes $[p]$ forms an *undirected graphical model* with respect to G if

$$X_i \perp\!\!\!\perp X_j \mid X_{[p]\setminus\{i,j\}} \quad \text{for all } (i,j) \notin E. \qquad (1)$$

When $\mathbf{P}$ is Gaussian with mean zero, covariance matrix Σ and precision matrix $\Theta := \Sigma^{-1}$, the setting we concentrate on in this paper, then (1) is equivalent to $\Theta_{ij} = 0$ for all $(i,j) \notin E$. By the Hammersley-Clifford Theorem, for strictly positive densities such as the Gaussian, (1) is equivalent to

$$X_i \perp\!\!\!\perp X_j \mid X_S \quad \text{for all } S \subseteq [p]\setminus\{i,j\} \text{ that separate } i,j,$$

where i,j are separated by S in G when i and j are in different connected components of G after removing the nodes S from G. In the Gaussian setting, $X_i \perp\!\!\!\perp X_j \mid X_S$ if and only if the corresponding *partial correlation coefficient* $\rho_{ij|S}$ is zero, which can be calculated from submatrices of Σ, namely

$$\rho_{ij|S} = -\frac{((\Sigma_{M,M})^{-1})_{i,j}}{\sqrt{((\Sigma_{M,M})^{-1})_{i,i}((\Sigma_{M,M})^{-1})_{j,j}}}, \\ \text{where} M = S \cup \{i,j\}.$$

MTP$_2$ distributions: A density function f on $\mathbb{R}^p$ is MTP$_2$ if

$$f(x)f(y) \le f(x \wedge y)f(x \vee y) \quad \text{for all } x, y \in \mathbb{R}^p,$$

where $\vee, \wedge$ denote the coordinate-wise minimum and maximum respectively (Fortuin et al., 1971; Karlin and Rinott, 1980). In particular, a Gaussian distribution is MTP$_2$ if and only if its precision matrix Θ is an M-matrix, i.e. $\Theta_{ij} \le 0$ for all $i \neq j$ (Bølviken, 1982; Karlin and Rinott, 1983). This implies that all partial correlation coefficients are non-negative, i.e., $\rho_{ij|S} \ge 0$ for all i,j,S (Karlin and Rinott, 1983). In addition, for MTP$_2$ distributions it holds that $X_i \perp\!\!\!\perp X_j \mid X_S$ if and only if i,j are separated in G given S (Fallat et al., 2017). Hence i,j are connected in G given S if and only if $\rho_{ij|S} > 0$.

MTP$_2$ distributions are relevant for various applications. In particular, Gaussian tree models with latent variables are MTP$_2$ up to sign (Lauritzen et al., 2019); this includes the important class of single factor analysis models. As an example, in (Slawski and Hein, 2015) MTP$_2$ was used for data measuring students' performance on different math subjects, an application where a factor analysis model with a single latent factor measuring general mathematical ability seems fitting. In addition, factor analysis models are used frequently in psychology and finance; the

MTP_2 constraint has been applied to a dataset from psychology in (Lauritzen et al., 2019) and auctions in (Hubbard et al., 2012). MTP_2 was also used in the modelling of global stock prices, motivated by the fact that asset price changes are usually positively correlated (Agrawal et al., 2019); in particular, the authors reported that the correlation matrix of the daily returns of 5 global stocks is an inverse M-matrix (Agrawal et al., 2019, Figure 1). In the same paper, the authors also showed that using a covariance matrix among stocks estimated under MTP_2 achieves better performance at portfolio selection than other state-of-the-art methods.

Algorithms for learning Gaussian graphical models: An algorithm is called *consistent* if the estimated graph converges to the true graph G as the sample size N goes to infinity. *CMIT*, an algorithm proposed in (Anandkumar et al., 2012), is most related to the approach in this paper. Starting in the complete graph, edge (i, j) is removed if there exists $S \subseteq [p] \setminus \{i, j\}$ with $|S| \leq \eta$ (for a tuning parameter η that represents the maximum degree of the underlying graph) such that the corresponding empirical partial correlation coefficient satisfies $|\hat{\rho}_{ij|S}| \leq \lambda_{N,p}$. For consistent estimation, the tuning parameter $\lambda_{N,p}$ needs to be selected carefully depending on the sample size N and number of nodes p. Intuitively, if $(i, j) \notin G$, then $\rho_{ij|S} = 0$ for all S that separate (i, j). Since $\hat{\rho}_{ij|S}$ concentrates around $\rho_{ij|S}$, it holds with high probability that there exists $S \subseteq [p]\setminus\{i, j\}$ for which $|\hat{\rho}_{ij|S}| \leq \lambda_{N,p}$, so that edge (i, j) is removed from G. Other estimators such as graphical lasso (Ravikumar et al., 2011) and neighborhood selection (Meinshausen and Bühlmann, 2006) also require a tuning parameter: $\lambda_{N,p}$ represents the coefficient of the ℓ_1 penalty and critically depends on N and p for consistent estimation. Finally, with respect to estimation specifically under the MTP_2 constraint, the authors in (Slawski and Hein, 2015) propose thresholding the MLE $\widehat{\Omega}$ of the precision matrix, which can be obtained by solving the following convex optimization problem:

$$\widehat{\Omega} := \min_{\Omega \succeq 0,\ \Omega_{ij} \leq 0\ \forall i \neq j} -\log\det(\Omega) + \operatorname{trace}(\Omega\hat{\Sigma}), \qquad (2)$$

where $\hat{\Sigma}$ is the sample covariance matrix. The threshold quantile q is a tuning parameter, and apart from empirical evidence that thresholding works well, there are no known theoretical consistency guarantees for this procedure.

In addition to relying on a specific tuning parameter for consistent estimation, existing estimators require additional conditions with respect to the underlying distribution. The consistency guarantees of graphical lasso (Ravikumar et al., 2011) and moment matching approaches such as CLIME (Cai et al., 2011) require that the diagonal elements of Σ are upper bounded by a constant and that the minimum edge weight $\min_{i \neq j, \Theta_{ij} \neq 0} |\Theta_{ij}| \geq C\sqrt{\log(p)/N}$ for some positive constant C. Consistency of CMIT (Anandkumar et al., 2012) also requires the minimum edge weight condition. Consistency of CLIME requires a bounded matrix L_1 norm of the precision matrix Θ, which implies that all diagonal elements of Θ are bounded.

Learning a precision matrix without adjusting any tuning parameters: Another recent line of work similar to ours considers estimating high-dimensional Gaussian precision matrices without the tuning of parameters. The most prominent such approach is TIGER (Liu and Wang, 2017) and related works include scaled and organic lasso (Sun and Zhang, 2012; Yu and Bien, 2019). These estimators have the desirable property that the estimated precision matrix $\hat{\Theta}$ is guaranteed to converge to the true Θ without requiring any adjustment of the regularization parameter. However, the support of the estimated $\hat{\Theta}$ is not guaranteed to converge to the underlying graph G (see e.g. Theorem 4.3 of (Liu and Wang, 2017)), which is the particular task we are interested in this paper.

3 Algorithm and Consistency Guarantees

Algorithm 1 is our proposed procedure for learning a Gaussian graphical model under the MTP_2 constraint. In the following, we first describe Algorithm 1 in detail and then prove its consistency without the need of performing any adjustment of tuning parameters.

Similar to CMIT (Anandkumar et al., 2012), Algorithm 1 starts with the fully connected graph $\hat{G}$ and sequentially removes edges based on conditional independence tests. The algorithm iterates with respect to a parameter ℓ that starts at $\ell = 0$. In each iteration, for all pairs of nodes i, j such that the edge $(i, j) \in \hat{G}$ and node i has at least ℓ neighbors (denoted by $\text{adj}_i(\hat{G})$), the algorithm considers all combinations of subsets S of $\text{adj}_i(\hat{G})$ excluding j that have size ℓ and all nodes $k \neq i, j$ that are not in S. For each combination of subset S and node k, it calculates the empirical partial correlation coefficient $\hat{\rho}_{ij|S\cup\{k\}}$. Importantly, $\hat{\rho}_{ij|S\cup\{k\}}$ is calculated only on a *subset* (which we refer to as a *batch*) of size $M := N^{\gamma}$ that we draw randomly from the N samples. If any of these empirical partial correlation coefficients are negative, then edge $i - j$ is deleted from $\hat{G}$ (and no further tests are performed on (i, j)). Each iteration of the algorithm increases ℓ by 1

```
Algorithm 1 Structure learning under total positivity
Input: Matrix of observations $\hat{X} \in \mathbf{R}^{N\times p}$ with sample size $N$ on $p$ nodes.
Output: Estimated graph $\hat{G}$.
Set $\hat{G}$ as the completely connected graph over the vertex set $[p]$; set $\ell := -1$;
repeat
    set $\ell = \ell + 1$;
    repeat
        select a (new) ordered pair $(i, j)$ that are adjacent in $\hat{G}$ and such that $|\mathrm{adj}_i(\hat{G}) \setminus \{j\}| \geq \ell$;
        repeat
            choose a (new) subset $S \subseteq \mathrm{adj}_i(\hat{G}) \setminus \{j\}$ with $|S| = \ell$ and then choose a (new) node $k \in [p] \setminus S \cup \{i, j\}$;
            calculate the empirical partial coefficient $\hat{\rho}_{ij|S\cup\{k\}}$ using randomly drawn data with batch size $M := N^{\gamma}$; if $\hat{\rho}_{ij|S\cup\{k\}} < 0$, delete $i - j$ from $\hat{G}$;
        until edge $i - j$ is deleted from $\hat{G}$ or all $S$ and $k$ are considered;
    until all ordered pairs $i, j$ that are adjacent in $\hat{G}$ with $|\mathrm{adj}_i(\hat{G}) \setminus \{j\}| \geq \ell$ are considered;
until for each $i, j$, $\mathrm{adj}_i(\hat{G}) \setminus \{j\} < \ell$.
```

and the algorithm terminates when for all nodes i, j such that $(i, j) \in \hat{G}$, the neighborhood of i excluding j has size strictly less than ℓ.

The basic intuition behind Algorithm 1 is that if there is an edge $i - j$ in G, then all partial correlations $\rho_{ij|S}$ are positive because of the basic properties of MTP$_2$. In the limit of large N, this implies that all $\hat{\rho}_{ij|S}$ are positive. On the other hand, when i and j are not connected in the true underlying graph, then there exists a list of conditioning sets $S_1, \cdots, S_K$ such that $\rho_{ij|S_k} = 0$ for all $1 \leq k \leq K$. When K is large enough, then intuitively there should exist $1 \leq k \leq K$ such that $\hat{\rho}_{ij|S_k} < 0$ with high probability. However, since for overlapping conditioning sets the empirical partial correlations are highly correlated, we use separate batches of data for their estimation. This leads to a procedure for learning the underlying Gaussian graphical model by deleting edges based on the signs of empirical partial correlation coefficients.

Having provided the high level intuition behind Algorithm 1, we now prove its consistency under common assumptions on the underlying data generating process. Let d denote the maximum degree of the true underlying graph G. For any positive semidefinite matrix A, let $\lambda_{\min}(A)$ and $\lambda_{\max}(A)$ denote the minimum and maximum eigenvalues of A respectively.

Condition 3.1. *There exist positive constants $\sigma_{\min}$ and $\sigma_{\max}$ such that for any subset of nodes $S \subseteq [p]$ with $|S| \leq d+4$, the true underlying covariance matrix satisfies*

$$\lambda_{\min}(\Sigma_S) \geq \sigma_{\min} \quad \textit{and} \quad \lambda_{\max}(\Sigma_S) \leq \sigma_{\max}.$$

Note that since $\lambda_{\max}(\Sigma_S) \leq \mathrm{trace}(\Sigma_S)$ and $|S| \leq d+4$, it is straightforward to show that a sufficient condition for $\lambda_{\max}(\Sigma_S) \leq \sigma_{\max}$ is that all diagonal entries of Σ scale as a constant. This condition is also required by many existing methods including graphical lasso and CLIME; see Section 2.

Similarly, a sufficient condition for $\lambda_{\min}(\Sigma_S) \geq \sigma_{\min}$ is that all diagonal entries of Θ scale as a constant (see the Supplementary Material for a proof); this assumption is also required by CLIME.

Condition 3.2. *There exists a positive constant c_ρ such that for any two nodes $i, j \in [p]$, if $(i, j) \in G$, then $\rho_{i,j|[p]\setminus\{i,j\}} \geq c_\rho \sqrt{(\log p)/(N^{3/4})}$.*

Condition 3.2 is a standard condition for controlling the minimum edge weight in G as required, for example, by graphical lasso. While the minimum threshold in our condition scales as $\sqrt{(\log p)/(N^{3/4})}$, graphical lasso only requires $\sqrt{(\log p)/N}$ (but instead requires a particular choice of tuning parameter and the incoherence condition).

Condition 3.3. *The size of p satisfies that $p \geq N^{\frac{1}{8}} + d + 2$.*

Condition 3.3 implies that the high-dimensional consistency guarantees of Algorithm 1 cannot be directly generalized to the low-dimensional setting where p scales as a constant. We now provide the main result of our paper, namely consistency of Algorithm 1.

Theorem 3.4. *Assume that the maximum neighbourhood size d scales as a constant and let Conditions 3.1-3.3 be satisfied with c_ρ sufficiently large. Then for any $\gamma \in (\frac{3}{4}, 1)$, there exist positive constants τ and C that depend on $(c_\rho, \sigma_{\max}, \sigma_{\min}, d, \gamma)$ such that with probability at least $1 - p^{-\tau} - p^2 e^{-CN^{\frac{1-\gamma}{2}\wedge(4\gamma-3)}}$, the graph estimated by Algorithm 1 is the same as the underlying graph G.*

Remark 3.1. The consistency guarantees of our algorithm hold for any $\gamma \in (\frac{3}{4}, 1)$. This means that our algorithm does not require tuning of the parameter γ to consistently estimate the underlying graph G. Note that this is in contrast to other methods like graphical lasso or CLIME, where the consistency guarantees require a specific choice of the tuning parameter in the algorithm, which is unknown a priori. This is advantageous, since our algorithm can consistently estimate the graph without running any computationally expensive tuning parameter selection approaches, such as stability selection (Meinshausen and Bühlmann, 2010). By setting $\frac{1-\gamma}{2} = (4\gamma - 3)$, we obtain that the *theoretically optimal* value is $\gamma = 7/9$, as this leads to the best asymptotic rate. However, as seen in Section 4, in practice different values of γ can lead to different results. In particular, higher values of γ empirically lead to removing less edges since the overlap between batches is higher and thus the empirical partial correlation coefficients are more correlated with each other.

Remark 3.2. In applications where domain knowledge regarding the graph sparsity is available, γ can still be tuned to incorporate such knowledge to improve estimation accuracy. We see it as a benefit of our method that a tuning parameter can be used when one has access to domain knowledge, but doesn't have to be tuned in order to obtain consistent estimates, since it is provably consistent for all $\gamma \in (\frac{3}{4}, 1)$.

Proof of Theorem 3.4: In the following, we provide an overview of the proof of our main result. Theorems 3.5 and 3.6 show that at iteration $\ell = d + 1$, the graph $\hat{G}$ estimated by Algorithm 1 is exactly the same as the underlying graph G. The proof is then completed by showing that Algorithm 1 stops exactly at iteration $\ell = d + 1$. All proofs are provided in the Supplementary Material.

We start with Theorem 3.5, which bounds the *false negative rate* of Algorithm 1, i.e. showing that all edges (i, j) in the true graph G are retained.

Theorem 3.5 (False negative rate)**.** *Under Conditions 3.1 and 3.2 and c_ρ sufficiently large, there exists a positive constant τ that depends on $(c_\rho, \sigma_{\max}, \sigma_{\min}, d)$ such that with probability at least $1 - p^{-\tau}$, the graph $\hat{G}$ estimated by Algorithm 1 at iteration $\ell = d+1$ contains all edges $(i, j) \in G$.*

The proof of Theorem 3.5 is based on concentration inequalities in estimating partial correlation coefficients. The high-level intuition behind the proof is that because the empirical partial correlation coefficients concentrate exponentially around the true partial correlation coefficients, then with high probability if an edge exists, no empirical partial correlation coefficient will be negative; as a consequence, Algorithm 1 will not eliminate the edge.

The following theorem bounds the *false positive rate*; namely, it shows that with high probability Algorithm 1 will delete all edges (i, j) that are not in the true graph G.

Theorem 3.6 (False positive rate)**.** *Under the same conditions as Theorem 3.4, there exists positive constants C, τ that depend on $(c_\rho, \sigma_{\max}, \sigma_{\min}, d, \gamma)$ such that with probability at least $1 - p^{-\tau} - p^2 e^{-C \frac{1-\gamma}{2} \wedge 4\gamma - 3}$, the graph $\hat{G}$ estimated by Algorithm 1 at iteration $\ell = d + 1$ does not contain any edges $(i, j) \notin G$.*

The proof of Theorem 3.6 relies heavily on the following lemma that considers the orthant probability of partial correlation coefficients. Recall in Algorithm 1 that for a particular edge $i - j$ in the estimated graph $\hat{G}$ at a given iteration, we calculate a series of empirical partial correlation coefficients with different conditioning sets. The only way Algorithm 1 will not delete the edge is if all empirical partial correlation coefficients are ≥ 0. Thus given 2 nodes i, j for which $(i, j) \notin G$, we need to upper bound the orthant probability that all empirical partial correlation coefficients computed by Algorithm 1 are non-negative. As we will discuss next, the use of batches is critical for this result.

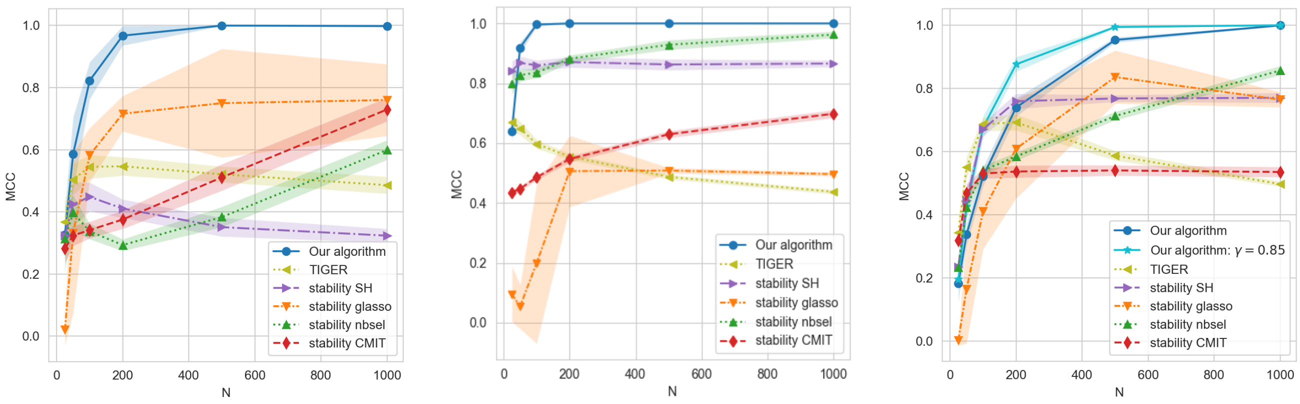
Lemma 3.7. *Consider a pair of nodes $(i, j) \notin G$. Assume that there exists $K := N^{\frac{1-\gamma}{2}}$ sets of nodes $S_1, \cdots, S_K \subseteq [p] \setminus \{i, j\}$ with $|S_k| \leq d + 2$ that satisfy $\rho_{ij|S_k} = 0$. Then there exists positive constants C and N_0 that depends on $(\sigma_{\max}, \sigma_{\min}, d)$ such that*

$$\Pr(\hat{\rho}_{ij|S_k} > 0 \quad \forall k \in [K]) \leq \exp(-C N^{\frac{1-\gamma}{2} \wedge 4\gamma - 3}). \tag{3}$$

To provide intuition for the proof of Lemma 3.7, consider a scenario where the batch size M is chosen small enough such that the batches used to estimate the different $\hat{\rho}_{ij|S_k}$'s have no overlap. Since in this case all $\hat{\rho}_{ij|S_k}$'s are independent, the bound in Lemma 3.7 can easily be proven, namely: for some positive constant $\delta < 1$, it holds that

$$\begin{aligned} \Pr(\hat{\rho}_{ij|S_k} > 0 \quad \forall k \in [K]) &= \prod_{k=1}^{K} \Pr(\hat{\rho}_{ij|S_k} > 0) \\ &\leq \delta^K = \exp\big(- \log(1/\delta) \cdot N^{\frac{1-\gamma}{2}} \big). \end{aligned}$$

However, for small batch size M the empirical partial correlation coefficients $\hat{\rho}_{ij|S}$ don't concentrate around $\rho_{ij|S}$, which may result in false negatives. In the proof of Lemma 3.7 we show that choosing a batch size of $M = N^\gamma$ guarantees the required concentration result as well as a sufficiently weak dependence among the empirical partial correlation coefficients $\hat{\rho}_{ij|S_k}$'s to obtain the exponential upper bound in (3) as in the independent case. Lemma 3.7 implies Theorem 3.6



(a) Random graphs

(b) Chain graphs

(c) Grid graphs

Figure 1: Comparison of different algorithms evaluated on MCC across (a) random, (b) chain, (c) grid graphs with $p = 100$ and $N \in \{25, 50, 100, 200, 500, 1000\}$. For each graph and choice of p and N, results are shown as an average across 20 trials. The shaded areas correspond to ± 1 standard deviation of MCC over 20 trials.

by taking uniform control over all edges $(i, j) \notin G$. Finally, to complete the proof of Theorem 3.4, it remains to show that Algorithm 1 terminates at iteration $\ell = d + 1$.

Proof of Theorem 3.4. It follows from Theorem 3.5 and Theorem 3.6 that with probability at least $1 - p^{-\tau} - p^2 e^{-CN^{\frac{1-\gamma}{2} \wedge 4\gamma - 3}}$, the graph estimated by Algorithm 1 at iteration $\ell = d + 1$ is exactly the same as G. Since the maximum degree of G is at most d, it matches the stopping criterion of Algorithm 1. As a consequence, Algorithm 1 terminates at iteration $\ell = d + 1$. □

4 Empirical Evaluation

In the following, we evaluate the performance of our algorithm for structure recovery in MTP$_2$ Gaussian graphical models in the high-dimensional, sparse regime. We first compare the performance of Algorithm 1 to various other methods on synthetically generated datasets and then present an application to graphical model estimation on financial data. The code to reproduce our experimental results is available at `https://github.com/puma314/MTP2-no-tuning-parameter`.

4.1 Synthetic Data

Given a precision matrix $\Theta \in \mathbb{R}^{p \times p}$, we generate N i.i.d. samples $x^{(1)}, \ldots, x^{(N)} \sim \mathcal{N}(0, \Theta^{-1})$. We let $\hat{\Sigma} = \frac{1}{N} \sum_{i=1}^{N} (x^{(i)})(x^{(i)})^T$ denote the sample covariance matrix. To analyze the performance of our algorithm in various scenarios, we vary N for $p = 100$. In addition, we consider three different sparsity patterns in the underlying precision matrix Θ that are similarly considered by Slawski and Hein (2015), namely:

Grid: Let B be the adjacency matrix of a 2d-grid of size $\sqrt{p}$. Let $\delta := 1.05 \cdot \lambda_1(B)$, $\tilde{\Theta} := \delta I - B$ and $\Theta = D\tilde{\Theta}D$, where D is a diagonal matrix such that $\Sigma = \Theta^{-1}$ has unit diagonal entries.

Random: Same as for *grid* above, but with B replaced with a symmetric matrix having 0 diagonal and one percent non-zero off diagonal entries uniform on $[0, 1]$ chosen uniformly at random.

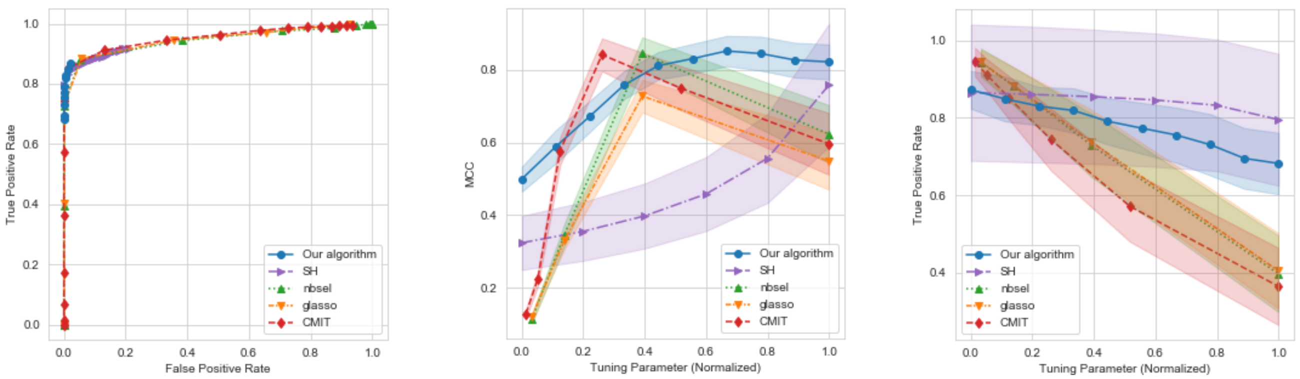
Chain: We let $\Sigma^* := (\sigma^*_{jk}) = (0.9^{|j-k|}), j, k = 1, \ldots, p$. Then we take $\Omega := (\Sigma^*)^{-1}$.

Our primary interest in comparing different algorithms is their performance at recovering the underlying graph structure associated with Θ. Similarly as in (Slawski and Hein, 2015), in Figure 1 we evaluate their performance using Matthew's correlation coefficient (MCC):

$$\text{MCC} = \frac{\text{TP} \cdot \text{TN} - \text{FP} \cdot \text{FN}}{((\text{TP} + \text{FP})(\text{TP} + \text{FN})(\text{TN} + \text{FP})(\text{TN} + \text{FN}))^{1/2}},$$

where TP, TN, FP and FP denote the number of true positives, true negatives, false positives and false negatives respectively. Intuitively, MCC measures the correlation between the presence of edges in the true and estimated graphs. Thus, a higher MCC score means less number of false positives *and* false negatives. Since MCC combines true positive rates (TPR) and false positive rates (FPR), we think it is a compelling metric. MCC has also been used in similar work (Slawski and Hein, 2015). In the appendix, we also provide evaluation results based on TPR and FPR.

Choice of Parameters: We fix $p = 100$ and vary $N = 25, 50, 100, 200, 500, 1000$ to analyze how the ratio p/N



(a) ROC curve

(b) MCC

(c) True positive rate

Figure 2: (a) ROC curves, (b) MCC, and (c) true positive rate versus normalized tuning parameter for random graphs with $p = 100$ and $N = 500$ across 30 trials. The shaded regions correspond to ±1 standard deviation of MCC (TPR resp.) across 30 trials.

affects performance for the various algorithms. For each setup and value of N, we do 20 trials of each algorithm and report the average of the MCCs across the trials.

Methods Compared: We benchmark our algorithm against a variety of state-of-the-art methods for structure learning in Gaussian graphical models (see Section 2) for a range of tuning parameters:

- SH: Slawski and Hein (Slawski and Hein, 2015) considered the same problem as in this paper. For comparison to their algorithm we use the same range of tuning parameters as considered by them, namely $q \in \{0.7, 0.75, 0.8, 0.85, 0.9, 0.95, 0.99\}$.
- *glasso:* For graphical lasso (Friedman et al., 2008) we vary the sparsity parameter around the the theoretically motivated tuning parameter of $\sqrt{\log(p)/n}$, namely $\lambda \in \{0.055, 0.16, 0.45, 1.26, 3.55, 10\}$.
- *nbsel:* For neighborhood selection (Meinshausen and Bühlmann, 2006) we use the same λ values as for *glasso*.
- *TIGER:* For TIGER (Liu and Wang, 2017), we use the theoretically optimal value $\lambda := \pi\sqrt{\frac{\log(p)}{n}}$.
- *CMIT:* This algorithm (Anandkumar et al., 2012) has two tuning parameters. Since the runtime is $p^{\eta+2}$ in the maximal size of the conditioning set η, we set $\eta = 1$ for computational reasons. For λ, we use the the same values as for *glasso*.
- *Our algorithm:* We use the asymptotically optimal choice of $\gamma = 7/9$ (see Remark 3.1) and also compare to $\gamma = 0.85$, which falls in the allowable range $(0.75, 1)$.

For the comparison based on MCC in Figure 1, we use *stability selection* (Meinshausen and Bühlmann, 2010), where an algorithm is run multiple times with different subsamples of the data for each tuning parameter and an edge is included in the estimated graph if it is selected often enough (we used 80%).

Discussion: Figure 1 compares the performance of the various methods based on MCC for random graphs, chain graphs and grid graphs. Compared with the algorithm that has similar theoretical properties as ours, namely TIGER, our algorithm has better overall performance across all simulation set-ups. For the other state-of-the-art methods, Figure 1(a) shows that our algorithm is able to offer a significant improvement for random graphs over competing methods. Also on chain graphs (Figure 1(b)) our algorithm is competitive with the other algorithms, with *SH* and *nbsel* performing comparably. For the grid graph (Figure 1(c)), for $N \leq 200$ *SH* with stability selection outperforms our algorithm with $\gamma = 7/9$. However, it is important to note that stability selection is a major advantage for the compared algorithms and comes at a significant computational cost. Moreover, by varying γ in our algorithm its performance can be increased and becomes competitive to *SH* with stability selection. Both points are discussed in more detail in the Supplementary Material. Another interesting phenomenon is that in Figure 1(c), our algorithm with $\gamma = 0.85$ performs better than the "theoretically optimal" $\gamma = 7/9$, which may seem to contradict our theoretical results. Notice, however, that "theoretical optimality" holds for $N \to \infty$. In the finite sample regime considered here factors such as $\sigma_{\min}$, $\sigma_{\max}$ and d can influence the optimal choice.

To evaluate the sensitivity of the various algorithms to their respective tuning parameters, we generate an ROC curve for each algorithm on random graphs with $p = 100$ and $N \in \{25, 50, 100, 200, 500, 1000\}$, of which $N = 500$ is shown in Figure 2(a); see the Supplementary Material for more details and plots. All algorithms perform similarly in terms of their ROC curves. Note that since our algorithm can only choose γ from the range $(0.75, 1)$, its false positive rate is upper bounded and thus it is impossible to get a full "ROC" curve. Figure 2(b) and (c) show the MCC and true positive rate (TPR) for each algorithm as a function of the tuning parameter normalized to vary between $[0, 1]$. Our algorithm is the least sensitive to variations in the tuning parameter, as it has one of the smallest ranges in both MCC and TPR (the y-axes) as compared to the other algorithms. Our algorithm also shows the smallest standard deviations in MCC and in TPR, showing its consistency across trials (especially compared to *SH*). We here concentrate on TPR since the variation in FPR between all algorithms is small across trials. Taken together, it is quite striking that our algorithm with fixed γ generally outperforms methods with stability selection.

4.2 Application to Financial Data

We now examine an application of our algorithm to financial data. The MTP$_2$ constraint is relevant for such data, since the presence of a latent global market variable leads to positive dependence among stocks (Hennessy and Lapan, 2002; Müller and Scarsini, 2005). We consider the daily closing prices for $p = 452$ stocks that were consistently in the S& P 500 index from January 1, 2003 to January 1, 2018, which results in a sample size of $N = 1257$. Due to computational limitations of stability selection primarily with *CMIT*, we performed the analysis on the first $p = 100$ of the 452 stocks. The 100 stocks are categorized into 10 sectors, known as the Global Industry Classification Standard (GICS) sectors. This dataset is gathered from Yahoo Finance and has also been analyzed in (Liu et al., 2012).

A common task in finance is to estimate the covariance structure between the log returns of stocks. Let $S_j^{(t)}$ denote the closing price of stock j on day t and let $X_j^{(t)} := \log(S_j^{(t)}/S_j^{(t-1)})$ denote the log return of stock j from day $t-1$ to t. Denoting by $X := (X_1, \ldots, X_{100})^T$ the random vector of daily log returns of the 100 stocks in the data set, then our goal is to estimate the undirected graphical model of X. We do this by treating the 1257 data points $X^{(t)} := (X_1^{(t)}, \ldots, X_{100}^{(t)})$ corresponding to the days $t = 1, \ldots, 1257$ as i.i.d. realizations of the random vector X.

As in Section 4.1, we compare our method to *SH*, *glasso* (using both stability selection and cross-validation), *nbsel*, *CMIT* (using both stability selection and the hyperparameter with the best performance) and TIGER. Note that here we cannot assess the performance of the various methods using MCC since the graph structure of the true underlying graphical model is unknown. Instead, we assess each estimated graph based on its *modularity coefficient*, namely the performance at grouping stocks from the same sector together. Table 1 shows that our method using fixed $\gamma = 7/9$ outperforms all other methods in grouping the stocks. For further details on the analysis see the Supplementary Material.

Method	Modularity Coefficient
Our Algorithm ($\gamma = 7./9.$)	0.482
Slawski-Hein with st. sel.	0.418
Neighborhood selection with st. sel.	0.350
Graphical Lasso with st. sel.	0.
Cross-validated graphical lasso	0.253
CMIT with st. sel.	-0.0088
CMIT with best hyperparameter	-0.0085
TIGER	-0.5

Table 1: Modularity scores of the estimated graphs; higher score indicates better clustering performance; "st. sel" stands for "stability selection". For our algorithm we used the theoretically optimal value of $\gamma = 7/9$.

5 Discussion

In this paper, we proposed a tuning-parameter free, constraint-based estimator for learning the structure of the underlying Gaussian graphical model under the constraint of MTP$_2$. We proved consistency of our algorithm in the high-dimensional setting without relying on an unknown tuning parameter. We further benchmarked our algorithm against existing algorithms in the literature with both simulated and real financial data, thereby showing that it outperforms existing algorithms in both settings. A limitation of our algorithm is that its time complexity scales as $O(p^d)$; it would be interesting in future work to develop a more computationally efficient algorithm for graphical model estimation under MTP$_2$. Another limitation is that our algorithm is only provably consistent in the high-dimensional setting. However, the strong empirical performance of our algorithm as compared to existing algorithms is quite striking, given in particular these results are from fixed γ. To our knowledge, this is the first tuning-parameter free algorithm for structure recovery in Gaussian graphical models with consistency guarantees.

Acknowledgements

We thank Dheeraj Nagaraj, Cheng Mao and Philippe Rigollet and the anonymous reviewers for helpful discussions. The authors acknowledge support by NSF (DMS-1651995), ONR (N00014-17-1-2147 and N00014-18-1-2765), IBM, a Sloan Fellowship and a Simons Investigator Award. At the time this research was completed, Yuhao Wang and Uma Roy was at the Massachusetts Institute of Technology.

References

R. Agrawal, U. Roy, and C. Uhler. Covariance matrix estimation under total positivity for portfolio selection. *arXiv preprint arXiv:1909.04222*, 2019.

A. Anandkumar, V. Y. Tan, F. Huang, and A. S. Willsky. High-dimensional Gaussian graphical model selection: Walk summability and local separation criterion. *Journal of Machine Learning Research*, 13 (Aug):2293–2337, 2012.

O. Banerjee, L. E. Ghaoui, and A. d'Aspremont. Model selection through sparse maximum likelihood estimation for multivariate Gaussian or binary data. *Journal of Machine Learning Research*, 9(Mar):485–516, 2008.

A. Belloni, V. Chernozhukov, and L. Wang. Square-root lasso: pivotal recovery of sparse signals via conic programming. *Biometrika*, 98(4):791–806, 2011.

E. Bølviken. Probability inequalities for the multivariate normal with non-negative partial correlations. *Scandinavian Journal of Statistics*, pages 49–58, 1982.

T. Cai, W. Liu, and X. Luo. A constrained ℓ_1 minimization approach to sparse precision matrix estimation. *Journal of the American Statistical Association*, 106(494):594–607, 2011.

T. T. Cai, W. Liu, and H. H. Zhou. Estimating sparse precision matrix: optimal rates of convergence and adaptive estimation. *The Annals of Statistics*, 44 (2):455–488, 2016.

S. Fallat, S. Lauritzen, K. Sadeghi, C. Uhler, N. Wermuth, and P. Zwiernik. Total positivity in Markov structures. *The Annals of Statistics*, 45(3):1152–1184, 2017.

J. Fan, Y. Feng, and Y. Wu. Network exploration via the adaptive LASSO and SCAD penalties. *The Annals of Applied Statistics*, 3(2):521, 2009.

C. M. Fortuin, P. W. Kasteleyn, and J. Ginibre. Correlation inequalities on some partially ordered sets. *Communications in Mathematical Physics*, 22(2): 89–103, 1971.

J. Friedman, T. Hastie, and R. Tibshirani. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3):432–441, 2008.

D. A Hennessy and H. E. Lapan. The use of Archimedean copulas to model portfolio allocations. *Mathematical Finance*, 12(2):143–154, 2002.

T. P. Hubbard, T. Li, and H. J. Paarsch. Semiparametric estimation in models of first-price, sealed-bid auctions with affiliation. *Journal of Econometrics*, 168(1):4–16, 2012.

C. Johnson, A. Jalali, and P. Ravikumar. High-dimensional sparse inverse covariance estimation using greedy methods. In *Artificial Intelligence and Statistics*, pages 574–582, 2012a.

M. Johnson, S. P. Khudanpur, M. Ostendorf, and R. Rosenfeld. *Mathematical Foundations of Speech and Language Processing*, volume 138. Springer Science & Business Media, 2012b.

S. Karlin and Y. Rinott. Classes of orderings of measures and related correlation inequalities. I. Multivariate totally positive distributions. *Journal of Multivariate Analysis*, 10(4):467–498, 1980.

S. Karlin and Y. Rinott. M-matrices as covariance matrices of multinormal distributions. *Linear Algebra and its Applications*, 52:419–438, 1983.

H. Kishino and P. J. Waddell. Correspondence analysis of genes and tissue types and finding genetic links from microarray data. *Genome Informatics*, 11:83–95, 2000.

C. Lam and J. Fan. Sparsistency and rates of convergence in large covariance matrix estimation. *The Annals of Statistics*, 37(6B):4254, 2009.

S. Lauritzen, C. Uhler, and P. Zwiernik. Maximum likelihood estimation in Gaussian models under total positivity. *The Annals of Statistics*, 47:1835––1863, 2019.

H. Liu and L. Wang. Tiger: A tuning-insensitive approach for optimally estimating Gaussian graphical models. *Electronic Journal of Statistics*, 11(1):241–294, 2017.

H. Liu, F. Han, and C. Zhang. Transelliptical graphical models. In *Advances in Neural Information Processing Systems*, pages 800–808, 2012.

P. Loh and M. J. Wainwright. Support recovery without incoherence: A case for nonconvex regularization. *The Annals of Statistics*, 45(6):2455–2482, 2017.

N. Meinshausen and P. Bühlmann. High-dimensional graphs and variable selection with the lasso. *The Annals of Statistics*, 34(3):1436–1462, 2006.

N. Meinshausen and P. Bühlmann. Stability selection. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 72(4):417–473, 2010.

A. Müller and M. Scarsini. Archimedean copulae and positive dependence. *Journal of Multivariate Analysis*, 93(2):434–445, 2005.

P. Ravikumar, M. J. Wainwright, G. Raskutti, and B. Yu. High-dimensional covariance estimation by minimizing ℓ_1-penalized log-determinant divergence. *Electronic Journal of Statistics*, 5:935–980, 2011.

X. Shen, W. Pan, and Y. Zhu. Likelihood-based selection and sharp parameter estimation. *Journal of the American Statistical Association*, 107(497):223–232, 2012.

M. Slawski and M. Hein. Estimation of positive definite M-matrices and structure learning for attractive Gaussian Markov random fields. *Linear Algebra and its Applications*, 473:145–179, 2015.

D. W. Soh and S. Tatikonda. Identifiability in Gaussian graphical models. *arXiv preprint arXiv:1806.03665*, 2018.

T. Sun and C. Zhang. Scaled sparse linear regression. *Biometrika*, 99(4):879–898, 2012.

T. Sun and C. Zhang. Sparse matrix inversion with scaled lasso. *The Journal of Machine Learning Research*, 14(1):3385–3418, 2013.

H. Wang, C. Reeson, and C. M. Carvalho. Dynamic financial index models: Modeling conditional dependencies via graphs. *Bayesian Analysis*, 6(4):639–664, 2011.

G. Yu and J. Bien. Estimating the error variance in a high-dimensional linear model. *Biometrika*, 106(3): 533–546, 2019.

M. Yuan. High dimensional inverse covariance matrix estimation via linear programming. *Journal of Machine Learning Research*, 11(Aug):2261–2286, 2010.

M. Yuan and Y. Lin. Model selection and estimation in the Gaussian graphical model. *Biometrika*, 94(1): 19–35, 2007.

P. Zwiernik. *Semialgebraic Statistics and Latent Tree Models*. Chapman and Hall/CRC, 2015.