GRAPHICAL MODELS AND ITS ESTIMATION IN TIME SERIES ANALYSIS

YUEN TSZ PANG

MPhil

The Hong Kong Polytechnic University

2019

The Hong Kong Polytechnic University

Department of Applied Mathematics

Graphical Models and its Estimation in Time Series Analysis

Yuen Tsz Pang

A THESIS SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS

FOR THE DEGREE OF MASTER OF PHILOSOPHY

May 2018

Certificate of Originality

I hereby declare that this thesis is my own work and that, to the best of my knowledge and belief, it reproduces no material previously published or written, nor material that has been accepted for the award of any other degree or diploma, except where due acknowledgement has been made in the text.

_____ (Signed)

YUEN Tsz Pang (Name of student)

To my parents, my brother, and my friends.

Abstract

Graphical time series models encode the dynamic relationships among the variables in multivariate time series in graphs, in which nodes represent the variables and edges characterize the conditional dependence. In applications, the graph structure is not known in advance, and it is of interest to estimate and determine the graph based on samples. To determine graphical time series models, we propose two estimation methods based on sparse vector autoregressive models.

An alternating maximization method is introduced to estimate sparse vector autoregressive models under sparsity constraints on both the autoregressive coefficients and the inverse noise covariance matrix. This alternating method estimates sparse vector autoregressive models by considering the maximum likelihood estimation with the sparsity constraints as a biconcave problem. Such optimization problem is concave when either the autoregressive coefficients or the inverse covariance matrix is fixed. Simulation experiments study the estimation performance of the alternating method and compare with other non-linear optimization methods. We also introduce two approaches in determining the sparsity constraints. These two methods are studied by simulation studies for comparisons. Real data examples are provided as illustrations.

The sparsity constraints in the alternating maximization method, however, require being identified before the estimation procedure. A penalized likelihood estimation for vector autoregressive models is proposed to encourage sparsity on both the autoregressive coefficients and the inverse noise covariance matrix. This penalization method implements penalty terms on the autoregressive coefficients and the off-diagonal elements of the inverse covariance matrix to achieve parsimonious models. The finite sample properties of the penalized likelihood estimator are investigated by simulation experiments. A real dataset application is presented for demonstration.

Publications arising from the thesis

- Yuen, T. P., Wong, H., & Yiu, K. F. C. (2016, August). *On constrained estimation of graphical time series models.* Paper presented at the 22nd International Conference on Computational Statistics, Oviedo, Spain.

- Yuen, T. P., Wong, H., & Yiu, K. F. C. (2018). On constrained estimation of graphical time series models. *Computational Statistics and Data Analysis*, *124*, 27–52.

Acknowledgements

I would like to express great reverence for my supervisors, Prof. Wong Heung and Prof. Yiu Ka Fai, Cedric, for their guidance, patience, and their continuous support for my study, leading to this work. Besides my advisors, I would like to thank Dr. Ng Chi Tim for his valuable discussions and ingenious suggestions. I also thank the teachers, colleagues, and friends in the department for their kindnesses and assistance. Finally, I want to thank my parents, my brother, and my friends for their encouragement during my study.

Contents

List of Figures

List of Tables

List of Notation

$\mathbb{C}$	The set of complex numbers.
$\mathbb{Z}$	The set of integers.
$\mathbb{N}_0$	The set of non-negative integers.
$\overline{z}$	The complex conjugate of the complex number z.
$\lvert z \rvert$	The absolute value or modulus of z.
$\lvert \mathcal{S} \rvert$	The cardinality of the set $\mathcal{S}$.
a_{ij} or $(\mathbf{A})_{ij}$	The (i,j)-th entry of the matrix $\mathbf{A}$.
a_i	The i-th entry of the vector $\mathbf{a}$.
$\mathbf{I}_K$	$K \times K$ identity matrix.
$\mathbf{A} \succ 0$	$\mathbf{A}$ is positive definite.
$\mathbf{A}^*$	Conjugate transpose of the matrix $\mathbf{A}$.
$\det(\mathbf{A})$ or $\det \mathbf{A}$	Determinant of the matrix $\mathbf{A}$.
$\mathbf{diag}(\mathbf{a})$	Diagonal matrix with vector $\mathbf{a}$ as main diagonal.
$\lVert \mathbf{A} \rVert_F$	Frobenius norm of the matrix $\mathbf{A}$.
$\mathbf{A} \otimes \mathbf{B}$	Kronecker product of $\mathbf{A}$ and $\mathbf{B}$.
$\mathbf{A}^{-1}$	Matrix inverse of the matrix $\mathbf{A}$.
$\operatorname{trace}(\mathbf{A})$ or $\operatorname{trace} \mathbf{A}$	Trace of the matrix $\mathbf{A}$.
$\mathbf{A}^\top$	Transpose of the matrix $\mathbf{A}$.
$\mathbf{vec}(\mathbf{A})$	Vectorization of the matrix $\mathbf{A}$.
$\mathbf{vech}(\mathbf{A})$	Half-vectorization of the symmetric matrix $\mathbf{A}$.
$X \perp\!\!\!\perp Y \lvert Z$	For random variables X, Y, and Z, X and Y are conditionally independent given Z.

Chapter 1

Introduction

1.1 Graphical Models

Probabilistic graphical models connect the concept of conditional independencies among random variables to graph theory by representing the dependencies in a graph. These independence relationships are typically visualized by an undirected graph. Each node in the graph represents a variable, and the absence of an edge between two nodes indicates the corresponding variables are conditionally independent, given the remaining variables. Researchers have also investigated the use of directed graphs to characterize the possible causal relations between variables (Pearl, 1995; Lauritzen & Richardson, 2002). Probabilistic graphical models offer several advantages to analysis complex probabilistic models, including the use of graphical models as a tool to design and motivate new models. Sophisticated probabilistic models require complicated computations to perform inference which can be represented diagrammatically using graphical models. The recent development of deep belief networks takes these advantages of graphical models

to construct new models (Salakhutdinov & Hinton, 2009; Ranganath et al., 2015).

In the situation where the variables are discrete, research on linking up the undirected graphical models with log-linear models for multi-way contingency tables was performed (Wermuth, 1976; Darroch et al., 1980). By analogy with the log-linear models for tables of counts, graphical models for continuous variables based on a multivariate normal distribution are broadly studied. Edwards (1995) and Lauritzen (1996) give a general introduction to graphical models.

Suppose a K-dimensional random variable $\mathbf{X} = (X_1, \cdots, X_K)^\top$ follows a multivariate normal distribution with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$. A Gaussian graphical model can be determined from the inverse covariance matrix $\boldsymbol{\Theta} = \boldsymbol{\Sigma}^{-1}$ (also known as the precision matrix). Knowing that all the conditional distributions of $\mathbf{X}$ are also normal, the inverse covariance matrix incorporates the information about the partial covariances between the variables. That is, the covariances between two variables conditioned on the remaining variables. See Anderson (2003) for details. Given prior information on the conditional independencies between variables, Dempster (1972) formulated the covariance selection problem to estimate Gaussian graphical models from samples by considering the following maximization problem,

$$\begin{aligned} &\text{maximize} \quad \log\det\boldsymbol{\Theta} - \operatorname{trace}(\mathbf{S}\boldsymbol{\Theta}) \\ &\text{subject to} \quad \theta_{ij} = 0, \quad (i,j) \in \mathcal{S}, \end{aligned}$$

where $\mathbf{S}$ is the sample covariance matrix, $\mathcal{S}$ is a set of index pairs of known conditionally independent nodes. This maximization problem can be formulated as a log-determinant maximization problem with linear constraints (Vandenberghe et al., 1998), which can be solved by the interior-point

method based solvers such as SDPT3 (Tütüncü et al., 2003) or SeDuMi (Sturm, 1999). For a large scale covariance selection problem, Dahl et al. (2008) discussed several algorithms, including Newton's method and coordinate descent algorithm, to improve the computation efficiency.

The growing interest in high dimensional data analysis has led to the development of sparse Gaussian graphical models for improving the interpretability. To achieve model sparsity, researchers have proposed the use of penalized likelihood estimation by considering the problem,

$$\begin{aligned} \text{maximize} \quad & \log\det\boldsymbol{\Theta} - \text{trace}\left(\mathbf{S}\boldsymbol{\Theta}\right) \\ \text{subject to} \quad & \rho\left(\boldsymbol{\Theta}\right) \leq k, \end{aligned}$$

where $\mathbf{S}$ is the sample covariance matrix, $\rho\left(\cdot\right)$ is a regularization term, and k is a tuning parameter. Various studies suggest the use of L_1 regularization (Tibshirani, 1996) in the estimation (Yuan & Lin, 2006; d'Aspremont et al., 2008; Li & Toh, 2010). That is, $\rho(\boldsymbol{\Theta}) = \sum_{i,j} |\theta_{ij}|$. Friedman et al. (2008) connected the L_1 regularized problem to the neighbourhood-based Lasso regression approach and called this method graphical Lasso algorithm. Researchers have also applied non-convex penalty functions to the problem to ameliorate the statistical bias issue encountered when using the L_1 penalty (Fan et al., 2009; Rothman et al., 2008). The non-convex penalties include the smoothly clipped absolute deviation (SCAD) penalty (Fan & Li, 2001) and the minimax concave penalty (MCP) (Zhang, 2010).

1.2 Graphical Time Series Models

Brillinger (1996) and Dahlhaus (2000) extended the graphical models from

multivariate random variables to multiple time series, to explore the interrelationships between the series components. Brillinger (1981) discussed the concept of conditional independence could be extended to multivariate time series. This extension, in particular, suggested that the conditional independence between two components of a multivariate Gaussian stationary time series process can be identified by the partial spectral coherence of the time series process. These frequency domain statistics measure the linear association of two series components, given the linear effects of the remaining components (Koopmans, 1995; Brillinger, 1981). Dahlhaus (2000) proposed the use of an undirected conditional correlation graph, similar to the Gaussian graphical models, to visualize the dynamic interrelationships among the series based on the partial spectral coherencies. This graph is particularly called the conditional independence graph (CIG) under normality assumption on the multiple series. The author also discussed a hypothesis test on partial spectral coherence to determine the structure of the conditional correlation graph. Researchers have utilized this approach to study the brain connectivity in neuroscience (Dahlhaus et al., 1997; Medkour et al., 2009).

Oxley et al. (2004) studied the dynamic relationships among variables in multivariate time series by constructing a sparse structural vector autoregressive (SVAR) model and depicted the model by a directed acyclic graph (DAG). To identify a parsimonious graph, the authors applied the partial correlations of variables to determine the CIG and hence estimate a sparse SVAR model based on the graph. The estimation of a sparse SVAR model, however, requires restrictions on the structure of the coefficients matrix such that the model is identifiable. The monograph by Tunnicliffe-Wilson et al. (2015) provided a detailed procedure and illustrations on graphical modeling of structural vector autoregressive models.

Eichler (2012) introduced an alternative approach to analyzing the dynamic relationships among the series components based on ordinary time series models. Such graphical time series models are built on the concept of Granger causality (Granger, 1969) and are encoded by mixed graphs. Each vertex of the graph represents a component series, directed edges are characterized by the possible Granger causal relationships, and undirected edges indicate the contemporaneous dependence structure. Eichler (2012) also discussed the application of such graphical time series models on various multivariate time series models, such as vector autoregressive (VAR) models and multivariate autoregressive conditional heteroscedasticity (ARCH) models. In this thesis, we will consider the estimation of such graphical time series models based on sparse Gaussian VAR models. The autoregressive coefficients characterize the directed edges, and the undirected edges are determined by the non-diagonal elements of the inverse noise covariance matrix.

To estimate sparse VAR models, one can first construct a conditional correlation graph by computing the partial spectral coherencies from the samples. With the conditional correlation graph, we can impose sparsity constraints on the VAR model to reduce model complexity. Songsiri et al. (2009) formulated the problem of maximum likelihood estimation of VAR models subject to conditional independence constraints based on the inverse of the spectral density matrix. The authors proposed a convex relaxation of the problem so that the estimation is done in a tractable way and proved the relaxation is exact when the sample autocovariance matrix is block-Toeplitz.

Davis et al. (2016) presented a two-stage estimation procedure for fitting sparse VAR models by considering the constrained maximum likelihood estimation on VAR models with zero constraints on the autoregressive coefficients. The zero constraints are selected according to the partial

spectral coherencies together with the Bayesian information criterion (BIC) (Schwarz, 1978). The fitted model is then refined to reduce the number of parameters further by using the t-ratios of the estimated autoregressive coefficients in the second stage. Songsiri et al. (2009) and Davis et al. (2016) also explored the penalized likelihood estimation on VAR models, by imposing L_1 regularization on the autoregressive coefficients, to achieve sparsity. Researchers have also discussed the use of penalized regression approach for VAR modeling (Valdés-Sosa et al., 2005; Hsu et al., 2008; Song & Bickel, 2011; Ren et al., 2013; Songsiri, 2013; Jung et al., 2015). The penalized regression method, however, ignores the contemporaneous dependence structure in multivariate time series (Song & Bickel, 2011). This ignorance is because a loss function of the sum of squared residuals is used, which does not consider the noise covariance matrix into account.

To encourage model sparsity, we first consider a constrained maximum likelihood estimation on VAR models with sparsity constraints on both the autoregressive coefficients and the inverse of the noise covariance matrix. These sparsity constraints are predetermined using the conditional correlation graph. An iterative algorithm is proposed to estimate the sparse VAR model by considering the maximum likelihood estimation with the sparsity constraints as a "biconcave" problem. That is, the optimization problem is concave when either the autoregressive coefficients or the inverse noise covariance matrix is fixed (Gorski et al., 2007).

The second strategy to build sparse VAR models is to consider the penalized likelihood estimation on VAR models. The autoregressive coefficients and the off-diagonal elements of the inverse covariance matrix are penalized by penalty functions, such as L_1 (Tibshirani, 1996), SCAD (Fan & Li, 2001) and MCP (Zhang, 2010). We adopt the local linear approximation

(LLA), proposed by Zou & Li (2008), on the penalty functions to solve the penalized estimation problem.

1.3 Outline of the Thesis

In this thesis, we will discuss the estimation of graphical time series models based on the sparse Gaussian vector autoregressive (VAR) processes. Two methods of the estimation for sparse Gaussian VAR models will be presented, namely a constrained likelihood estimation method and a penalized likelihood estimation method.

Chapter 2 reviews the VAR models and its maximum likelihood estimation, as a prelude to the introduction of the two estimation methods. The partial cross-correlations and the spectral analysis of time series will then be delivered, which act as bases for determining the conditional correlation graph. The chapter ends by introducing the graphical time series models, including the conditional correlation graph and the Granger causality graph. With the conditional correlation graph, we can identify the sparsity constraints that are implemented in the constrained likelihood estimation method.

Chapter 3 presents the constrained likelihood estimation method by first giving the problem formulation. We show this problem is biconcave and propose an iterative procedure to solve the problem. The procedures for identifying the sparsity structure are provided. We end this chapter by some numeric results, including simulation studies and real data applications, to illustrate the constrained likelihood estimation method.

Chapter 4 exhibits the penalized likelihood estimation for sparse VAR models. A brief introduction to the penalized estimation methods will be

provided in this chapter followed by the problem formulation. We next present the estimation procedure and investigate the finite sample properties of the penalized likelihood estimator through simulation experiments. We exemplify the penalized likelihood estimation method by a real data application and end this chapter.

Chapter 5 concludes and suggests directions for future research.

Chapter 2

Graphical Time Series Models

Graphical models represent the conditional independence between random variables in multivariate data. Brillinger (1996) gave the first remarks on graphical models for time series which provide a tool to visualize the interrelationships between components of multivariate time series processes. Dahlhaus (2000) applied undirected graphs to depict the conditional correlation structure of multiple time series. Oxley et al. (2004) utilized a directed acyclic graph to represent the dynamic relationships between components of a multivariate time series by considering structural vector autoregressive models. Eichler (2012) encoded the dynamic interdependencies among variables of multiple time series by mixed graphs. Each node represents a component of the series, directed edges are characterized by the possible Granger causal relationships between variables, and undirected edges capture the contemporaneous conditional dependence structure.

The first part of this chapter presents the vector autoregressive (VAR) models and its estimation method. We then review two graphical time series models in the final part of this chapter. The models are the conditional correlation graphs (Dahlhaus, 2000) and the causality graphs (Eichler, 2012).

2.1 Vector Autoregressive Models

Consider a K-dimensional VAR(p) model (VAR model of order p),

$$\mathbf{y}_t = \boldsymbol{\nu} + \mathbf{A}_1 \mathbf{y}_{t-1} + \cdots + \mathbf{A}_p \mathbf{y}_{t-p} + \mathbf{u}_t, \tag{2.1}$$

where $\mathbf{y}_t = (y_{1,t}, \cdots, y_{K,t})^\top$ is a K-dimensional vector, $\mathbf{A}_1, \cdots, \mathbf{A}_p$ are $K \times K$ autoregressive coefficients matrices, $\boldsymbol{\nu}$ is a K-dimensional vector of intercepts, $\mathbf{u}_t = (u_{1,t}, \cdots, u_{K,t})^\top$ is a K-dimensional Gaussian noise vector with mean $\mathbf{0}$ and a $K \times K$ non-singular covariance matrix $\boldsymbol{\Sigma}_u$, and $t = 1, \cdots, T$. We further assume that the process is stable, i.e. $\det\left(\mathbf{I}_K - \sum_{l=1}^{p} \mathbf{A}_l z^l\right) \neq 0$, for $z \in \mathbb{C}$, $|z| \leq 1$, and p pre-sample values, $\mathbf{y}_{-p+1}, \cdots, \mathbf{y}_0$, are available. The compact form of (2.1) is

$$\mathbf{Y} = \mathbf{B}\mathbf{Z} + \mathbf{U},$$

where $\mathbf{Y} = (\mathbf{y}_1, \cdots, \mathbf{y}_T)$, $\mathbf{B} = (\boldsymbol{\nu}, \mathbf{A}_1, \cdots, \mathbf{A}_p)$ is a $K \times (Kp+1)$ matrix containing the coefficients and the intercepts, $\mathbf{Z}_t = \left(1, \mathbf{y}_t^\top, \cdots, \mathbf{y}_{t-p+1}^\top\right)^\top$, $\mathbf{Z} = (\mathbf{Z}_0, \cdots, \mathbf{Z}_{T-1})$, and $\mathbf{U} = (\mathbf{u}_1, \cdots, \mathbf{u}_T)$. The log-likelihood function of the conditional maximum likelihood estimation, assuming the VAR(p) model is Gaussian, is

$$l(\mathbf{B}, \boldsymbol{\Sigma}_u) = -\frac{KT}{2}\log 2\pi - \frac{T}{2}\log\det\boldsymbol{\Sigma}_u - \frac{1}{2}\operatorname{trace}\left[(\mathbf{Y} - \mathbf{B}\mathbf{Z})^\top \boldsymbol{\Sigma}_u^{-1} (\mathbf{Y} - \mathbf{B}\mathbf{Z})\right]. \tag{2.2}$$

We can obtain from (2.2) that the conditional maximum likelihood estimators (MLE) of $\mathbf{B}$ and $\boldsymbol{\Sigma}_u$ are

$$\hat{\mathbf{B}} = \mathbf{Y}\mathbf{Z}^\top \left(\mathbf{Z}\mathbf{Z}^\top\right)^{-1} \text{ and } \quad \hat{\boldsymbol{\Sigma}}_u = \frac{1}{T}\left(\mathbf{Y} - \hat{\mathbf{B}}\mathbf{Z}\right)\left(\mathbf{Y} - \hat{\mathbf{B}}\mathbf{Z}\right)^\top, \tag{2.3}$$

respectively, see Chapter 3.4 of Lütkepohl (2005). The log-likelihood function in (2.2) can also be rewritten as

$$l(\boldsymbol{\beta}, \boldsymbol{\Sigma}_u) = -\frac{KT}{2}\log 2\pi - \frac{T}{2}\log\det\boldsymbol{\Sigma}_u \\ -\frac{1}{2}\left[\mathbf{y} - \left(\mathbf{Z}^\top \otimes \mathbf{I}_K\right)\boldsymbol{\beta}\right]^\top \left(\mathbf{I}_T \otimes \boldsymbol{\Sigma}_u^{-1}\right)\left[\mathbf{y} - \left(\mathbf{Z}^\top \otimes \mathbf{I}_K\right)\boldsymbol{\beta}\right],$$

where $\boldsymbol{\beta} = \mathbf{vec}(\mathbf{B})$ is a $K(Kp+1)$-dimensional vector by stacking the coefficients matrix $\mathbf{B}$, $\mathbf{y} = \mathbf{vec}(\mathbf{Y})$, $\mathbf{I}_K$ is the $K \times K$ identity matrix, and $\otimes$ denotes the Kronecker product. Suppose there are linear constraints on $\boldsymbol{\beta}$ which are in the form $\mathbf{C}\boldsymbol{\beta} = \mathbf{c}$, where $\mathbf{C}$ is an $N \times (K^2p + K)$ matrix of known constants of rank N, and $\mathbf{c}$ is an N-dimensional vector of known constants. Then, the constrained maximum likelihood estimators of $\boldsymbol{\beta}$ and $\boldsymbol{\Sigma}_u$ are

$$\hat{\boldsymbol{\beta}} = \tilde{\boldsymbol{\beta}} + \left((\mathbf{Z}\mathbf{Z}^\top)^{-1} \otimes \hat{\boldsymbol{\Sigma}}_u\right)\mathbf{C}^\top\left[\mathbf{C}\left((\mathbf{Z}\mathbf{Z}^\top)^{-1} \otimes \hat{\boldsymbol{\Sigma}}_u\right)\mathbf{C}^\top\right]^{-1}\left(\mathbf{c} - \mathbf{C}\tilde{\boldsymbol{\beta}}\right) \text{ and}$$
$$\hat{\boldsymbol{\Sigma}}_u = \frac{1}{T}\left(\mathbf{Y} - \hat{\mathbf{B}}\mathbf{Z}\right)\left(\mathbf{Y} - \hat{\mathbf{B}}\mathbf{Z}\right)^\top,$$

respectively, where $\tilde{\boldsymbol{\beta}} = \left[(\mathbf{Z}\mathbf{Z}^\top)^{-1}\mathbf{Z} \otimes \mathbf{I}_K\right]\mathbf{y}$.

A K-dimensional VAR(p) model consists of $K(Kp+1)$ parameters, or K^2p parameters if the intercepts are excluded when the model is fully parametrized. Researchers have proposed various methods to overcome the over-parametrization issue when the model dimension is high relative to the sample size. One of these approaches is to identify the zero autoregressive (AR) coefficients by the conditional correlations between component series. The concept of conditional correlations between series components, together with two graphical time series models, is introduced in the next section.

2.2 Graphical Models

2.2.1 Conditional Correlation Graphs

Suppose $\{\mathbf{X}(t),\, t \in \mathbb{Z}\} = \left\{(X_1(t), \cdots, X_K(t))^\top,\, t \in \mathbb{Z}\right\}$ is a zero mean weakly stationary process, $\{X_a(t)\}$ and $\{X_b(t)\}$ are two distinct components, and $\mathcal{I}_{ab} = \{1, \cdots, K\} \backslash \{a, b\}$. The conditional correlations can be computed by determining the optimal linear filters for removing the linear effect of $\{\mathbf{X}_{\mathcal{I}_{ab}}(t)\}$ from each of $\{X_a(t)\}$ and $\{X_b(t)\}$. The optimal linear filters for removing the linear effect of $\{\mathbf{X}_{\mathcal{I}_{ab}}(t)\}$ from $\{X_a(t)\}$ minimize

$$\mathbb{E}\left[X_a(t) - \sum_{j \in \mathcal{I}_{ab}} \sum_{u \in \mathbb{Z}} g_{a,j}(u) X_j(t-u)\right]^2.$$

Denote the optimal linear filters by $\hat{g}_{a,j}(u)$ for $j \in \mathcal{I}_{ab}$ and $u \in \mathbb{Z}$, the remainders after removing the linear effect of $\{\mathbf{X}_{\mathcal{I}_{ab}}(t)\}$ from $\{X_a(t)\}$ and $\{X_b(t)\}$ are, respectively,

$$\begin{aligned} \varepsilon_{a|\mathcal{I}_{ab}}(t) &= X_a(t) - \sum_{j \in \mathcal{I}_{ab}} \sum_{u \in \mathbb{Z}} \hat{g}_{a,j}(u) X_j(t-u) \quad \text{and} \\ \varepsilon_{b|\mathcal{I}_{ab}}(t) &= X_b(t) - \sum_{j \in \mathcal{I}_{ab}} \sum_{u \in \mathbb{Z}} \hat{g}_{b,j}(u) X_j(t-u). \end{aligned} \tag{2.4}$$

Then the two processes, $\{X_a(t)\}$ and $\{X_b(t)\}$, are conditionally uncorrelated if and only if $\operatorname{cov}\left(\varepsilon_{a|\mathcal{I}_{ab}}(t),\, \varepsilon_{b|\mathcal{I}_{ab}}(t+u)\right) = 0$ for all $t, u \in \mathbb{Z}$.

Consider a graph $G = (V, E)$, where V denotes the set of vertices and $E \subset V \times V$ is the set of edges. We also assume that for $i, j \in V$, $(i, j) \in E$ if $(j, i) \in E$, i.e., the graph is undirected. Then, the conditional correlation graph of a weakly stationary multivariate process $\{\mathbf{X}(t),\, t \in \mathbb{Z}\}$ is defined

by

$$(i,j) \notin E \Leftrightarrow \{X_i(t)\} \perp\!\!\!\perp \{X_j(t)\} | \{\mathbf{X}_{\mathcal{I}_{ij}}(t)\}$$
$$\Leftrightarrow \operatorname{cov}\left(\varepsilon_{i|\mathcal{I}_{ij}}(t),\, \varepsilon_{j|\mathcal{I}_{ij}}(t+u)\right) = 0 \quad \text{for all } t, u \in \mathbb{Z}.$$

The edges of the graph can be determined by two approaches, namely the time domain approach and the frequency domain approach.

Time domain approach

The conditional correlation graph represents the linear association between two component series after removing the linear effect of all other components by two-sided filters. Similarly, we consider a bivariate VAR model to estimate the cross-correlation of the residuals, $\varepsilon_{a|V\setminus\{a,b\}}(t)$ and $\varepsilon_{b|V\setminus\{a,b\}}(t)$, following Hu et al. (2016). To illustrate the method, suppose $\mathbf{X}_V(t) = (X_1(t), X_2(t), X_3(t), X_4(t))^\top$, the VAR model to determine the partial cross-correlations of X_1 and X_2 given X_3 and X_4 is

$$\begin{pmatrix} X_1(t) \\ X_2(t) \end{pmatrix} = \begin{pmatrix} \mu_1(t) \\ \mu_2(t) \end{pmatrix} + \sum_{u=0}^{q} \mathbf{F}_u \begin{pmatrix} X_3(t-u) \\ X_4(t-u) \end{pmatrix} + \sum_{u=1}^{q} \mathbf{\Phi}_u \begin{pmatrix} X_1(t-u) \\ X_2(t-u) \end{pmatrix} + \begin{pmatrix} e_{1|\{3,4\}}(t) \\ e_{2|\{3,4\}}(t) \end{pmatrix}.$$

Then, the partial cross-correlations of X_1 and X_2 given the remaining processes, denoted by $\rho_{12|\mathcal{I}_{12}}(u)$, are the cross-correlations of $e_{1|\{3,4\}}(t)$ and $e_{2|\{3,4\}}(t)$, and other partial cross-correlations are computed similarly. The lag order q is determined by choosing a model that possesses the minimum BIC value among the bivariate models with various lag order over a pre-specified range of q. The approximate 5% error bound of $\pm 2/\sqrt{T}$ is adopted for testing the partial cross-correlations. We note that the time domain approach filters out the linear effect of the remaining components by one-sided

filters, which only consider the past and present observations in the filtering. An alternative to determining conditional correlation graphs of time series is the frequency domain method, which adopts two-sided filtering in the identification.

Frequency domain approach

To determine whether the two residual processes in (2.4) are uncorrelated at all lags, we can utilize the cross-spectral density of the two residuals in the frequency domain. The remainders are, in particular, conditionally uncorrelated at all lags if and only if the cross-spectral density of the two remainders, denoted by $f_{ab|\mathcal{I}_{ab}}(\lambda)$, is zero at all frequencies λ. This cross-spectral density, also known as the partial spectral density of $\{X_a(t)\}$ and $\{X_b(t)\}$ given $\{\mathbf{X}_{\mathcal{I}_{ab}}(t)\}$, is defined by

$$f_{ab|\mathcal{I}_{ab}}(\lambda) = \frac{1}{2\pi} \sum_{u \in \mathbb{Z}} \gamma_{ab|\mathcal{I}_{ab}}(u) e^{-i\lambda u}, \quad \lambda \in [-\pi, \pi],$$

where $\gamma_{ab|\mathcal{I}_{ab}}(u)$ is the cross-covariance function of the residual processes. Knowing that the cross-spectral density measures the degree of linear association between two variables in the frequency domain, the partial spectral density quantifies the degree of linear association between two components after removing the linear influence of the remaining variables. Koopmans (1995) and Brillinger (1981) mentioned that the computation of partial spectral density $f_{ab|\mathcal{I}_{ab}}(\lambda)$ could be formulated by

$$f_{ab|\mathcal{I}_{ab}}(\lambda) = f_{ab}(\lambda) - \mathbf{f}_{a\mathcal{I}_{ab}}(\lambda) \mathbf{f}_{\mathcal{I}_{ab}\mathcal{I}_{ab}}(\lambda)^{-1} \mathbf{f}_{\mathcal{I}_{ab}b}(\lambda)^{*},$$

where $\mathbf{A}^{*}$ is the conjugate transpose of the matrix $\mathbf{A}$; $\mathbf{f}_{a\mathcal{I}_{ab}}(\lambda)$, $\mathbf{f}_{\mathcal{I}_{ab}\mathcal{I}_{ab}}(\lambda)$ and $\mathbf{f}_{\mathcal{I}_{ab}b}(\lambda)$ are some partitioned matrices of the spectral density matrix. Hu

et al. (2016) gave an example of this computation.

The cross-spectral density and partial spectral density are typically normalized to spectral coherence $R_{ab}(\lambda)$ and partial spectral coherence $R_{ab|\mathcal{I}_{ab}}(\lambda)$ for the analysis of conditional correlation structure. They are defined by

$$R_{ab}(\lambda) = \frac{f_{ab}(\lambda)}{\sqrt{f_{aa}(\lambda) f_{bb}(\lambda)}} \quad \text{and} \quad R_{ab|\mathcal{I}_{ab}}(\lambda) = \frac{f_{ab|\mathcal{I}_{ab}}(\lambda)}{\sqrt{f_{aa|\mathcal{I}_{ab}}(\lambda) f_{bb|\mathcal{I}_{ab}}(\lambda)}},$$

respectively. Dahlhaus (2000) introduced an efficient method to compute the partial spectral coherencies from the inverse of the spectral density matrix. The partial spectral coherence can be evaluated by

$$R_{ab|\mathcal{I}_{ab}}(\lambda) = -\frac{g_{ab}^{-1}(\lambda)}{\sqrt{g_{aa}^{-1}(\lambda) g_{bb}^{-1}(\lambda)}},$$

where $g_{ab}^{-1}(\lambda)$ is the (a, b)-th element of the inverse spectral density matrix at frequency λ. In practice, we need to estimate the spectral density and infer whether the partial spectral coherence is zero from samples.

Suppose $\mathbf{X}(t) = (X_1(t), \cdots, X_K(t))^\top$, $t \in \mathbb{Z}$ is a multivariate weakly stationary time series with spectral density matrix $\mathbf{f}(\lambda) = [f_{ab}(\lambda)]$. Assume that there are T observations, we can estimate $\mathbf{f}(\lambda)$ by smoothing the periodogram $\mathbf{I}(\lambda) = [I_{ab}(\lambda)]$. The cross-periodogram $I_{ab}(\lambda)$ is defined as

$$I_{ab}(\lambda) = W_a(\lambda)\overline{W_b(\lambda)},$$

where $W_a(\lambda) = (h_2)^{-1/2} \sum_{t=1}^{T} h\left(\frac{t}{T}\right) X_a(t) e^{-i\lambda t}$, is the discrete Fourier transform of $\{X_a(t)\}$, and $h_2 = \sum_{t=1}^{T} h^2\left(\frac{t}{T}\right)$. Here, the function $h(\cdot)$ is the taper for correcting bias in the estimation. We apply the cosine taper in the

estimation, which is, for $0 \leq a \leq 1/2$,

$$h(x) = \begin{cases} [1 - \cos(\pi x/a)]/2, & 0 < x \leq a, \\ 1, & a < x \leq 1 - a, \\ [1 - \cos(\pi(1-x)/a)]/2, & 1 - a < x \leq 1, \\ 0, & \text{elsewhere.} \end{cases}$$

Other tapers can be found in Koopmans (1995, Chpater 9.2) and Brillinger (1981, Chapter 3.3). Denote $[x]$ be the largest integer less than or equal to x. Let $\lambda_j = \frac{2\pi j}{T}$, $j = -\left[\frac{T-1}{2}\right], \cdots, -1, 0, 1, \cdots, \left[\frac{T}{2}\right]$, be the j-th Fourier frequency, the spectral density is consistently estimated by

$$\hat{f}_{ab}(\lambda) = \left(2\pi \sum_{|k| \leq m_T} w_k \right)^{-1} \sum_{|k| \leq m_T} w_k I_{ab}(\lambda_{j+k}),$$

where w_k is a weight sequence, and m_T is the bandwidth depending on T. We implement the weight function $w_k = \cos(\frac{k\pi}{m_T})$, for $k = -m_T, \cdots, m_T$, in the estimation. We refer readers to Brillinger (1981, Chapter 3.3) and Wei (2006, Chapter 13.3) for alternative weight functions and their properties. The bandwidth m_T is selected by minimizing the cross-validated log likelihood,

$$\text{CVLL}(m_T) = \frac{1}{T} \sum_{j=1}^{[T/2]} \text{trace} \left[\mathbf{I}(\lambda_j) \hat{\mathbf{f}}_{-j}^{-1}(m_T, \lambda_j) \right] + \log\det \left[\hat{\mathbf{f}}_{-j}(m_T, \lambda_j) \right],$$

where $\hat{f}_{ab,-j}(m, \lambda_j) = \left(2\pi \sum_{\substack{|k| \leq m, \\ k \neq 0}} w_k \right)^{-1} \sum_{\substack{|k| \leq m, \\ k \neq 0}} w_k I_{ab}(\lambda_{j+k})$. The bandwidth selection by CVLL was studied by Beltrão & Bloomfield (1987) and Matsuda & Yajima (2004).

With the estimated spectral density matrix, the spectral coherence $\hat{R}_{ab}(\lambda)$

and partial spectral coherence $\hat{R}_{ab|\mathcal{I}_{ab}}(\lambda)$ can be estimated. Under the hypothesis of $R_{ab|\mathcal{I}_{ab}}(\lambda) = 0$, the test statistic $\frac{(n-q)\hat{R}^2_{ab|\mathcal{I}_{ab}}(\lambda)}{1-\hat{R}^2_{ab|\mathcal{I}_{ab}}(\lambda)}$ follows the F distribution with 2 and $2(n-q)$ degrees of freedom at each frequency λ. Here, n and q are the equivalent degrees of freedom and the number of components other than components a and b, see Koopmans (1995, Chpater 8.3) and Wei (2006, Chapter 13.3.4) for details. Similarly, the test statistic for testing zero coherence is given by $\frac{(n-1)\hat{R}^2_{ab}(\lambda)}{1-\hat{R}^2_{ab}(\lambda)}$, which follows the F distribution with 2 and $2(n-1)$ degrees of freedom. Alternatively, Dahlhaus (2000) proposed to use the supremum of the squared empirical partial spectral coherence, $\sup_{\lambda} |\hat{R}_{ab|\mathcal{I}_{ab}}(\lambda)|^2$, as a test statistic for zero partial coherence, which is asymptotically χ^2_2 distributed. Thus, the zero conditional correlation can be determined by the test statistic of partial spectral coherence because of the equivalence between zero conditional correlations at all lags and zero partial spectral coherencies at all frequencies.

We illustrate the conditional correlation graph by considering the following example. Consider the following 3-dimensional VAR(1) process,

$$\begin{pmatrix} x_{1,t} \\ x_{2,t} \\ x_{3,t} \end{pmatrix} = \begin{pmatrix} a_{11} & 0 & 0 \\ a_{21} & a_{22} & 0 \\ a_{31} & 0 & a_{33} \end{pmatrix} \begin{pmatrix} x_{1,t-1} \\ x_{2,t-1} \\ x_{3,t-1} \end{pmatrix} + \begin{pmatrix} \varepsilon_{1,t} \\ \varepsilon_{2,t} \\ \varepsilon_{3,t} \end{pmatrix},$$

where $\boldsymbol{\varepsilon}_t = (\varepsilon_{1,t}, \varepsilon_{2,t}, \varepsilon_{3,t})^\top \sim N(\mathbf{0}, \boldsymbol{\Sigma}_u)$ with $\boldsymbol{\Sigma}_u^{-1} = \begin{pmatrix} \theta_{11} & \theta_{12} & \theta_{13} \\ \theta_{12} & \theta_{22} & 0 \\ \theta_{13} & 0 & \theta_{33} \end{pmatrix}$. Figure 2.1 illustrates the conditional correlation graph of this series, where components X_2 and X_3 are conditionally uncorrelated.

Davis et al. (2016) proposed to impose zero constraints on the AR coefficients when the two components are conditionally uncorrelated. That is, $(\mathbf{A}_l)_{ij} = (\mathbf{A}_l)_{ji} = 0$ for $i \neq j$ and $l = 1, \cdots, p$ when $\{X_i(t)\}$ and $\{X_j(t)\}$ are

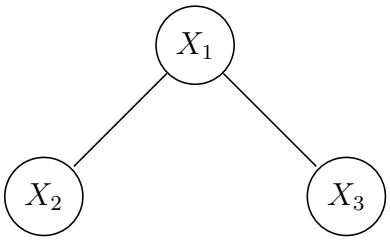


FIGURE 2.1: A partial correlation graph represents a 3-dimensional VAR(1) process, where components X_2 and X_3 are conditionally uncorrelated.

conditionally uncorrelated. We can determine from the conditional correlation graph that two components are conditionally uncorrelated at all lags, including lag zero. The corresponding elements of the inverse covariance matrix $\mathbf{\Sigma}_u^{-1}$ seem to be insignificant. We, therefore, suggest implementing sparsity constraints on $\mathbf{\Sigma}_u^{-1}$ in addition to the AR coefficients, based on the inferred conditional correlation graph, to achieve parsimonious model. These restrictions on both the AR coefficients and the inverse covariances become essential when the Granger causality graph is constructed.

2.2.2 Granger Causality Graphs

Eichler (2012) applied mixed graphs to visualize the causal relationships between the components of multivariate stationary time series. Similar to the conditional correlation graphs, each vertex of the graph represents a component series. The directed edges between vertices indicate the presence of Granger-causality (Granger, 1969), and the undirected edges encode the contemporaneous conditional correlation structure. The concept of Granger-causality from a process $\{X_i(t)\}$ to another process $\{X_j(t)\}$ is based on investigating whether the prediction of $X_j(t+1)$, at time t, can be improved by using all relevant information up to time t apart from $\{X_i(t)\}$. Consider $\{\mathbf{X}(t),\, t \in \mathbb{Z}\} = \{X_k(t),\, t \in \mathbb{Z} \,\text{and}\, k = 1, \cdots, K\}$ is a K-dimensional stationary process. By denoting $V = \{1, \cdots, K\}$, $\mathscr{X}_A(t) = \{\mathbf{X}_A(s),\, s \leq t\}$ be the set of all past and present values of $\{\mathbf{X}_A(t)\}$

at time t for $A \subseteq V$. Then the formal definition of Granger-noncausality is given by, for i, $j \in V$ and $i \neq j$, $\{X_i(t)\}$ is Granger-noncausal for $\{X_j(t)\}$ relative to $\{\mathbf{X}_V(t)\}$ if

$$\mathscr{X}_i(t+1) \perp\!\!\!\perp \mathscr{X}_j(t) | \mathscr{X}_{V\setminus\{j\}}(t) \quad \text{for all } t \in \mathbb{Z}.$$

Similarly, $\{X_i(t)\}$ and $\{X_j(t)\}$ are contemporaneous conditionally uncorrelated with respect to $\{\mathbf{X}_V(t)\}$ if

$$\mathscr{X}_i(t+1) \perp\!\!\!\perp \mathscr{X}_j(t+1) | \mathscr{X}_V(t),\ \mathscr{X}_{V\setminus\{i,j\}}(t+1) \quad \text{for all } t \in \mathbb{Z}.$$

Therefore, the Granger causality graph can be defined according to above definitions. We consider a mixed graph $G = (V, E_u, E_d)$, where V is the set of vertices, $E_u \subset V \times V$ is the set of undirected edges, and $E_d \subset V \times V$ is the set of directed edges. Then, the Granger causality graph is defined by

(i) $(i,j) \notin E_d \Leftrightarrow \mathscr{X}_i(t+1) \perp\!\!\!\perp \mathscr{X}_j(t) | \mathscr{X}_{V\setminus\{j\}}(t)$ for all $t \in \mathbb{Z}$;

(ii) $(i,j) \notin E_u \Leftrightarrow \mathscr{X}_i(t+1) \perp\!\!\!\perp \mathscr{X}_j(t+1) | \mathscr{X}_V(t),\ \mathscr{X}_{V\setminus\{i,j\}}(t+1)$ for all $t \in \mathbb{Z}$.

We consider a vector autoregressive process to illustrate the Granger causality graph. Let $\{\mathbf{X}(t),\ t \in \mathbb{Z}\}$ be a K-dimensional VAR(p) process,

$$\mathbf{X}(t) = \sum_{l=1}^{p} \mathbf{A}_l \mathbf{X}(t-l) + \boldsymbol{\varepsilon}(t),$$

where $\mathbf{A}_1, \cdots, \mathbf{A}_p$ are the AR coefficients matrices and $\boldsymbol{\varepsilon}(t)$ are identically and independently distributed with mean $\mathbf{0}$ and covariance $\boldsymbol{\Sigma}$. We also denote $\boldsymbol{\Theta} = \boldsymbol{\Sigma}^{-1} = (\theta_{ij})$. Then, the Granger causality graph of this process is characterized by

(i)

$$(i,j) \notin E_d \Leftrightarrow \mathscr{X}_i(t+1) \perp\!\!\!\perp \mathscr{X}_j(t) | \mathscr{X}_{V\setminus\{j\}}(t) \quad \text{for all } t \in \mathbb{Z}$$
$$\Leftrightarrow (\mathbf{A}_l)_{ji} = 0 \quad \text{for all } l \in \{1, \cdots, p\},$$

(ii)

$$(i,j) \notin E_u \Leftrightarrow \mathscr{X}_i(t+1) \perp\!\!\!\perp \mathscr{X}_j(t+1) | \mathscr{X}_V(t),\ \mathscr{X}_{V\setminus\{i,j\}}(t+1) \text{ for all } t \in \mathbb{Z}$$
$$\Leftrightarrow \varepsilon_i(t) \perp\!\!\!\perp \varepsilon_j(t) | \varepsilon_{V\setminus\{i,j\}}(t) \quad \text{for all } t \in \mathbb{Z}$$
$$\Leftrightarrow \theta_{ij} = \theta_{ji} = 0.$$

We consider the following VAR(1) process as an example,

$$\mathbf{X}(t) = \mathbf{A}_1 \mathbf{X}(t-1) + \boldsymbol{\varepsilon}(t),$$

where $\mathbf{A}_1 = \begin{pmatrix} a_{1,11} & 0 & a_{1,13} & 0 \\ a_{1,21} & a_{1,22} & 0 & 0 \\ a_{1,31} & 0 & a_{1,33} & 0 \\ 0 & a_{1,42} & 0 & a_{1,44} \end{pmatrix}$ and $\boldsymbol{\Theta} = \boldsymbol{\Sigma}^{-1} = \begin{pmatrix} \theta_{11} & \theta_{12} & \theta_{13} & 0 \\ \theta_{21} & \theta_{22} & 0 & \theta_{24} \\ \theta_{31} & 0 & \theta_{33} & 0 \\ 0 & \theta_{42} & 0 & \theta_{44} \end{pmatrix}$.

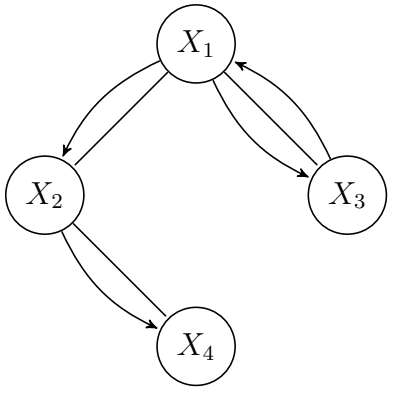


FIGURE 2.2: An example of a Granger causality graph.

Figure 2.2 shows the Granger causality graph of this VAR(1) process. We can observe from this graph that $\{X_1(t)\}$ causes $\{X_4(t)\}$ through $\{X_2(t)\}$,

but not $\{X_1(t)\}$ causes $\{X_4(t)\}$ with respect to the whole process $\{\mathbf{X}(t)\}$.

The Granger causality graphs reflect the dynamic interdependencies among the components of multiple time series processes. In contrast, Oxley et al. (2004) suggested the use of a directed acyclic graph (DAG) to describe a structural vector autoregressive (SVAR) model. Such graph represents each component at each time point by a vertex and encodes both the intra and interdependencies among the variables of the process.

Both graphs show the contemporaneous conditional dependencies. The Granger causality graph characterizes the undirected edges by the inverse innovation covariance matrix which exhibits the contemporaneous conditional dependence structure. The DAG, however, characterizes the directed edges between current variables based on the corresponding autoregressive coefficients of the current variables in the SVAR model. Indeed, both methods capture the contemporaneous conditional dependencies similarly. To explain this point, we consider an SVAR model and its reduced form. For more detailed exposition, see Tunnicliffe-Wilson et al. (2015). Consider an SVAR model of the form,

$$\mathbf{\Psi}_0\mathbf{x}_t = \mathbf{\Psi}_1\mathbf{x}_{t-1} + \mathbf{\Psi}_2\mathbf{x}_{t-2} + \cdots + \mathbf{\Psi}_p\mathbf{x}_{t-p} + \mathbf{e}_t,$$

where $\mathbf{\Psi}_0$ is non-singular, and the covariance matrix $\mathbf{D}$ of $\mathbf{e}_t$ is assumed to be diagonal. This model can be transformed to a VAR model, i.e.,

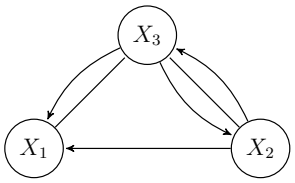
$$\begin{aligned}\mathbf{x}_t &= \mathbf{\Psi}_0^{-1}\mathbf{\Psi}_1\mathbf{x}_{t-1} + \mathbf{\Psi}_0^{-1}\mathbf{\Psi}_2\mathbf{x}_{t-2} + \cdots + \mathbf{\Psi}_0^{-1}\mathbf{\Psi}_p\mathbf{x}_{t-p} + \mathbf{\Psi}_0^{-1}\mathbf{e}_t \\ &= \mathbf{\Phi}_1\mathbf{x}_{t-1} + \mathbf{\Phi}_2\mathbf{x}_{t-2} + \cdots + \mathbf{\Phi}_p\mathbf{x}_{t-p} + \mathbf{u}_t,\end{aligned}$$

where $\mathbf{\Phi}_i = \mathbf{\Psi}_0^{-1}\mathbf{\Psi}_i$ for $i = 1, \cdots, p$, $\mathbf{u}_t = \mathbf{\Psi}_0^{-1}\mathbf{e}_t$, and the covariance matrix of $\mathbf{u}_t$ is $\mathbf{\Sigma}_u$. Therefore, the relation between the residuals $\mathbf{e}_t$ of the SVAR

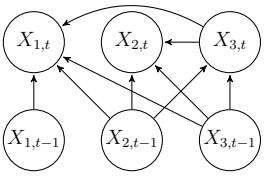
and the innovations $\mathbf{u}_t$ of the transformed model is

$$\boldsymbol{\Sigma}_u^{-1} = \boldsymbol{\Psi}_0^\top \mathbf{D}^{-1} \boldsymbol{\Psi}_0.$$

Thus, the inverse of innovation covariance matrix of the VAR model reflects the conditional dependence between the current variables given the past variables. The inclusion of $\boldsymbol{\Psi}_0$ in SVAR, however, provides an alternative way to capture the contemporaneous conditional dependences; see Figure 2.3.



(a) An example of a mixed graph describing a VAR model.



(b) An example of a DAG describing a SVAR model.

FIGURE 2.3: The graphical representation of VAR and SVAR models.

For the SVAR model, the covariance matrix $\mathbf{D}$ of $\mathbf{e}_t$ is assumed to be diagonal so that the model is identifiable. The dependence between the current variables is also assumed being recursive and not cyclical so that the matrix $\boldsymbol{\Psi}_0$ is triangular with unit diagonal after reordering the variables. Note that these restrictions are not required when building a VAR model to study the dynamic interdependencies between variables of a multivariate time series process.

2.3 Summary

In this chapter, we have reviewed the constrained maximum likelihood estimation on vector autoregressive (VAR) models which is crucial when sparsity constraints on the autoregressive coefficients are pre-specified. The sparsity constraints can be identified by computing the partial cross-correlations

or the partial spectral coherencies to determine the possible conditional correlation structure. Such conditional correlation structure of a multiple time series can be visualized by an undirected graph and called the conditional correlation graph.

Apart from the conditional correlation graph, we have presented the Granger causality graph which encodes the possible Granger-causality structure and the contemporaneous conditional correlation structure in a mixed graph. We also illustrated the connection between the causality graph and the VAR process by an example and compared the causality graph with the directed acyclic graph (DAG) representing a structural vector autoregressive (SVAR) model. The next chapter will introduce an alternating maximization method to estimate sparse VAR models, in which both the autoregressive coefficients and the inverse covariance matrix are constrained. The causality graph is utilized to visualize the estimated sparse VAR models.

Chapter 3

Constrained Likelihood Estimation Method

Graphical time series models express the dynamic interrelationships between variables of multivariate time series in graphs. Researchers have attempted to infer sparse graphical models to reduce the model complexity for better interpretation. Oxley et al. (2004) endeavoured the use of a directed acyclic graph to represent a sparse structural vector autoregressive (SVAR) model. Such sparse SVAR model is constructed based on a conditional independence graph, determined by the partial correlations of variables, to ensure the spareness of the SVAR model. Songsiri et al. (2009) studied a convex relaxation method for the estimation on vector autoregressive (VAR) models subject to conditional independencies constraints.

In the present chapter, we will introduce a constrained likelihood estimation method for sparse VAR models. Sparsity constraints are imposed on both the autoregressive (AR) coefficients and the inverse covariance matrix. This method is crucial if the graphical VAR model is of interest. We will formulate the model estimation problem as a "biconcave" problem

(Gorski et al., 2007). The optimization problem is concave when either the AR coefficients or the inverse noise covariance matrix is fixed. An alternating maximization method will be presented to solve the "biconcave" problem. We will study the estimation performance of the alternating maximization method by simulation experiments, assuming the sparsity structure is known, and compare with the interior point method and the direct search method. We also compare the time domain and frequency domain methods, discussed in Section 2.2.1, by simulation studies. The estimation method is applied to real datasets for illustration.

3.1 Problem Description

3.1.1 Problem Formulation

Most studies of the estimation on sparse VAR models have been confined to impose sparsity constraints on the AR coefficients, rather than on the inverse noise covariance matrix. We propose an alternating maximization method to impose sparseness on both the AR coefficients and the inverse of innovation covariance matrix. Recall from Section 2.1 that the log-likelihood function of the conditional maximum likelihood estimation, assuming the VAR(p) model is Gaussian, is

$$l(\mathbf{B}, \mathbf{\Sigma}_u) = -\frac{KT}{2}\log 2\pi - \frac{T}{2}\log\det\mathbf{\Sigma}_u - \frac{1}{2}\,\mathrm{trace}\left[(\mathbf{Y}-\mathbf{BZ})^\top \mathbf{\Sigma}_u^{-1}(\mathbf{Y}-\mathbf{BZ})\right].$$

Using the notation in Chapter 2, we consider the following problem,

$$\begin{aligned}
\underset{\mathbf{B},\boldsymbol{\Sigma}_u}{\text{maximize}} \quad & -\frac{KT}{2}\log 2\pi - \frac{T}{2}\log\det\boldsymbol{\Sigma}_u - \frac{1}{2}\operatorname{trace}\left[(\mathbf{Y}-\mathbf{BZ})^\top \boldsymbol{\Sigma}_u^{-1}(\mathbf{Y}-\mathbf{BZ})\right] \\
\text{subject to} \quad & \begin{cases} (\mathbf{A}_l)_{ij} = (\mathbf{A}_l)_{ji} = 0, \quad \text{for } l = 1,\cdots,p \text{ and } (i,j)\in\mathcal{S}, \\ \left(\boldsymbol{\Sigma}_u^{-1}\right)_{ij} = 0, \quad (i,j)\in\mathcal{S}, \\ \boldsymbol{\Sigma}_u^{-1} \succ 0, \end{cases}
\end{aligned} \tag{3.1}$$

where p is a pre-determined lag order and $\mathcal{S}$ contains the indices of the pairs of components that are conditionally uncorrelated, assuming that $(i,j) \in \mathcal{S}$ for $i < j$. This set is determined based on the identified partial correlation graph mentioned in Section 2.2.1 and will be discussed in Section 3.1.2. We rewrite the problem (3.1) as

$$\begin{aligned}
\underset{\mathbf{B},\boldsymbol{\Theta}}{\text{maximize}} \quad & -\frac{KT}{2}\log 2\pi + \frac{T}{2}\log\det\boldsymbol{\Theta} - \frac{1}{2}\operatorname{trace}\left[(\mathbf{Y}-\mathbf{BZ})^\top \boldsymbol{\Theta}(\mathbf{Y}-\mathbf{BZ})\right] \\
\text{subject to} \quad & \begin{cases} \mathbf{C}\boldsymbol{\beta} = \mathbf{0}, \\ \theta_{ij} = 0, \quad (i,j)\in\mathcal{S}, \\ \boldsymbol{\Theta} \succ 0, \end{cases}
\end{aligned} \tag{3.2}$$

where $\boldsymbol{\beta} = \mathbf{vec}(\mathbf{B})$, $\mathbf{C}$ is a matrix of known constants with full row rank, and $\mathbf{0}$ is a vector of zeros. Here, we utilize the relation $-\log\det(\boldsymbol{\Sigma}_u) = \log\det(\boldsymbol{\Sigma}_u^{-1})$, incorporate the zero constraints of the AR coefficients through $\mathbf{C}\boldsymbol{\beta} = \mathbf{0}$, and denote $\boldsymbol{\Theta}$ as $\boldsymbol{\Sigma}_u^{-1}$.

Theorem 3.1. *The optimization problem in* (3.2) *with respect to* $\mathbf{B}$ *and* $\boldsymbol{\Theta}$ *is biconcave.*

Proof. See Appendix A. □

Theorem 3.1 shows that the optimization problem is "biconcave" indicating that the problem is concave for either fixed $\mathbf{B}$ or $\mathbf{\Theta}$ (Gorski et al., 2007). Thus, we adopt an iterative algorithm, called the Alternate Convex Search (ACS) algorithm (Gorski et al., 2007; Hastie et al., 2015), by first estimating $\mathbf{B}$ followed by estimating $\mathbf{\Theta}$, until a stopping criterion is satisfied. The objective function of the problem for fixed positive definite $\mathbf{\Theta}$ is strictly concave and is strictly concave on the set of positive definite matrices for fixed $\mathbf{B}$. Therefore, a unique maximizer in each subproblem is obtained. Gorski et al. (2007) stated that each accumulation point generated by the ACS algorithm is a stationary point of the objective function under an assumption. The assumption is that the set of all accumulation points generated by the ACS algorithm form a connected, compact set. Note that the solution obtained using the ACS algorithm is not guaranteed to be the global optimum of the problem.

The solution of $\mathbf{\Theta}$ for fixed $\mathbf{B}$ at each iteration guarantees the positive definiteness of the inverse of innovation covariance matrix. This positive definiteness is not ensured when the problem is solved by traditional iterative numerical procedures like Newton-Raphson method. The suggested iterative method does not require the computation of the Hessian or the information matrix explicitly comparing to the Newton-Raphson method.

We note the zero constraints are chosen based on the identified conditional correlation graph mentioned in Section 2.2.1. Before introducing the proposed alternating maximization method, we discuss the possible methods in identifying the constraint structure in the next section.

3.1.2 Estimation of the Structure

To identify the constraint structure, we first determine the partial correlation graph of a series by the frequency or time domain methods, introduced in Section 2.2.1. Suppose a conditional correlation graph of Figure 2.1 is identified, the constraint structure for model estimation is

$$\theta_{23} = \theta_{32} = 0 \quad \text{and} \quad (\mathbf{A}_l)_{23} = (\mathbf{A}_l)_{32} = 0, \quad \text{for } l = 1, \cdots, p.$$

The lag order p is determined by standard information criteria, such as BIC or HQC (Hannan & Quinn, 1979), before applying the alternating maximization method. In the calculation of the information criteria, we count all the unconstrained autoregressive coefficients and the unconstrained inverse noise covariances at the upper triangular part of the matrix as the number of parameters m, i.e., $m = K^2p + \frac{K(K+1)}{2} - (2p+1)|\mathcal{S}|$.

In practice, some of the partial cross-correlations (partial spectral coherencies) are marginally significant at few lags (frequencies) leading to some weak links in the estimated partial correlation graph. We can therefore further reduce the number of parameters. For the marginal partial cross-correlations, we rank them by their absolute values, $\max_u |\hat{\rho}_{ab|\mathcal{I}_{ab}}(u)|$, (or the supremum of the test statistics of partial spectral coherencies, $\sup_\lambda \frac{(n-q)\hat{R}^2_{ab|\mathcal{I}_{ab}}(\lambda)}{1-\hat{R}^2_{ab|\mathcal{I}_{ab}}(\lambda)}$), in descending order. Then we can exclude some of the initially marginal partial cross-correlations in a forward stepwise regression manner. We finally select the model that possesses the minimum BIC value among the fitted models. We next present the iterative estimation algorithm.

3.1.3 Proposed Iterative Method

The proposed alternating maximization procedure:

(i) Initialization: Set the initial estimates using the unconstrained maximum likelihood estimators in (2.3),

$$\hat{\mathbf{B}}_{(0)} = \mathbf{Y}\mathbf{Z}^\top \left(\mathbf{Z}\mathbf{Z}^\top\right)^{-1} \text{ and } \quad \hat{\mathbf{\Theta}}_{(0)} = \left[\frac{1}{T}\left(\mathbf{Y} - \hat{\mathbf{B}}_{(0)}\mathbf{Z}\right)\left(\mathbf{Y} - \hat{\mathbf{B}}_{(0)}\mathbf{Z}\right)^\top\right]^{-1}.$$

Remark 3.1. Although the initial estimate is not in the feasible region, we can consider the initialization as a warm start, since the solution of next iteration is feasible.

(ii) $\mathbf{B}$ step: Given the estimate of $\mathbf{\Theta}$ at the $(k-1)$-th iteration, denoted by $\hat{\mathbf{\Theta}}_{(k-1)}$, the estimate of $\mathbf{B}$ at the k-th iteration is

$$\begin{aligned}\hat{\boldsymbol{\beta}}_{(k)} = \mathbf{vec}\left(\hat{\mathbf{B}}_{(k)}\right) = \tilde{\boldsymbol{\beta}} - \left((\mathbf{Z}\mathbf{Z}^\top)^{-1} \otimes \hat{\mathbf{\Theta}}^{-1}_{(k-1)}\right) \\ \mathbf{C}^\top\left[\mathbf{C}\left((\mathbf{Z}\mathbf{Z}^\top)^{-1} \otimes \hat{\mathbf{\Theta}}^{-1}_{(k-1)}\right)\mathbf{C}^\top\right]^{-1}\mathbf{C}\tilde{\boldsymbol{\beta}},\end{aligned} \tag{3.3}$$

where $\tilde{\boldsymbol{\beta}} = \mathbf{vec}\left(\hat{\mathbf{B}}_{(0)}\right) = \left[(\mathbf{Z}\mathbf{Z}^\top)^{-1}\mathbf{Z} \otimes \mathbf{I}_K\right]\mathbf{y}$, which is computed in the initialization stage. That means the components involving $\tilde{\boldsymbol{\beta}}$ in (3.3) need not be recalculated at each iteration.

(iii) $\mathbf{\Theta}$ step: Given $\hat{\mathbf{B}}_{(k)}$, we solve for $\mathbf{\Theta}$ using

$$\begin{aligned}\hat{\mathbf{\Theta}}_{(k)} = \underset{\mathbf{\Theta}\succ 0}{\arg\max} \quad & \log\det\mathbf{\Theta} - \operatorname{trace}\left(\mathbf{S}_{(k)}\mathbf{\Theta}\right) \\ \text{subject to} \quad & \theta_{ij} = 0, \quad (i,j) \in \mathcal{S},\end{aligned}$$

where $\mathbf{S}_{(k)} = \frac{1}{T}\left(\mathbf{Y} - \hat{\mathbf{B}}_{(k)}\mathbf{Z}\right)\left(\mathbf{Y} - \hat{\mathbf{B}}_{(k)}\mathbf{Z}\right)^\top$.

(iv) Repeat step (ii) and (iii) until a stopping criterion is met, say $\|\hat{\mathbf{B}}_{(k+1)} - \hat{\mathbf{B}}_{(k)}\|_F < \varepsilon$ and $\|\hat{\mathbf{\Theta}}_{(k+1)} - \hat{\mathbf{\Theta}}_{(k)}\|_F < \varepsilon$. Here, $\|\cdot\|_F$ denotes the Frobenius norm, and ε is a small positive number, for example, $\epsilon = 10^{-6}$.

The Lagrange dual function of the covariance selection problem in step (iii) is

$$
\begin{aligned}
g(\boldsymbol{\nu}) &= \inf_{\boldsymbol{\Theta} \succ 0} \left(\log\det \boldsymbol{\Theta} - \operatorname{trace}(\boldsymbol{\Theta}\mathbf{S}) - 2 \sum_{(i,j)\notin\mathcal{S}} \nu_{ij}\theta_{ij} \right) \\
&= -\log\det \left(\mathbf{S} + \sum_{(i,j)\notin\mathcal{S}} \nu_{ij} \left(\mathbf{e}_i\mathbf{e}_j^\top + \mathbf{e}_j\mathbf{e}_i^\top \right) \right) - K,
\end{aligned}
$$

where $\mathbf{e}_i$ is an i-th unit vector of dimension K, and ν_{ij} are the Lagrange multipliers for the equality constraints. The dual problem is

$$
\begin{aligned}
&\underset{\boldsymbol{\Gamma} \succ 0}{\text{minimize}} && -\log\det \boldsymbol{\Gamma} \\
&\text{subject to} && \gamma_{ij} = s_{ij}, \quad (i,j) \in \mathcal{S},
\end{aligned}
$$

where $\boldsymbol{\Gamma} = \mathbf{S} + \sum_{(i,j)\notin\mathcal{S}} \nu_{ij} \left(\mathbf{e}_i\mathbf{e}_j^\top + \mathbf{e}_j\mathbf{e}_i^\top \right)$, which is a determinant maximization problem (Vandenberghe et al., 1998). Therefore, it can be solved by semidefinite programming (SDP) solvers, such as SDPT3 (Tütüncü et al., 2003) or SeDuMi (Sturm, 1999).

3.2 Numerical Results

3.2.1 Simulation

In the simulation study, we consider five different stable VAR models, in which the autoregressive coefficient matrix $\mathbf{A}_l$ and the inverse noise covariance matrix $\boldsymbol{\Sigma}_u^{-1}$ have the same structure (i.e. $(\mathbf{A}_l)_{ij} = (\mathbf{A}_l)_{ji} = (\boldsymbol{\Sigma}_u^{-1})_{ij} = (\boldsymbol{\Sigma}_u^{-1})_{ji} = 0$, $1 \le i < j \le K$, $l = 1, \cdots, p$), to measure the performance of the estimation method. The inverses noise covariance matrices of each model are positive definite. We perform the experiments using the following models:

Model 1. $\mathbf{y}_t^{(1)} = \mathbf{A}_1^{(1)}\mathbf{y}_{t-1}^{(1)} + \mathbf{u}_t^{(1)}, \quad \mathbf{u}_t^{(1)} \sim N(\mathbf{0}, \mathbf{\Sigma}_1)$,

Model 2. $\mathbf{y}_t^{(2)} = \mathbf{A}_1^{(2)}\mathbf{y}_{t-1}^{(2)} + \mathbf{u}_t^{(2)}, \quad \mathbf{u}_t^{(2)} \sim N(\mathbf{0}, \mathbf{\Sigma}_2)$,

Model 3. $\mathbf{y}_t^{(3)} = \mathbf{A}_1^{(3)}\mathbf{y}_{t-1}^{(3)} + \mathbf{u}_t^{(3)}, \quad \mathbf{u}_t^{(3)} \sim N(\mathbf{0}, \mathbf{\Sigma}_3)$,

Model 4. $\mathbf{y}_t^{(4)} = \mathbf{A}_1^{(4)}\mathbf{y}_{t-1}^{(4)} + \mathbf{u}_t^{(4)}, \quad \mathbf{u}_t^{(4)} \sim N(\mathbf{0}, \mathbf{\Sigma}_4)$,

Model 5. $\mathbf{y}_t^{(5)} = \mathbf{A}_1^{(5)}\mathbf{y}_{t-1}^{(5)} + \mathbf{A}_2^{(5)}\mathbf{y}_{t-2}^{(5)} + \mathbf{u}_t^{(5)}, \quad \mathbf{u}_t^{(5)} \sim N(\mathbf{0}, \mathbf{\Sigma}_5)$,

where

$$\mathbf{A}_1^{(1)} = \begin{pmatrix} -0.7458 & 0.3938 & -0.9575 \\ -0.1824 & -0.6798 & 0 \\ -0.1779 & 0 & 0.4294 \end{pmatrix}, \ \mathbf{\Sigma}_1^{-1} = \begin{pmatrix} 1.3030 & -1.0613 & 0.8662 \\ -1.0613 & 1.4196 & 0 \\ 0.8662 & 0 & 2.6625 \end{pmatrix},$$

$$\mathbf{A}_1^{(2)} = \begin{pmatrix} 0.9508 & 0 & 0.4352 & 0 \\ 0 & -0.8232 & 0.5138 & 0.0274 \\ -0.8592 & -0.8289 & 0.6247 & 0.5984 \\ 0 & -0.4878 & -0.1426 & -0.6542 \end{pmatrix}, \ \mathbf{\Sigma}_2^{-1} = \begin{pmatrix} 1.7975 & 0 & 0.1025 & 0 \\ 0 & 3.1785 & 0.8908 & 0.5532 \\ 0.1025 & 0.8908 & 0.4838 & 0.0586 \\ 0 & 0.5532 & 0.0586 & 4.1300 \end{pmatrix},$$

$$\mathbf{A}_1^{(3)} = \begin{pmatrix} 0.4352 & -0.6552 & 0.4154 & 0.3930 & -0.5200 & 0.2256 \\ 0.1478 & -0.4932 & 0 & 0 & 0 & 0 \\ -0.7940 & 0 & -0.8933 & 0 & 0 & 0 \\ 0.5894 & 0 & 0 & -0.1478 & 0 & 0 \\ -0.8009 & 0 & 0 & 0 & -0.4169 & 0 \\ 0.4197 & 0 & 0 & 0 & 0 & -0.2439 \end{pmatrix}, \ \mathbf{\Sigma}_3^{-1} = \begin{pmatrix} 1 & 0.4 & 0.4 & 0.4 & 0.4 & 0.4 \\ 0.4 & 1 & 0 & 0 & 0 & 0 \\ 0.4 & 0 & 1 & 0 & 0 & 0 \\ 0.4 & 0 & 0 & 1 & 0 & 0 \\ 0.4 & 0 & 0 & 0 & 1 & 0 \\ 0.4 & 0 & 0 & 0 & 0 & 1 \end{pmatrix},$$

$$\mathbf{A}_1^{(4)} = \begin{pmatrix} 0.2177 & 0.3066 & 0 & 0 & 0 & 0.3775 \\ -0.6324 & -0.6650 & 0.0214 & 0 & 0 & 0 \\ 0 & -0.2749 & -0.7509 & 0.4482 & 0 & 0 \\ 0 & 0 & -0.3046 & -0.8066 & 0.9940 & 0 \\ 0 & 0 & 0 & -0.7313 & 0.5054 & 0.7959 \\ -0.0587 & 0 & 0 & 0 & -0.5140 & -0.9470 \end{pmatrix}, \ \mathbf{\Sigma}_4^{-1} = \begin{pmatrix} 1 & 0.4 & 0 & 0 & 0 & 0.4 \\ 0.4 & 1 & 0.4 & 0 & 0 & 0 \\ 0 & 0.4 & 1 & 0.4 & 0 & 0 \\ 0 & 0 & 0.4 & 1 & 0.4 & 0 \\ 0 & 0 & 0 & 0.4 & 1 & 0.4 \\ 0.4 & 0 & 0 & 0 & 0.4 & 1 \end{pmatrix},$$

$$\mathbf{A}_1^{(5)} = \begin{pmatrix} -0.6 & 0.4 & 0 & 0 & 0 & 0.4 \\ 0.4 & -0.6 & 0.4 & 0 & 0 & 0 \\ 0 & 0.4 & -0.6 & 0.4 & 0 & 0 \\ 0 & 0 & 0.4 & -0.6 & 0.4 & 0 \\ 0 & 0 & 0 & 0.4 & -0.6 & 0.4 \\ 0.4 & 0 & 0 & 0 & 0.4 & -0.6 \end{pmatrix}, \ \mathbf{A}_2^{(5)} = \begin{pmatrix} -0.3 & 0.2 & 0 & 0 & 0 & 0.2 \\ 0.2 & -0.3 & 0.2 & 0 & 0 & 0 \\ 0 & 0.2 & -0.3 & 0.2 & 0 & 0 \\ 0 & 0 & 0.2 & -0.3 & 0.2 & 0 \\ 0 & 0 & 0 & 0.2 & -0.3 & 0.2 \\ 0.2 & 0 & 0 & 0 & 0.2 & -0.3 \end{pmatrix} \text{ and}$$

$$\mathbf{\Sigma}_5^{-1} = \begin{pmatrix} 1 & -0.3 & 0 & 0 & 0 & -0.3 \\ -0.3 & 1 & -0.3 & 0 & 0 & 0 \\ 0 & -0.3 & 1 & -0.3 & 0 & 0 \\ 0 & 0 & -0.3 & 1 & -0.3 & 0 \\ 0 & 0 & 0 & -0.3 & 1 & -0.3 \\ -0.3 & 0 & 0 & 0 & -0.3 & 1 \end{pmatrix}.$$

Model 1 is a stable VAR(1) model of dimension three. The second and the third components of the series are Granger non-causal and contemporaneously independent (i.e. $\left(\mathbf{A}_1^{(4)}\right)_{23} = \left(\mathbf{A}_1^{(4)}\right)_{32} = 0$ and $\left(\mathbf{\Sigma}_1^{-1}\right)_{23} = \left(\mathbf{\Sigma}_1^{-1}\right)_{32} = 0$). Model 2 is a 4-dimensional VAR(1) model, where the first and second; and the first and fourth components of the multivariate time series are Granger non-causal and contemporaneous conditionally independent. We further extend the investigation of the proposed method to some higher dimension stable VAR models. Model 3 is a 6-dimensional stable

VAR(1) model with every node connected to the first node in the mixed graph. Model 4 is a stable VAR(1) model in which both the autoregressive coefficient matrix and the inverse noise covariance matrix have a Toeplitz structure. To explore the performance of the estimation method on VAR model with higher lag order p, we consider a 6-dimensional VAR(2) model with Toeplitz autoregressive coefficient matrices and Toeplitz inverse noise covariance matrix.

The experiments are carried out with sample size T of 100, 200, 500, 1000 over 500 replications using MATLAB R2016b on a Linux based workstation with two 2.1 GHz CPUs and 503 GB main memory. We use SDPT3 (Tütüncü et al., 2003) to estimate the inverse noise covariance matrix in the experiments. SDPT3 is a MATLAB based convex optimization tool for solving the semidefinite programming problem. As a comparison to the alternating maximization method, we also solve the optimization problem, assuming the true sparsity structure is known, by two widely used algorithms in nonlinear optimization. They are the interior point algorithm and the direct search method by the MATLAB command 'fmincon' and 'patternsearch', respectively. We impose the positive definiteness constraint, in the two comparison methods, based on the fact that the leading principal minors of the inverse covariance matrix are positive. The following metrics are computed for comparisons: the bias of the AR coefficient estimates,

$$\text{Bias} = \sum_{l=1}^{p} \sum_{i,j=1}^{K} \left| E\left(\left(\hat{\mathbf{A}}_l \right)_{i,j} \right) - (\mathbf{A}_l)_{ij} \right|;$$

the variance of the AR coefficient estimates,

$$\text{Variance} = \sum_{l=1}^{p} \sum_{i,j=1}^{K} \text{Var}\left[\left(\hat{\mathbf{A}}_l \right)_{ij} \right];$$

and the mean squared error (MSE) of the AR coefficient estimates,

$$\text{MSE} = \sum_{l=1}^{p} \sum_{i,j=1}^{K} \left\{ \left[E\left(\left(\hat{\mathbf{A}}_l \right)_{ij} \right) - \left(\mathbf{A}_l \right)_{ij} \right]^2 + \text{Var} \left[\left(\hat{\mathbf{A}}_l \right)_{ij} \right] \right\}.$$

For the inverse noise covariance estimates, the upper triangular part of the estimates is considered in the computation of three metrics since the estimates are symmetric.

We also perform the simulation experiments with unknown structure (including the lag order) and estimate the structure using the frequency domain and time domain methods described in Section 3.1.2. The metrics are modified to account for the error incurred in selecting a wrong lag order. That is, the p in the above formulas are changed to be the maximum between the determined lag order and the true lag order, and $\left(\mathbf{A}_l \right)_{ij}$ is defined to be zero whenever $l > p$.

Simulation with known structure

Tables 3.1–3.5 document the bias, variance and mean squared error (MSE) of the estimates using the three mentioned algorithms (the alternating maximization method, the interior-point method, and the direct search method) for the five studied models. These three metrics are compiled using the simulation results whenever the corresponding algorithm converges. The columns 'T', 'Method', 'NC', 'Cputime' and 'Iterations' are, respectively, the sample size, the optimization method used, the number of incomplete experiments, due to non-convergence, out of 500 replications, the average CPU time consumed in seconds and the average number of iterations involved in solving the problem. The value in parenthesis is the standard deviation of the corresponding measurement. We consider completion of

the **B** step followed by the **Θ** step as one iteration of the alternating maximization method.

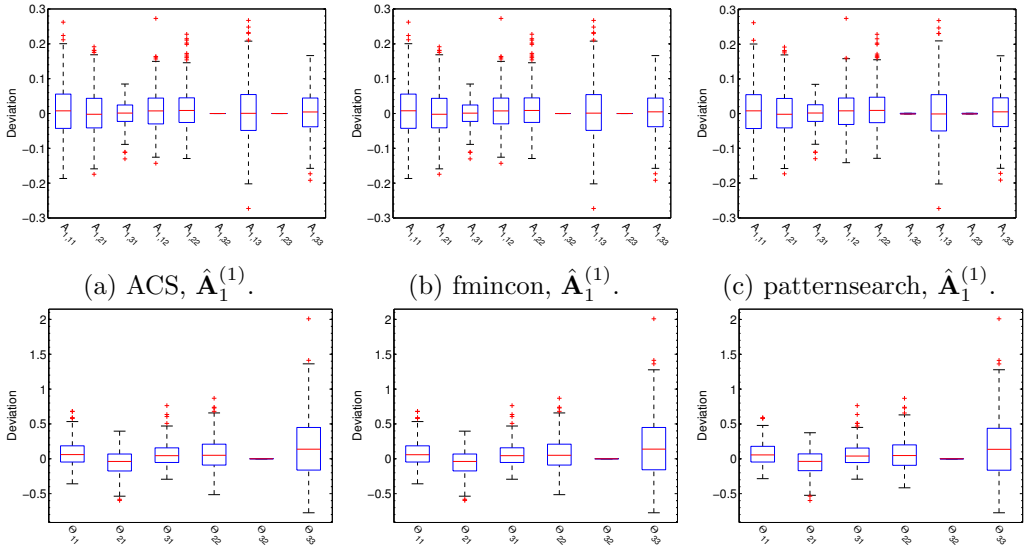
Model 1

TABLE 3.1: Simulation results for Model 1 over 500 replications. The metrics are compiled using the results whenever the corresponding algorithm converges. NC indicates the number of incomplete experiments, Cputime is the average CPU time consumed in seconds, and Iteration is the average number of iterations involved. Standard deviations are in the parentheses.

					$\hat{\mathbf{A}}$			$\hat{\mathbf{\Sigma}}_u^{-1}$		
T	Method	NC	Cputime	Iterations	Bias	Variance	MSE	Bias	Variance	MSE
100	ACS	0	2.3934 (0.8676)	4.1700 (0.6147)	0.0387	0.0280	0.0284	0.4284	0.3047	0.3498
	fmincon	3	10.5573 (0.9191)	30.4064 (1.5358)	0.0392	0.0279	0.0283	0.4297	0.3046	0.3501
	patternsearch	22	1598.7132 (686.5228)	9.0732 (0.2687)	0.0382	0.0280	0.0284	0.4055	0.2937	0.3346
200	ACS	0	2.1998 (0.7500)	3.7740 (0.5133)	0.0229	0.0141	0.0142	0.2831	0.1323	0.1520
	fmincon	3	10.9860 (1.0686)	30.8008 (1.6346)	0.0223	0.0141	0.0142	0.2814	0.1325	0.1520
	patternsearch	4	1555.4650 (518.1913)	9.0444 (0.2061)	0.0227	0.0141	0.0142	0.2812	0.1320	0.1516
500	ACS	0	2.3436 (1.0875)	3.3520 (0.4781)	0.0152	0.0055	0.0056	0.0944	0.0471	0.0494
	fmincon	2	12.4463 (1.4386)	31.8514 (1.7028)	0.0151	0.0055	0.0055	0.0938	0.0472	0.0494
	patternsearch	0	1681.1368 (456.9749)	9.0320 (0.1762)	0.0154	0.0055	0.0056	0.0948	0.0471	0.0494
1000	ACS	0	1.8870 (0.5567)	3.1000 (0.3003)	0.0060	0.0026	0.0026	0.0455	0.0258	0.0262
	fmincon	6	12.9251 (1.5171)	32.6377 (1.9756)	0.0061	0.0026	0.0026	0.0443	0.0256	0.0260
	patternsearch	0	1493.6605 (280.7624)	9.0240 (0.1532)	0.0060	0.0026	0.0026	0.0456	0.0258	0.0262

Table 3.1 is the simulation results for Model 1 using the three studied methods, namely the alternating maximization method (denoted by 'ACS'), the interior-point algorithm (denoted by 'fmincon') and the direct search method (denoted by 'patternsearch'). Few simulation experiments using the interior-point method do not converge successfully. The direct search method terminates before obtaining a solution in some of the experiments, especially when the sample size is low. The alternating maximization method consumes less CPU time comparing to the two other methods, while the direct search method spends the most. The average number of iterations for the ACS and the direct search methods decreases as the sample size increases. The three metrics (bias, variance, and MSE) for both the AR

coefficient and the inverse covariance estimates drop steadily as the sample size increases for the three studied methods. We also generate boxplots of deviations of the estimates (i.e., $\hat{\theta} - \theta$) to gain a better insight into the dispersion of the estimates for each method.



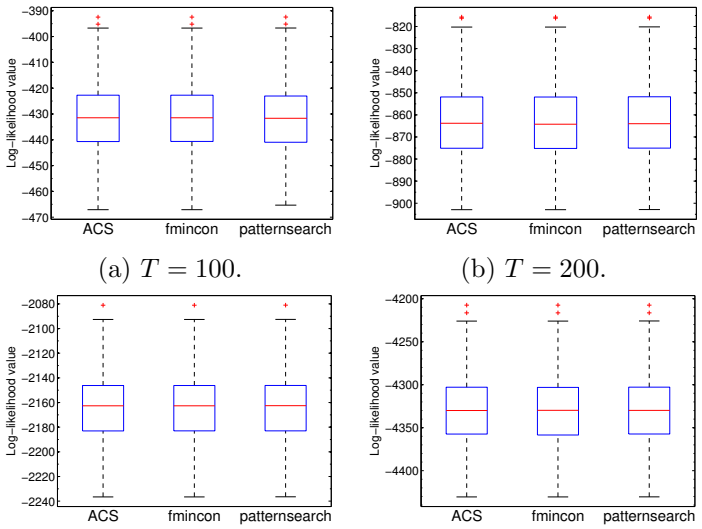
(a) ACS, $\hat{\mathbf{A}}_1^{(1)}$. (b) fmincon, $\hat{\mathbf{A}}_1^{(1)}$. (c) patternsearch, $\hat{\mathbf{A}}_1^{(1)}$.

(d) ACS, $\hat{\mathbf{\Sigma}}_1^{-1}$. (e) fmincon, $\hat{\mathbf{\Sigma}}_1^{-1}$. (f) patternsearch, $\hat{\mathbf{\Sigma}}_1^{-1}$.

FIGURE 3.1: Boxplot of deviations of the estimates for Model 1 when $T = 100$.

Figure 3.1 depicts boxplots of deviations of the AR coefficient (on the upper panel), and the inverse covariance (on the lower panel) estimates for Model 1 when $T = 100$. The deviations are computed whenever the corresponding algorithm converges. We can observe from the boxplots that all algorithms restrict the corresponding AR coefficient and inverse covariance estimates to zero. For the unconstrained estimates, both three algorithms obtain estimates that possess similar dispersion. We next consider the log-likelihood values to investigate the convergence properties of the three studied algorithms.

Figure 3.2 shows boxplots of the log-likelihood values, computed using the obtained AR coefficient and inverse noise covariance estimates, for the three investigated methods with different sample sizes. It is observed that



(a) $T = 100$. (b) $T = 200$.

(c) $T = 500$. (d) $T = 1000$.

FIGURE 3.2: Boxplot of loglikelihood values for Model 1.

the three algorithms obtain similar log-likelihood values, and the average log-likelihood values are less dispersed as the sample size increases. The results indicate that the log-likelihood values obtained from these three methods converge to some values that are close to each other, whenever the methods converge.

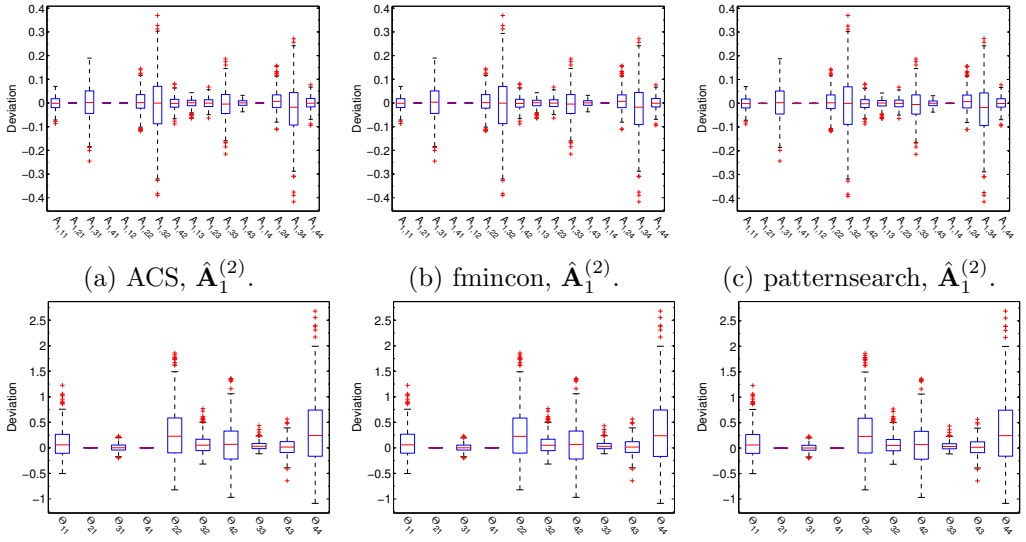
Model 2

Table 3.2 is the simulation results for Model 2 using the methods mentioned above, namely the alternating maximization method (denoted by 'ACS'), the interior-point algorithm (denoted by 'fmincon') and the direct search method (denoted by 'patternsearch'). Few simulation experiments using the interior-point method do not converge successfully. The direct search method can obtain a solution for all experiments. The alternating maximization method consumes less CPU time comparing to the two other methods, while the direct search method spends the most. The average number of iterations for the ACS method decreases as the sample size increases. The three metrics (bias, variance, and MSE) for both the AR coefficient

TABLE 3.2: Simulation results for Model 2 over 500 replications. The metrics are compiled using the results whenever the corresponding algorithm converges. NC indicates the number of incomplete experiments, Cputime is the average CPU time consumed in seconds, and Iteration is the average number of iterations involved. Standard deviations are in the parentheses.

					$\hat{\mathbf{A}}$			$\hat{\mathbf{\Sigma}}_u^{-1}$		
T	Method	NC	Cputime	Iterations	Bias	Variance	MSE	Bias	Variance	MSE
100	ACS	0	2.0978 (0.2157)	4.5740 (0.6674)	0.0639	0.0420	0.0428	0.9019	1.0303	1.2401
	fmincon	6	19.7053 (1.6192)	46.3907 (2.0270)	0.0646	0.0416	0.0424	0.8936	1.0316	1.2374
	patternsearch	0	1157.4870 (464.8214)	8.8020 (0.3989)	0.0640	0.0419	0.0428	0.9081	1.0316	1.2443
200	ACS	0	2.1394 (0.1960)	4.0420 (0.4740)	0.0375	0.0184	0.0188	0.3747	0.4274	0.4644
	fmincon	9	21.0697 (2.0892)	47.0428 (2.0633)	0.0382	0.0184	0.0188	0.3737	0.4277	0.4643
	patternsearch	0	1108.8456 (370.5133)	8.8300 (0.3760)	0.0377	0.0185	0.0188	0.3788	0.4277	0.4655
500	ACS	0	1.9902 (0.1787)	3.6020 (0.4940)	0.0207	0.0086	0.0086	0.1515	0.1640	0.1706
	fmincon	3	21.9270 (2.1989)	47.9779 (2.3473)	0.0209	0.0086	0.0087	0.1492	0.1633	0.1697
	patternsearch	0	956.3934 (259.5371)	8.8280 (0.3778)	0.0208	0.0086	0.0086	0.1540	0.1641	0.1709
1000	ACS	0	1.8614 (0.1828)	3.2380 (0.4263)	0.0090	0.0037	0.0038	0.0651	0.0833	0.0843
	fmincon	5	22.8292 (1.8971)	48.8465 (2.0980)	0.0091	0.0038	0.0038	0.0661	0.0836	0.0847
	patternsearch	0	876.7742 (229.0570)	8.8040 (0.3974)	0.0091	0.0038	0.0038	0.0669	0.0833	0.0844

and the inverse covariance estimates decline gradually as the sample size raises for the three studied methods. We plot boxplots of deviations of the estimates to investigate the dispersion of the estimates for each method.

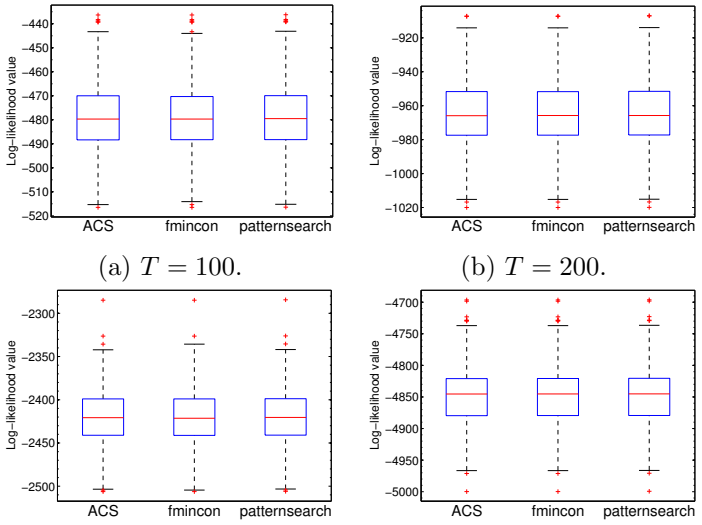


(a) ACS, $\hat{\mathbf{A}}_1^{(2)}$. (b) fmincon, $\hat{\mathbf{A}}_1^{(2)}$. (c) patternsearch, $\hat{\mathbf{A}}_1^{(2)}$.

(d) ACS, $\hat{\mathbf{\Sigma}}_2^{-1}$. (e) fmincon, $\hat{\mathbf{\Sigma}}_2^{-1}$. (f) patternsearch, $\hat{\mathbf{\Sigma}}_2^{-1}$.

FIGURE 3.3: Boxplot of deviations of the estimates for Model 2 when $T = 100$.

Figure 3.3 depicts boxplots of deviations of the AR coefficient (on the upper panel), and the inverse covariance (on the lower panel) estimates for Model 2 when $T = 100$, respectively. We can observe from the boxplots that all algorithms constrain the corresponding AR coefficients and inverse covariance estimates to zero. For the unconstrained estimates, both three methods obtain estimates that possess similar dispersion. We next consider the log-likelihood values to explore the convergence properties of the three studied algorithms.



(a) $T = 100$. (b) $T = 200$.

(c) $T = 500$. (d) $T = 1000$.

FIGURE 3.4: Boxplot of log-likelihood values for Model 2.

Figure 3.4 is boxplots of the log-likelihood values, computed using the obtained AR coefficient and inverse noise covariance estimates, for the three investigated methods. It is observed that the three methods obtain similar log-likelihood values, and the average log-likelihood values become less disperse as the sample size raises. The results suggest that the three studied algorithms converge to some log-likelihood values that are close to each other, whenever the methods converge.

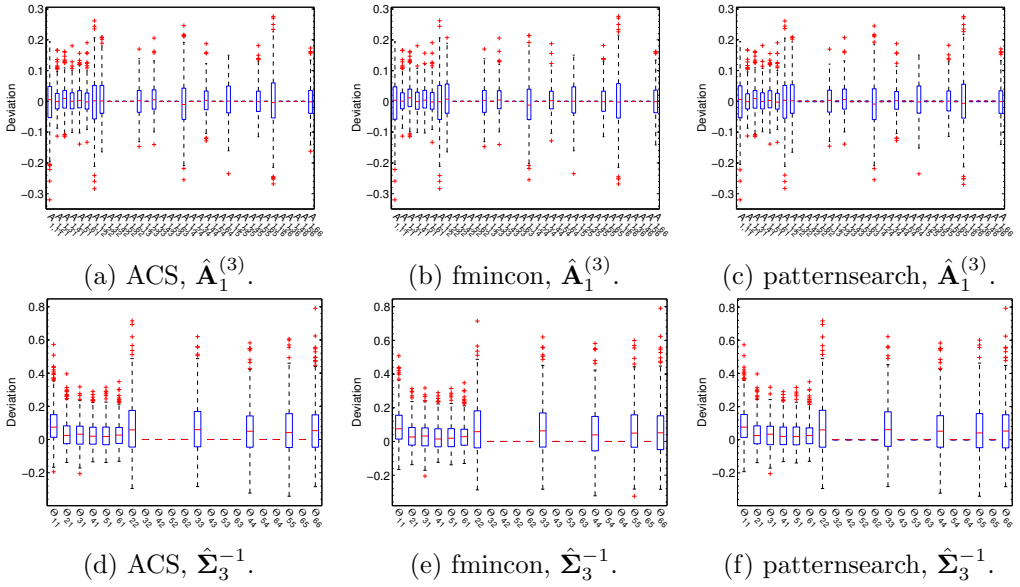
Model 3

TABLE 3.3: Simulation results for Model 3 over 500 replications. The metrics are compiled using the results whenever the corresponding algorithm converges. NC indicates the number of incomplete experiments, Cputime is the average CPU time consumed in seconds, and Iteration is the average number of iterations involved. Standard deviation are in the parentheses.

					$\hat{\mathbf{A}}$			$\hat{\mathbf{\Sigma}}_u^{-1}$		
T	Method	NC	Cputime	Iterations	Bias	Variance	MSE	Bias	Variance	MSE
100	ACS	0	2.8861 (0.2478)	5.9700 (0.6588)	0.0654	0.0574	0.0578	0.5558	0.1676	0.2003
	fmincon	108	48.7554 (6.3255)	62.6556 (2.0657)	0.0779	0.0576	0.0582	0.5717	0.1675	0.2022
	patternsearch	58	5330.8996 (1468.9006)	8.8507 (0.3568)	0.0645	0.0580	0.0584	0.5552	0.1683	0.2009
200	ACS	0	2.5836 (0.2224)	5.0420 (0.4250)	0.0413	0.0268	0.0270	0.2798	0.0759	0.0843
	fmincon	128	48.1210 (3.9664)	64.8763 (2.2798)	0.0511	0.0272	0.0275	0.2890	0.0779	0.0868
	patternsearch	13	4744.7387 (1022.6095)	8.8665 (0.3404)	0.0462	0.0267	0.0270	0.2821	0.0756	0.0842
500	ACS	0	2.2662 (0.2193)	4.1800 (0.3846)	0.0276	0.0104	0.0104	0.1098	0.0293	0.0307
	fmincon	160	50.8929 (3.7852)	66.6559 (2.4858)	0.0246	0.0106	0.0106	0.1193	0.0300	0.0314
	patternsearch	0	4455.3172 (711.3365)	8.8740 (0.3322)	0.0287	0.0104	0.0105	0.1136	0.0294	0.0308
1000	ACS	0	2.3402 (0.3563)	3.9920 (0.1094)	0.0072	0.0053	0.0053	0.0516	0.0145	0.0148
	fmincon	202	67.2302 (18.4431)	67.5201 (2.7392)	0.0104	0.0054	0.0054	0.0471	0.0147	0.0150
	patternsearch	0	6027.8816 (2050.3609)	8.8940 (0.3081)	0.0082	0.0053	0.0053	0.0543	0.0146	0.0149

Table 3.3 is the simulation results for Model 3 using the three explored methods, namely the alternating maximization method (denoted by 'ACS'), the interior-point algorithm (denoted by 'fmincon') and the direct search method (denoted by 'patternsearch'). Some simulation experiments using the interior-point method do not converge successfully, and more non-convergence cases occur when the sample size raises. The direct search method does not obtain a solution in some experiments, especially when the sample size is low. This situation is alleviated as the sample size increases. The alternating maximization method consumes less CPU time comparing to the two other methods, while the direct search method spends the most. The average number of iterations for the ACS method decreases as the sample size increases. The three metrics (bias, variance, and MSE) for both the AR coefficient and the inverse covariance estimates decline gradually as the

sample size raises for the three studied methods. We plot boxplots of deviations of the estimates to investigate the dispersion of the estimates for each method.



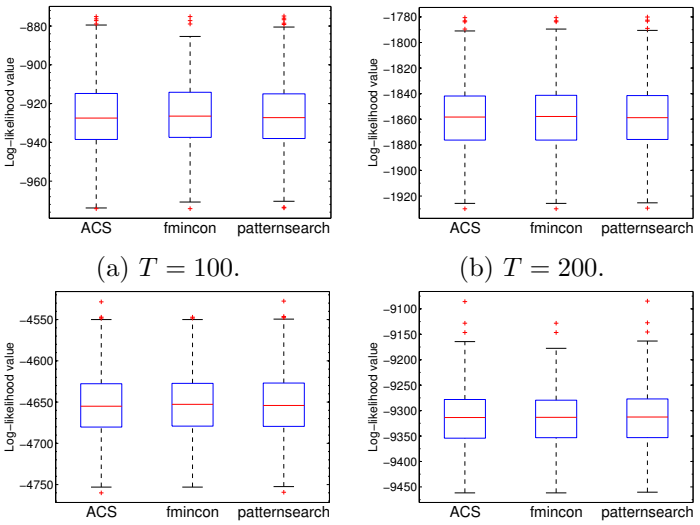
(a) ACS, $\hat{\mathbf{A}}_1^{(3)}$. (b) fmincon, $\hat{\mathbf{A}}_1^{(3)}$. (c) patternsearch, $\hat{\mathbf{A}}_1^{(3)}$.

(d) ACS, $\hat{\mathbf{\Sigma}}_3^{-1}$. (e) fmincon, $\hat{\mathbf{\Sigma}}_3^{-1}$. (f) patternsearch, $\hat{\mathbf{\Sigma}}_3^{-1}$.

FIGURE 3.5: Boxplot of deviations of the estimates for Model 3 when $T = 100$.

Figure 3.5 shows boxplots of deviations of the AR coefficient (on the upper panel) and the inverse covariance (on the lower panel) estimates for Model 3 when $T = 100$. We can observe from the boxplots that all algorithms restrict the corresponding AR coefficient and inverse covariance estimates to zero. Both three methods obtain unconstrained estimates that possess similar dispersion. Furthermore, the estimates with larger true parameter values are more dispersed. We next consider the log-likelihood values to explore the convergence properties of the three studied algorithms.

Figure 3.6 displays boxplots of the log-likelihood values, computed using the obtained AR coefficient and inverse noise covariance estimates, for the three investigated methods. We can observe from the boxplots that all three methods obtain similar log-likelihood values, and the average log-likelihood values are less dispersed as the sample size raises. The findings suggests



(a) $T = 100$. (b) $T = 200$.

(c) $T = 500$. (d) $T = 1000$.

FIGURE 3.6: Boxplot of the log-likelihood values for Model 3.

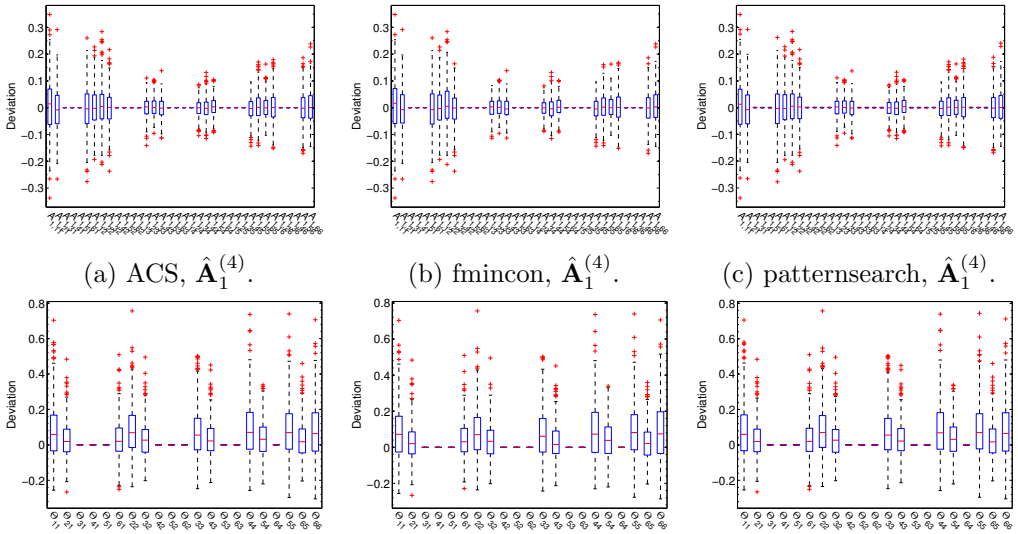
that the three algorithms converge to some log-likelihood values that are close to each other, whenever the methods converge.

Model 4

TABLE 3.4: Simulation results for Model 4 over 500 replications. The metrics are compiled using the results whenever the corresponding algorithm converges. NC indicates the number of incomplete experiments, Cputime is the average CPU time consumed in seconds, and Iteration is the average number of iterations involved. Standard deviations are in the parentheses.

					$\hat{\mathbf{A}}$			$\hat{\mathbf{\Sigma}}_u^{-1}$		
T	Method	NC	Cputime	Iterations	Bias	Variance	MSE	Bias	Variance	MSE
100	ACS	0	2.4622 (0.9674)	5.3380 (0.5517)	0.0569	0.0600	0.0603	0.6600	0.1999	0.2432
	fmincon	167	59.1187 (18.9916)	68.5826 (2.2028)	0.0753	0.0627	0.0632	0.7176	0.2052	0.2561
	patternsearch	3	3022.3045 (1159.8824)	8.0141 (0.1180)	0.0577	0.0602	0.0604	0.6658	0.2005	0.2445
200	ACS	0	2.3448 (0.8252)	4.6520 (0.5014)	0.0313	0.0297	0.0298	0.3118	0.0900	0.0997
	fmincon	182	60.2929 (21.7626)	70.2987 (2.1597)	0.0418	0.0296	0.0298	0.3459	0.0881	0.1001
	patternsearch	0	2862.6623 (881.2467)	8.0040 (0.0632)	0.0328	0.0297	0.0298	0.3201	0.0903	0.1006
500	ACS	0	2.0152 (0.5884)	4.0160 (0.1407)	0.0196	0.0124	0.0124	0.1207	0.0337	0.0352
	fmincon	206	68.8011 (20.4213)	72.4048 (2.7407)	0.0198	0.0124	0.0124	0.1413	0.0337	0.0358
	patternsearch	0	3026.0082 (1031.8243)	8.0000 (0.0000)	0.0207	0.0124	0.0124	0.1256	0.0338	0.0354
1000	ACS	0	1.9544 (0.6206)	3.9380 (0.2414)	0.0095	0.0059	0.0059	0.0514	0.0161	0.0163
	fmincon	244	63.3283 (12.1464)	73.6055 (3.0196)	0.0117	0.0058	0.0058	0.0590	0.0162	0.0166
	patternsearch	0	2554.6416 (650.7686)	8.0000 (0.0000)	0.0102	0.0059	0.0059	0.0542	0.0161	0.0164

Table 3.4 is the simulation results for Model 4 using the three investigated methods, namely the alternating maximization method (denoted by 'ACS'), the interior-point algorithm (denoted by 'fmincon') and the direct search method (denoted by 'patternsearch'). Some simulation experiments using the interior-point method do not converge successfully, and more non-convergence cases occur when the sample size raises. Few simulation experiments do not obtain a solution using the direct search method before termination when the sample size is 100. The alternating maximization method consumes less CPU time comparing to the two other methods, while the direct search method spends the most. The average number of iterations for the ACS method decreases as the sample size increases. The three metrics (bias, variance, and MSE) for both the AR coefficient and the inverse covariance estimates decline gradually as the sample size raises for the three studied methods. We plot boxplots of deviations of the estimates to investigate the dispersion of the estimates for each method.



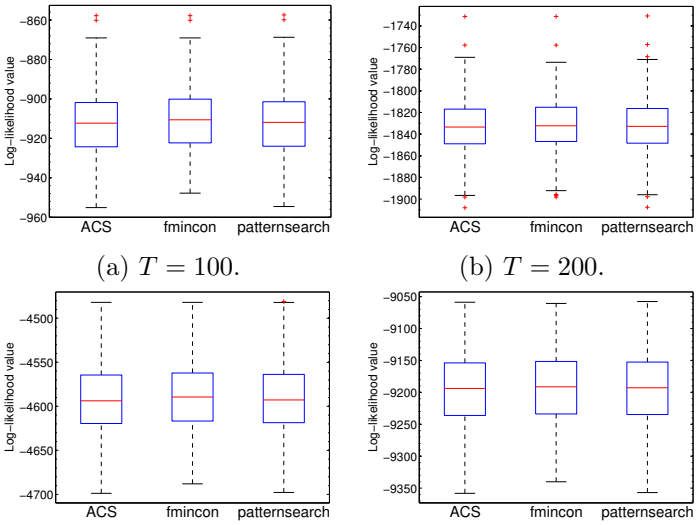
(a) ACS, $\hat{\mathbf{A}}_1^{(4)}$. (b) fmincon, $\hat{\mathbf{A}}_1^{(4)}$. (c) patternsearch, $\hat{\mathbf{A}}_1^{(4)}$.

(d) ACS, $\hat{\mathbf{\Sigma}}_4^{-1}$. (e) fmincon, $\hat{\mathbf{\Sigma}}_4^{-1}$. (f) patternsearch, $\hat{\mathbf{\Sigma}}_4^{-1}$.

FIGURE 3.7: Boxplot of deviations of the AR coefficient estimates for Model 4 when $T = 100$.

Figure 3.7 shows boxplots of deviations of the AR coefficient (on the upper panel) and the inverse covariance (on the lower panel) estimates for

Model 4 when $T = 100$, respectively. We can observe from the boxplot that all algorithms restrict the corresponding AR coefficient and inverse covariance estimates to zero. For the unconstrained estimates, both three methods obtain estimates that carry similar dispersion. Besides, the estimates with larger true parameter value are more disperse. We next consider the log-likelihood values to investigate the convergence properties of the three studied algorithms.



(a) $T = 100$. (b) $T = 200$.

(c) $T = 500$. (d) $T = 1000$.

FIGURE 3.8: Boxplot of the log-likelihood values for Model 4.

Figure 3.8 is boxplots of the log-likelihood values, computed using the obtained AR coefficient and inverse noise covariance estimates, for the three investigated methods. We can observe from the boxplots that the three algorithms obtain similar log-likelihood values, and the average log-likelihood values are less dispersed as the sample size increases. The results indicate that the three algorithms converge to some log-likelihood values that are close to each other, whenever the methods converge.

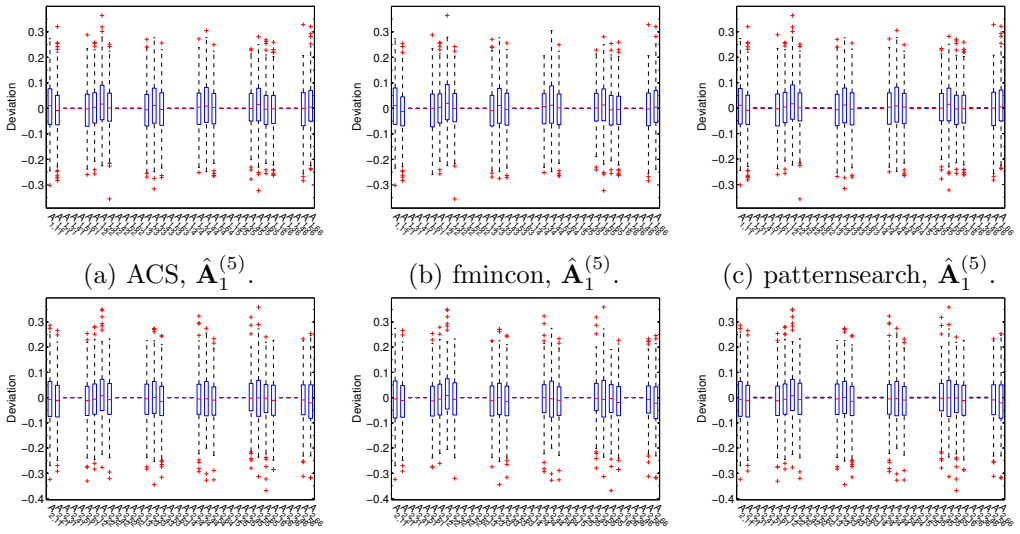
Model 5

TABLE 3.5: Simulation results for Model 5 over 500 replications. The metrics are compiled using the results whenever the corresponding algorithm converges. NC indicates the number of incomplete experiments, Cputime is the average CPU time consumed in seconds, and Iteration is the average number of iterations involved. Standard deviations are in the parentheses.

					$\hat{\mathbf{A}}$			$\hat{\mathbf{\Sigma}}_u^{-1}$		
T	Method	NC	Cputime	Iterations	Bias	Variance	MSE	Bias	Variance	MSE
100	ACS	0	2.8934 (0.2696)	6.2340 (0.6066)	0.2682	0.3019	0.3047	0.8525	0.2329	0.3094
	fmincon	120	97.0799 (9.4558)	94.6684 (8.3720)	0.3044	0.3033	0.3068	0.8943	0.2338	0.3179
	patternsearch	0	4851.2951 (268.9346)	8.0000 (0.0000)	0.2722	0.3019	0.3048	0.8653	0.2338	0.3126
200	ACS	0	2.6351 (0.2383)	5.1580 (0.4067)	0.1455	0.1461	0.1469	0.3766	0.0967	0.1122
	fmincon	123	104.6940 (7.4198)	99.8806 (5.9679)	0.1694	0.1470	0.1481	0.3925	0.0982	0.1151
	patternsearch	0	4616.1554 (340.5639)	8.0000 (0.0000)	0.1478	0.1462	0.1470	0.3850	0.0970	0.1131
500	ACS	0	2.3304 (0.2798)	4.2760 (0.4475)	0.0640	0.0585	0.0587	0.1597	0.0362	0.0389
	fmincon	127	118.6068 (25.2486)	101.4236 (6.9447)	0.0732	0.0587	0.0590	0.1730	0.0359	0.0392
	patternsearch	0	4859.9765 (1098.6708)	8.0000 (0.0000)	0.0662	0.0585	0.0587	0.1648	0.0362	0.0392
1000	ACS	0	2.2658 (0.1815)	4.0000 (0.0000)	0.0390	0.0291	0.0291	0.0699	0.0174	0.0179
	fmincon	145	147.1842 (12.9014)	99.3380 (7.9624)	0.0355	0.0286	0.0286	0.0683	0.0173	0.0178
	patternsearch	0	5945.0423 (250.1966)	8.0000 (0.0000)	0.0409	0.0291	0.0292	0.0737	0.0174	0.0180

Table 3.5 documents the simulation results for Model 5 using the three studied algorithms, namely the alternating maximization method (denoted by 'ACS'), the interior-point algorithm (denoted by 'fmincon') and the direct search method (denoted by 'patternsearch'). Some simulation experiments using the interior-point method do not converge successfully. The direct search method obtains a solution in all simulation experiments. The alternating maximization method consumes less CPU time comparing to the two other methods, while the direct search method spends the most. The average computation time and the average number of iterations for the ACS method decrease as the sample size increases. The three metrics (bias, variance, and MSE) for both the AR coefficient and the inverse covariance estimates decline gradually as the sample size raises for the three studied

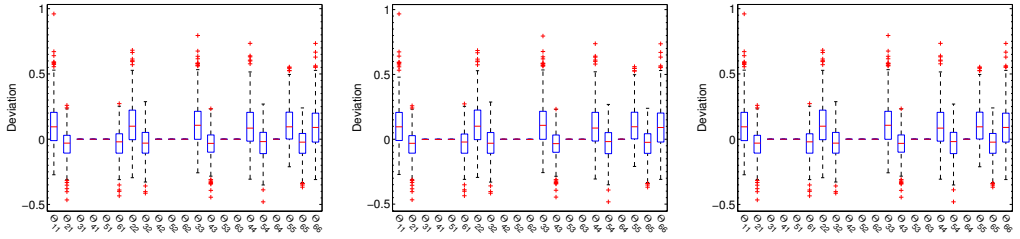
methods. We plot boxplots of deviations of the estimates to investigate the dispersion of the estimates for each method.



(a) ACS, $\hat{\mathbf{A}}_1^{(5)}$. (b) fmincon, $\hat{\mathbf{A}}_1^{(5)}$. (c) patternsearch, $\hat{\mathbf{A}}_1^{(5)}$.

(d) ACS, $\hat{\mathbf{A}}_2^{(5)}$. (e) fmincon, $\hat{\mathbf{A}}_2^{(5)}$. (f) patternsearch, $\hat{\mathbf{A}}_2^{(5)}$.

FIGURE 3.9: Boxplot of deviations of the estimates for Model 5 when $T = 100$.

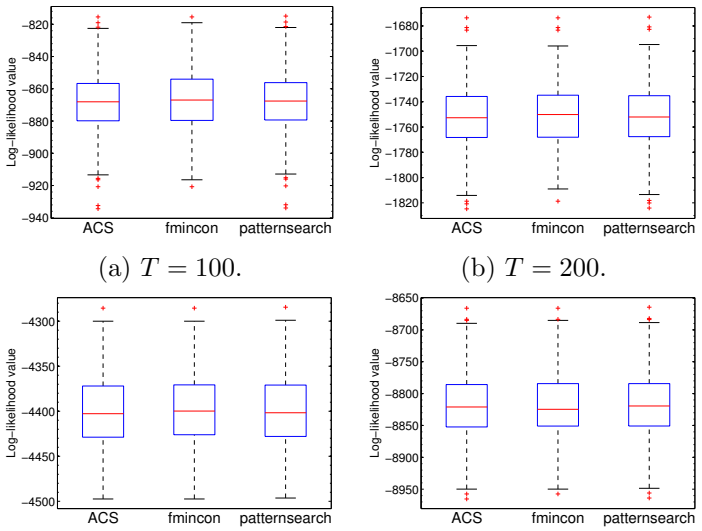


(a) ACS, $\hat{\mathbf{\Sigma}}_5^{-1}$. (b) fmincon, $\hat{\mathbf{\Sigma}}_5^{-1}$. (c) patternsearch, $\hat{\mathbf{\Sigma}}_5^{-1}$.

FIGURE 3.10: Boxplot of deviations of the inverse covariance estimates for Model 5 when $T = 100$.

Figure 3.9 displays boxplots of deviations of the AR coefficient of lag 1 (on the upper panel), and the lag 2 AR coefficient (on the lower panel) estimates for Model 5 when $T = 100$. Figure 3.10 is boxplots of deviations of the inverse covariance estimates when $T = 100$. We can observe from the two figures that all algorithms restrict the corresponding AR coefficient and inverse covariance estimates to zero. For the unconstrained estimates, both three methods obtain estimates that carry similar dispersion. Furthermore, the estimates with larger true parameter values are more dispersed. We

next explore the convergence properties of the three studied algorithms by considering the log-likelihood values.



(a) $T = 100$. (b) $T = 200$.

(c) $T = 500$. (d) $T = 1000$.

FIGURE 3.11: Boxplot of the log-likelihood values for Model 5.

Figure 3.11 displays boxplots of the log-likelihood values, computed using the obtained AR coefficient and inverse noise covariance estimates, for the three investigated methods with various sample sizes. It is observed that the ACS and the direct search methods obtain similar log-likelihood values, while the interior point method is slightly different. This is because the interior point method does not obtain a solution in many simulation experiments. We can also see from the figure that the variability of the average log-likelihood values decreases when the sample size raises. The results suggest that the log-likelihood values obtained from these three methods converge to some values that are close to each other, whenever the methods converge.

In summary, the simulation results reflect that the alternating maximization method is more robust and is rare to obtain a solution that has a significant deviation from the actual parameter. The other two methods,

however, fail to converge in some cases, especially when the number of parameters is large. It seems that the alternating method has an advantage that it always converges while the other two methods may not. The alternating method consumes less CPU time to obtain a solution comparing to the other two algorithms. The results obtained using the alternating method are similar to that acquired by the other two methods whenever these methods converge.

Simulation with unknown structure

Tables 3.6–3.10 document the bias, variance and mean squared error (MSE) of the estimates using the frequency domain and the time domain methods introduced in Section 3.1.2, assuming the lag order and sparsity structure are unknown. These three metrics are compiled using all simulation results. The columns 'T', 'Method', 'Cputime' and '$\hat{p}$' are, respectively, the sample size, the algorithm applied, the average CPU time consumed in seconds and the average lag order determined. For the inverse noise covariance estimate, the upper triangular part of the estimate is considered in computing the three metrics (bias, variance, and MSE). The value in parenthesis is the standard deviation of the corresponding measurement.

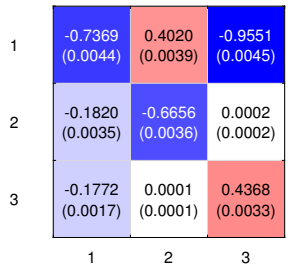
Model 1

TABLE 3.6: Simulation results for Model 1 over 500 replications. $\hat{p}$ is the average lag order determined, and Cputime is the average CPU time consumed in seconds. Standard deviations are in parenthesis.

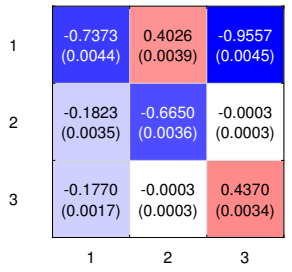
				$\hat{\mathbf{A}}$			$\hat{\mathbf{\Sigma}}_u^{-1}$		
T	Method	Cputime	$\hat{p}$	Bias	Variance	MSE	Bias	Variance	MSE
100	Time	1.2625 (0.3147)	1.1140 (0.4398)	0.0799	0.0771	0.0776	0.4904	0.3221	0.3809
	Frequency	2.5579 (0.1399)	1.1140 (0.4398)	0.0806	0.0775	0.0780	0.4903	0.3214	0.3804
200	Time	1.2214 (0.2728)	1.0200 (0.1538)	0.0260	0.0188	0.0189	0.2889	0.1334	0.1540
	Frequency	2.6691 (0.1126)	1.0200 (0.1538)	0.0260	0.0188	0.0189	0.2889	0.1334	0.1540
500	Time	1.2549 (0.2333)	1.0140 (0.1176)	0.0166	0.0063	0.0063	0.0960	0.0471	0.0495
	Frequency	3.1467 (0.1365)	1.0140 (0.1176)	0.0166	0.0063	0.0063	0.0960	0.0471	0.0495
1000	Time	1.8200 (0.6775)	1.0040 (0.0632)	0.0063	0.0030	0.0030	0.0458	0.0258	0.0263
	Frequency	5.1397 (0.1897)	1.0040 (0.0632)	0.0063	0.0030	0.0030	0.0458	0.0258	0.0263

Table 3.6 is the simulation results for Model 1 using the methods as mentioned earlier, namely the frequency domain method (denoted by 'Frequency') and the time domain approach (denoted by 'Time'). The frequency domain method consumes more CPU time comparing to the time domain approach. Both methods select a lag order of 2 or above in few experiments when the sample size is 100, and the over-selection of lag order is alleviated as the sample size raises.

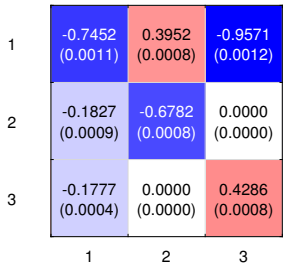
It is observed from the column $\hat{\mathbf{A}}$ of Table 3.6 that the frequency and time domain methods perform similarly, concerning the three metrics, in the estimation of AR coefficients. Figure 3.12 depicts the average AR estimates using the Frequency method and the Time method when $T = 100$ and $T = 1000$, together with the actual parameter value. From this figure, both methods obtain similar estimates at the same sample size, and the AR coefficient estimates at the positions (2,3) and (3,2) are, in particular, close to zero. The AR coefficient estimates deviate less from the true parameter value and possess less variability when the sample size increases.



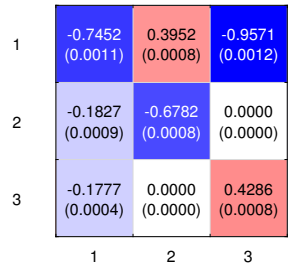
(a) Time ($T = 100$).



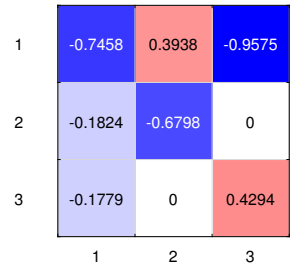
(b) Frequency ($T = 100$).



(c) Time ($T = 1000$).

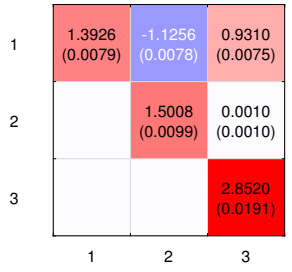


(d) Frequency ($T = 1000$).

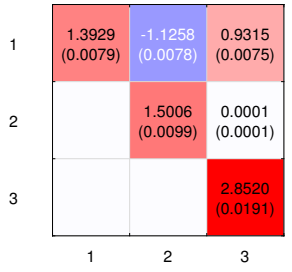


(e) True.

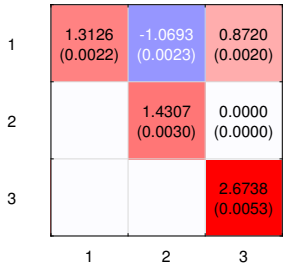
FIGURE 3.12: Average values of the AR coefficient estimates for Model 1, $\hat{\mathbf{A}}_1^{(1)}$. Standard errors are in parentheses.



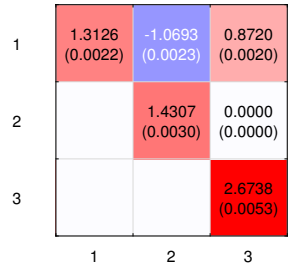
(a) Time ($T = 100$).



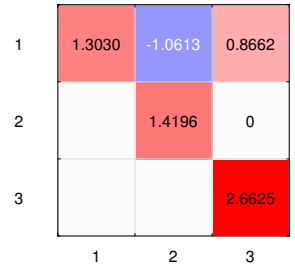
(b) Frequency ($T = 100$).



(c) Time ($T = 1000$).



(d) Frequency ($T = 1000$).



(e) True.

FIGURE 3.13: Average values of the inverse covariance estimates for Model 1, $\hat{\mathbf{\Sigma}}_1^{-1}$. Standard errors are in parentheses.

As shown in the column $\hat{\mathbf{\Sigma}}_u^{-1}$ of Table 3.6 that the inverse covariance estimates obtained by the frequency and time domain methods possess bias, variance, and MSE that are close in value. Figure 3.13 plots the average inverse covariance estimates using the two methods when $T = 100$ and $T = 1000$. We can observe from the figure that both methods perform

similarly in the estimation of inverse covariances at the same sample size, and the inverse covariance estimates at the positions (2,3) and (3,2) are in particular close to zero. The inverse covariance estimates deviate less from the true parameter value and are less disperse when the sample size raises.

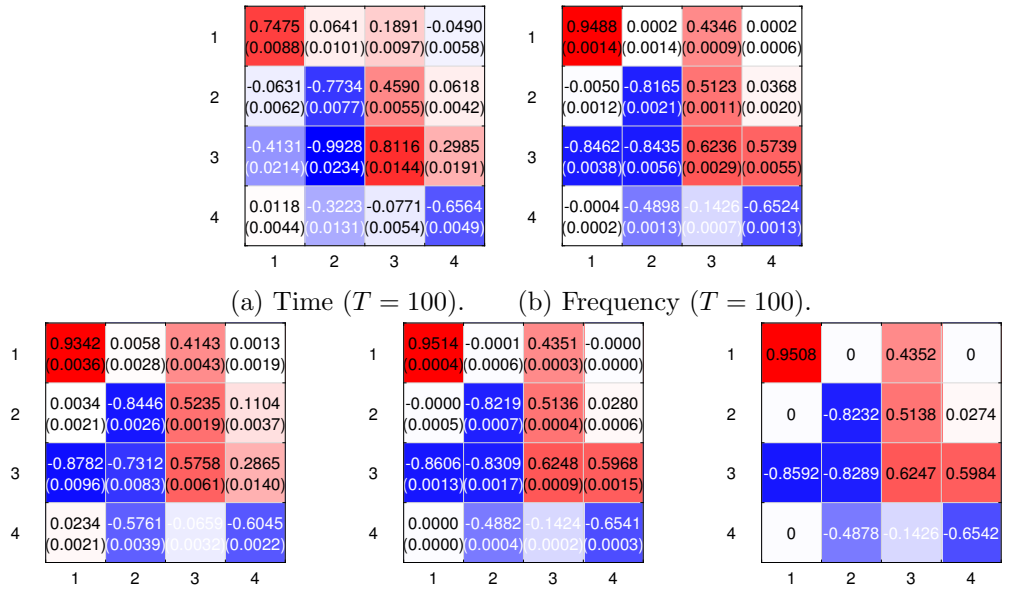
Model 2

TABLE 3.7: Simulation results for Model 2 over 500 replications. $\hat{p}$ is the average lag order determined, and Cputime is the average CPU time consumed in seconds. Standard deviations are in the parentheses.

				$\hat{\mathbf{A}}$			$\hat{\mathbf{\Sigma}}_u^{-1}$		
T	Method	Cputime	$\hat{p}$	Bias	Variance	MSE	Bias	Variance	MSE
100	Time	8.7147 (6.8711)	1.0040 (0.0632)	2.1144	1.1408	1.6423	3.8628	4.5129	8.7474
	Frequency	7.9339 (3.1356)	1.0040 (0.0632)	0.0880	0.0552	0.0564	0.9806	1.1169	1.3487
200	Time	8.8200 (6.4254)	1.0020 (0.0447)	1.7075	0.9154	1.2690	3.5558	3.7872	7.1700
	Frequency	8.2784 (1.9389)	1.0020 (0.0447)	0.0555	0.0335	0.0340	0.4283	0.4656	0.5087
500	Time	9.1705 (5.2187)	1.0000 (0.0000)	1.0707	0.5684	0.7512	2.3826	2.1777	4.1644
	Frequency	9.6923 (1.8836)	1.0000 (0.0000)	0.0247	0.0108	0.0109	0.1741	0.1779	0.1858
1000	Time	10.1260 (4.4215)	1.0000 (0.0000)	0.8777	0.2511	0.3856	1.5786	1.4830	2.8025
	Frequency	12.4939 (1.3267)	1.0000 (0.0000)	0.0088	0.0049	0.0049	0.0790	0.0895	0.0909

Table 3.7 documents the simulation results for Model 2 using the frequency domain method (denoted by 'Frequency') and the time domain approach (denoted by 'Time'). The frequency domain method, on average, consumes less CPU time when the sample size is 200 or below and consumes more when the sample size is 500 or above compared to the time domain approach. Both the Frequency method and the Time method select a lag order of 2 in few experiments when the sample size is 200 or below and choose the correct lag order in all experiments when the sample size is 500 or above.

We can observe from the column $\hat{\mathbf{A}}$ of Table 3.7 that the frequency domain method outperforms the time domain method, regarding the three metrics, in the estimation of AR coefficients for experiments of all sample



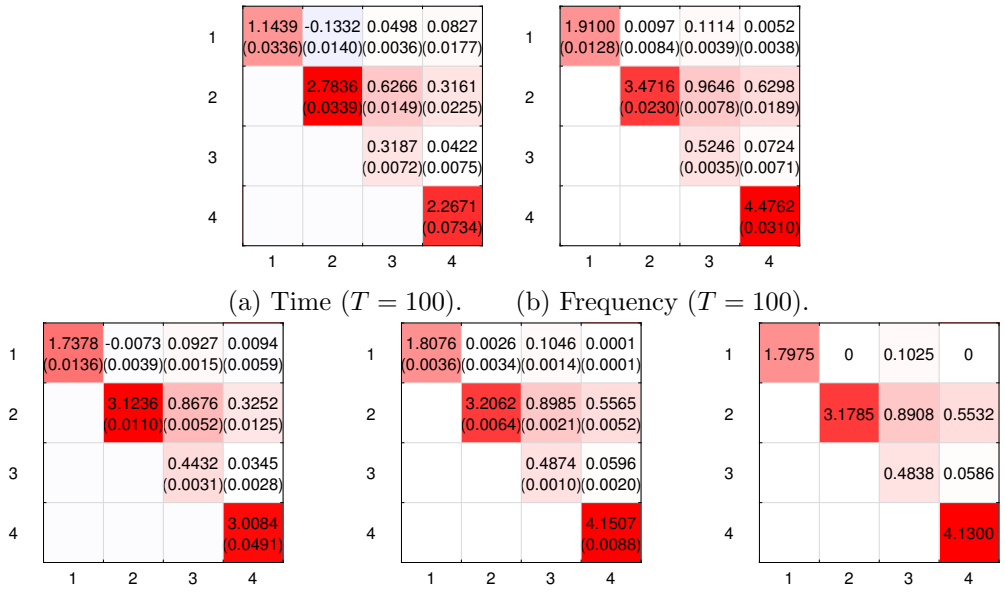
(a) Time ($T = 100$). (b) Frequency ($T = 100$).

(c) Time ($T = 1000$). (d) Frequency ($T = 1000$). (e) True.

FIGURE 3.14: Average values of the AR coefficient estimates for Model 2, $\hat{\mathbf{A}}_1^{(2)}$. Standard errors are in parentheses.

sizes. The performance of the two methods in the estimation of AR coefficients improves as the sample size raises. Figure 3.14 plots the average AR estimates using the Frequency method and the Time method when $T = 100$ and $T = 1000$. It is observed from this figure that the AR estimates obtained by the frequency domain method possess less bias from the actual parameter values comparing to the time domain method. The bias of estimates using the two methods reduces as the sample size increases to 1000. For the zero AR coefficients, both methods obtain estimates that are close to zero, and the frequency domain method performs better in the estimation.

The column $\hat{\mathbf{\Sigma}}_u^{-1}$ of Table 3.7 that the Frequency method performs better in the estimation of inverse covariances and the estimations improve as the sample size raises. Figure 3.15 depicts the average inverse covariance estimates using the two methods when $T = 100$ and $T = 1000$. We can observe from the figure that the inverse covariances estimated by the Time method deviate more from the true parameter value. As the sample size increases to 1000, the inverse covariance estimates deviate less from the



(a) Time ($T = 100$). (b) Frequency ($T = 100$).

(c) Time ($T = 1000$). (d) Frequency ($T = 1000$). (e) True.

FIGURE 3.15: Average values of the inverse covariance estimates for Model 2, $\hat{\mathbf{\Sigma}}_2^{-1}$. Standard errors are in parentheses.

actual parameter values for both methods.

Model 3

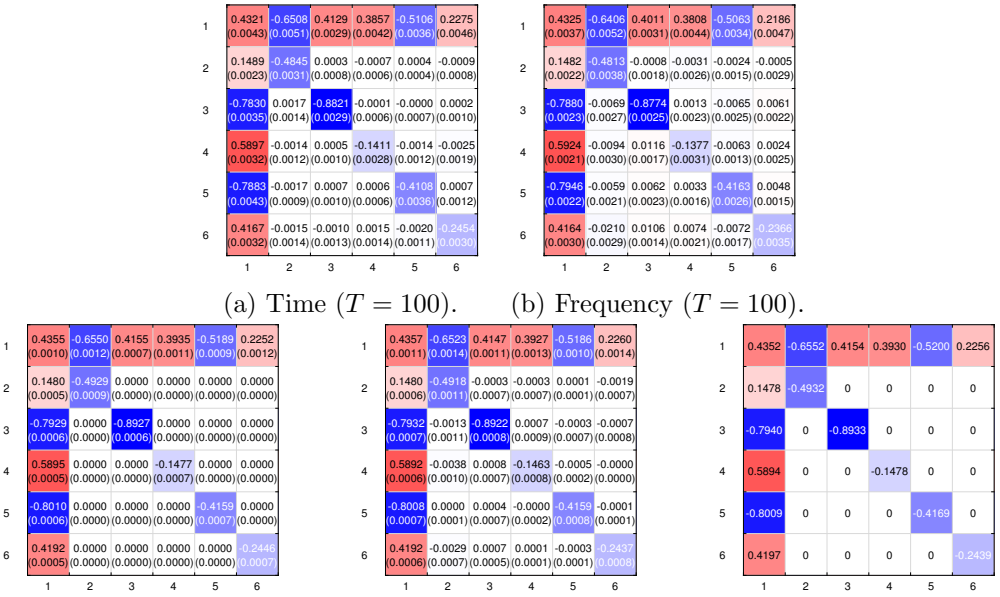
TABLE 3.8: Simulation results for Model 3 over 500 replications. $\hat{p}$ is the average lag order determined, and Cputime is the average CPU time consumed in seconds. Standard deviations are in the parentheses.

				$\hat{\mathbf{A}}$			$\hat{\mathbf{\Sigma}}_u^{-1}$		
T	Method	Cputime	$\hat{p}$	Bias	Variance	MSE	Bias	Variance	MSE
100	Time	6.4409 (3.6570)	1.0000 (0.0000)	0.1107	0.1172	0.1180	0.5739	0.2386	0.2715
	Frequency	23.2991 (9.3057)	1.0000 (0.0000)	0.2528	0.1388	0.1415	0.9130	0.2910	0.3648
200	Time	6.4688 (2.9980)	1.0000 (0.0000)	0.0428	0.0270	0.0272	0.2847	0.0771	0.0856
	Frequency	21.7309 (3.1045)	1.0000 (0.0000)	0.1263	0.0535	0.0542	0.4157	0.1082	0.1235
500	Time	6.2184 (2.6422)	1.0000 (0.0000)	0.0276	0.0104	0.0104	0.1098	0.0293	0.0307
	Frequency	21.2872 (1.9668)	1.0000 (0.0000)	0.0778	0.0228	0.0232	0.1657	0.0413	0.0438
1000	Time	7.9361 (2.8228)	1.0000 (0.0000)	0.0072	0.0053	0.0053	0.0516	0.0145	0.0148
	Frequency	23.9288 (2.0373)	1.0000 (0.0000)	0.0286	0.0111	0.0112	0.0738	0.0201	0.0207

Table 3.8 documents the simulation results for Model 3 using the frequency domain method (denoted by 'Frequency') and the time domain approach (denoted by 'Time'). The frequency domain method consumes more CPU

time comparing to the time domain approach. Both methods select the lag order correctly in all experiments.

It is observed from the column $\hat{\mathbf{A}}$ of Table 3.8 that the time domain method performs better, regarding the three metrics, in the estimation of AR coefficients, and both methods improve in the estimation as the sample size raises. Figure 3.16 plots the average AR estimates using the frequency and time domain methods when $T = 100$ and $T = 1000$. As shown in the figure, both methods obtain estimates that are close to the true parameter values, especially for the non-zero AR coefficients. For the zero AR coefficients, the time domain method performs better in the estimation.

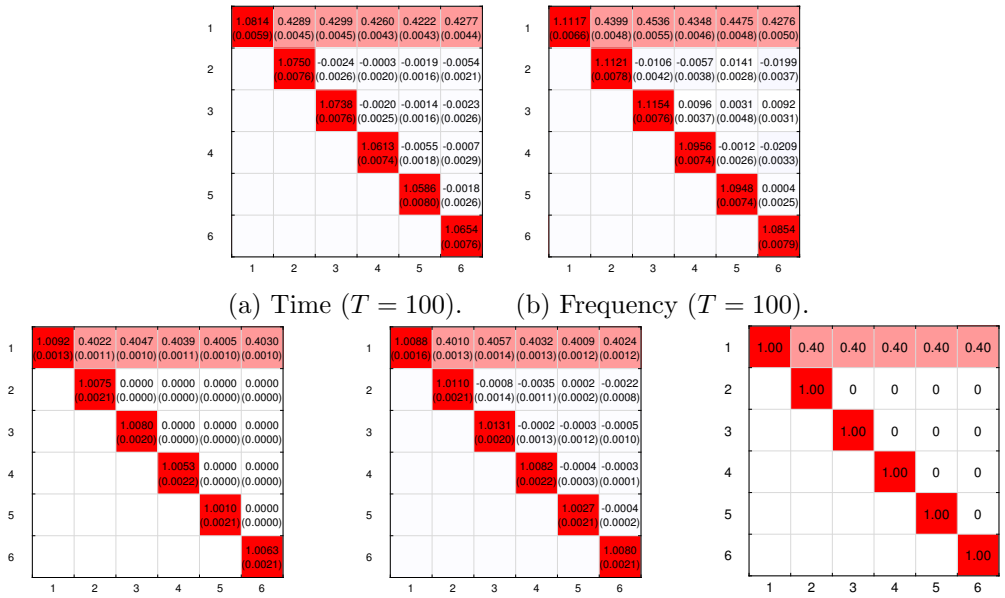


(a) Time ($T = 100$). (b) Frequency ($T = 100$).

(c) Time ($T = 1000$). (d) Frequency ($T = 1000$). (e) True.

FIGURE 3.16: Average values of the AR coefficient estimates for Model 3, $\hat{\mathbf{A}}_1^{(3)}$. Standard errors are in parentheses.

The column $\hat{\mathbf{\Sigma}}_u^{-1}$ of Table 3.8 shows that the time domain method obtains inverse covariance estimates that possess less bias, variance, and MSE. Both methods improve in the estimation performance when the sample size increases. Figure 3.17 shows the average inverse covariance estimates using the Frequency and the Time methods. The figure shows that both methods



(a) Time ($T = 100$). (b) Frequency ($T = 100$).

(c) Time ($T = 1000$). (d) Frequency ($T = 1000$). (e) True.

FIGURE 3.17: Average values of the inverse covariance estimates for Model 3, $\hat{\mathbf{\Sigma}}_3^{-1}$. Standard errors are in parentheses.

obtain estimates that are close to the actual parameter values and the time domain method performs better in the estimation of inverse covariances.

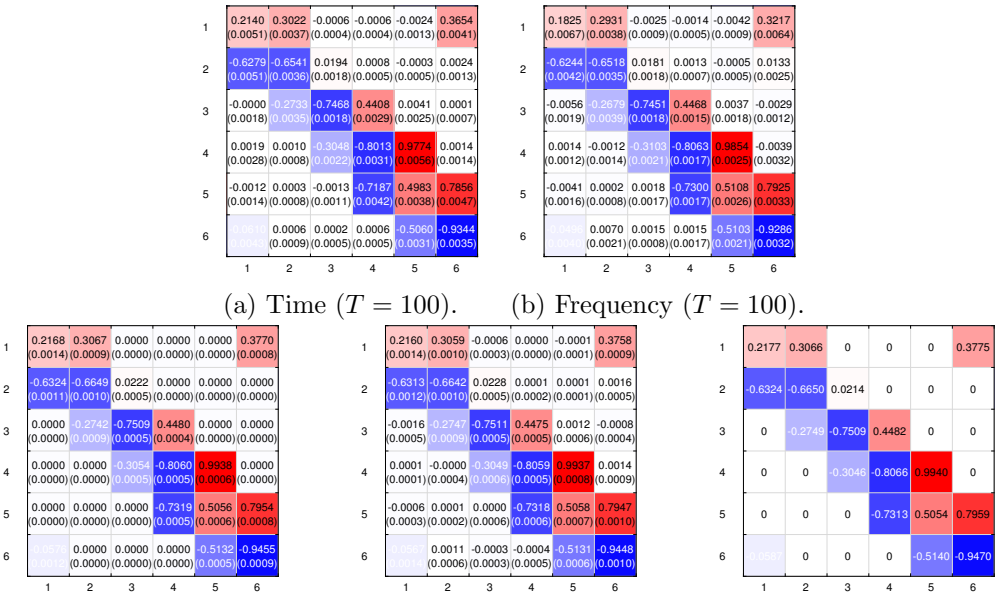
Model 4

TABLE 3.9: Simulation results for Model 4 over 500 replications. $\hat{p}$ is the average lag order determined, and Cputime is the average CPU time consumed in seconds. Standard deviations are in the parentheses.

				$\hat{\mathbf{A}}$			$\hat{\mathbf{\Sigma}}_u^{-1}$		
T	Method	Cputime	$\hat{p}$	Bias	Variance	MSE	Bias	Variance	MSE
100	Time	6.9903 (4.2977)	1.0000 (0.0000)	0.1455	0.1473	0.1485	0.5841	0.2848	0.3182
	Frequency	13.6009 (9.0822)	1.0000 (0.0000)	0.2570	0.1319	0.1377	0.7447	0.3324	0.3900
200	Time	5.6492 (2.4335)	1.0000 (0.0000)	0.0363	0.0335	0.0336	0.3102	0.0943	0.1039
	Frequency	18.5482 (9.0611)	1.0000 (0.0000)	0.1328	0.0591	0.0603	0.4007	0.1325	0.1477
500	Time	6.4786 (2.6200)	1.0000 (0.0000)	0.0196	0.0124	0.0124	0.1207	0.0337	0.0352
	Frequency	21.1352 (5.5363)	1.0000 (0.0000)	0.0475	0.0184	0.0185	0.1722	0.0414	0.0441
1000	Time	8.5184 (2.6142)	1.0000 (0.0000)	0.0095	0.0059	0.0059	0.0514	0.0161	0.0163
	Frequency	24.2196 (2.6749)	1.0000 (0.0000)	0.0272	0.0087	0.0088	0.0697	0.0193	0.0197

Table 3.9 documents the simulation results for Model 4 using the frequency domain method (denoted by 'Frequency') and the time domain approach

(denoted by 'Time'). The frequency domain method consumes more CPU time comparing to the time domain approach. Both the Frequency method and the Time method select the lag order correctly in all experiments.



(a) Time ($T = 100$). (b) Frequency ($T = 100$).

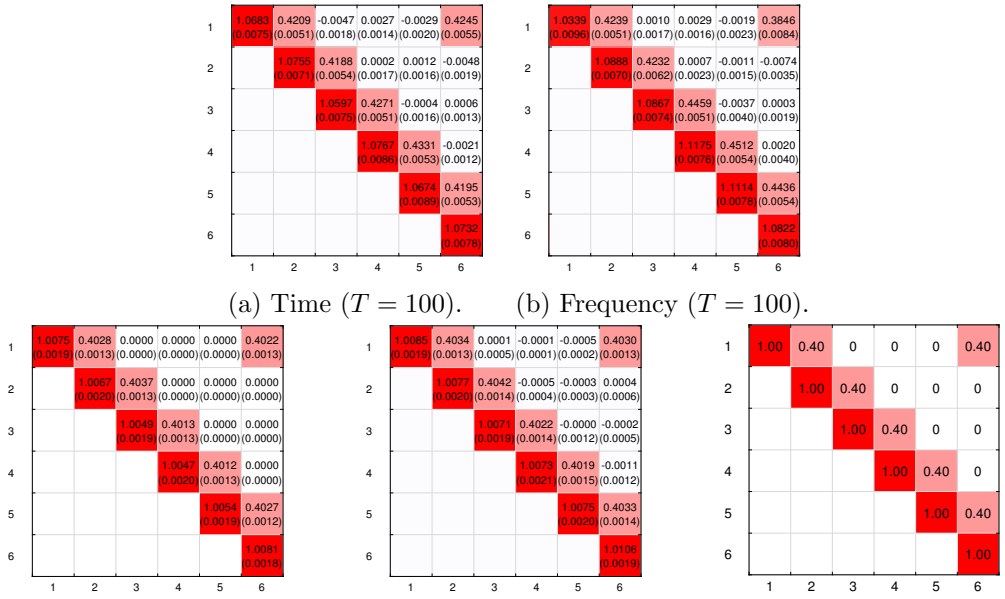
(c) Time ($T = 1000$). (d) Frequency ($T = 1000$). (e) True.

FIGURE 3.18: Average values of the AR coefficient estimates for Model 4, $\hat{\mathbf{A}}_1^{(4)}$. Standard errors are in parentheses.

The column $\hat{\mathbf{A}}$ of Table 3.9 shows that the time domain method performs better, concerning the three metrics, in the estimation of AR coefficients when the sample size is 200 or above. For the experiments with a sample size of 100, the time domain method obtains AR estimates that possess lower bias comparing to the frequency domain method, although the variance and MSE of the estimates are slightly higher. Both methods improve in the estimation performance for AR coefficients when the sample size increases. Figure 3.18 depicts the average AR estimates using the two methods when $T = 100$ and $T = 1000$. We can observe from the figure that both methods, in general, obtain estimates that are close to the true parameter values, and improvement in estimation is observed as the sample size increases to 1000.

As shown in the column $\hat{\mathbf{\Sigma}}_u^{-1}$ of Table 3.9 that the inverse covariance estimates obtained by the time domain method possess less bias, variance,

and MSE. Improvement in the estimation is observed as the sample size raises. Figure 3.19 displays the average inverse covariance estimates using the Frequency and Time methods when $T = 100$ and $T = 1000$. This figure shows that both methods perform similarly in the estimation and the time domain method is better in the estimation of inverse covariances.



(a) Time ($T = 100$). (b) Frequency ($T = 100$).

(c) Time ($T = 1000$). (d) Frequency ($T = 1000$). (e) True.

FIGURE 3.19: Average values of the inverse covariance estimates for Model 4, $\hat{\mathbf{\Sigma}}_4^{-1}$. Standard errors are in parentheses.

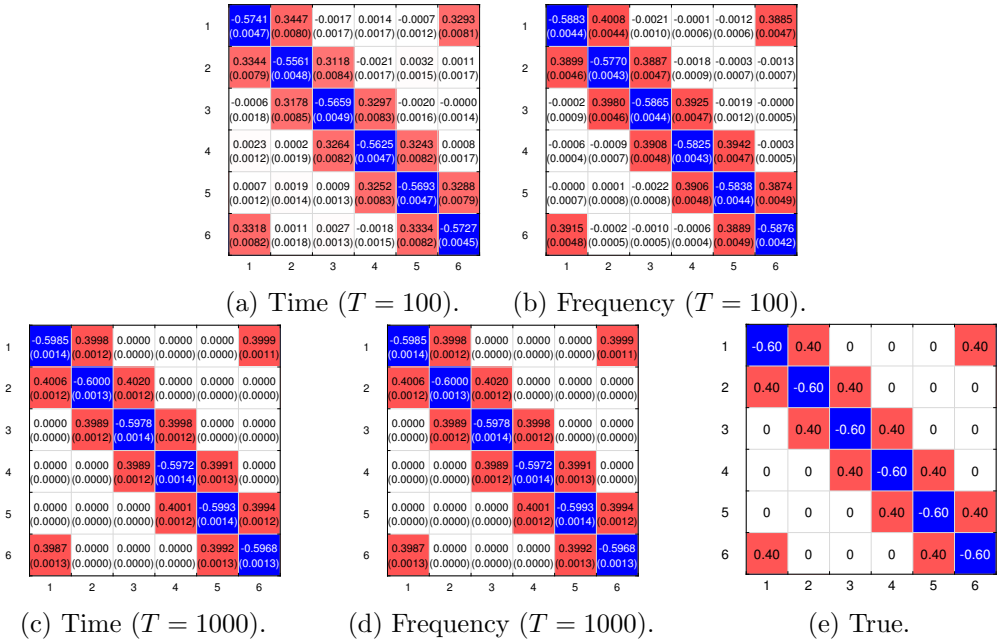
Model 5

TABLE 3.10: Simulation results for Model 5 over 500 replications. $\hat{p}$ is the average lag order determined, and Cputime is the average CPU time consumed in seconds. Standard deviations are in the parentheses.

				$\hat{\mathbf{A}}$			$\hat{\mathbf{\Sigma}}_u^{-1}$		
T	Method	Cputime	$\hat{p}$	Bias	Variance	MSE	Bias	Variance	MSE
100	Time	5.6652 (3.5030)	2.0000 (0.0000)	1.8869	0.7734	0.8776	0.3095	0.4717	0.4786
	Frequency	10.5889 (13.4758)	2.0000 (0.0000)	0.3906	0.3612	0.3658	0.8073	0.2689	0.3381
200	Time	4.4266 (2.0759)	2.0000 (0.0000)	0.2231	0.1816	0.1832	0.3374	0.1148	0.1276
	Frequency	11.6884 (8.3678)	2.0000 (0.0000)	0.1517	0.1471	0.1479	0.3808	0.0971	0.1129
500	Time	4.5902 (1.9192)	2.0000 (0.0000)	0.0640	0.0585	0.0587	0.1597	0.0362	0.0389
	Frequency	17.2995 (10.9602)	2.0000 (0.0000)	0.0640	0.0585	0.0587	0.1597	0.0362	0.0389
1000	Time	7.1894 (2.3372)	2.0000 (0.0000)	0.0390	0.0291	0.0291	0.0699	0.0174	0.0179
	Frequency	21.9524 (3.7277)	2.0000 (0.0000)	0.0390	0.0291	0.0291	0.0699	0.0174	0.0179

Table 3.10 reports the simulation results for Model 5 using the frequency domain method (denoted by 'Frequency') and the time domain approach (denoted by 'Time'). The frequency domain method consumes more CPU time comparing to the time domain approach. Both the methods choose the lag order correctly in all experiments.

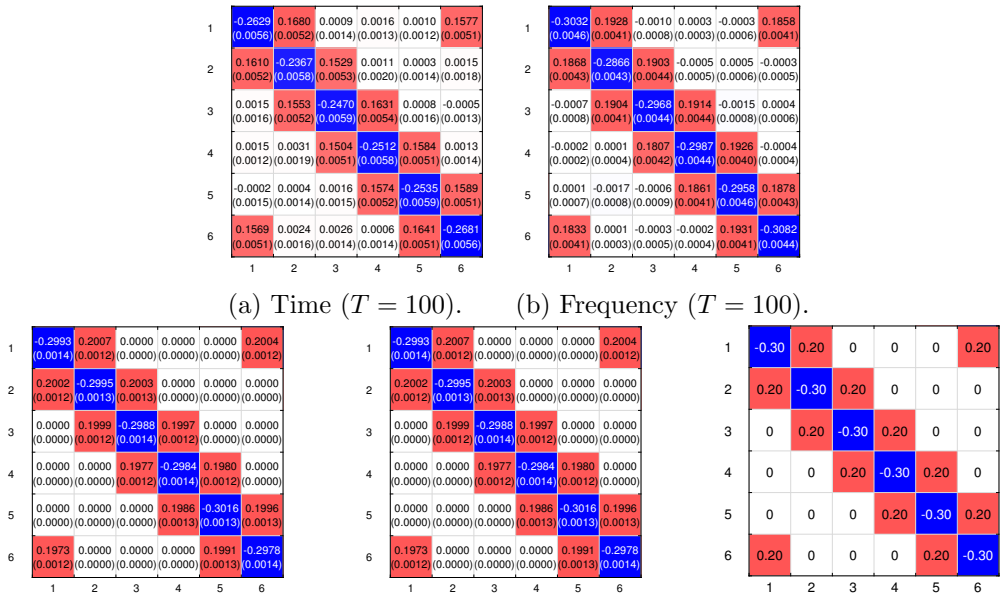
The column $\hat{\mathbf{A}}$ of Table 3.9 shows that the frequency domain method performs better, regarding the three metrics, in the estimation of AR coefficients when the sample size is 200 or below. Both methods behave similarly in the estimation of AR coefficients when the sample size is 500 or above. Figure 3.20 (3.21) depicts the average AR estimates of lag 1 (lag 2) using the two methods when $T = 100$ and $T = 1000$. As shown in the figures, the AR estimates obtained using the Frequency method deviate less from the actual value when the sample size is 100, especially for the non-zero AR coefficients, comparing to the Time method. Both methods obtain similar AR coefficient estimates when the sample size is larger ($T = 1000$).



(a) Time ($T = 100$). (b) Frequency ($T = 100$).

(c) Time ($T = 1000$). (d) Frequency ($T = 1000$). (e) True.

FIGURE 3.20: Average values of the AR coefficient of lag 1 estimates for Model 5, $\hat{\mathbf{A}}_1^{(5)}$. Standard errors are in parentheses.

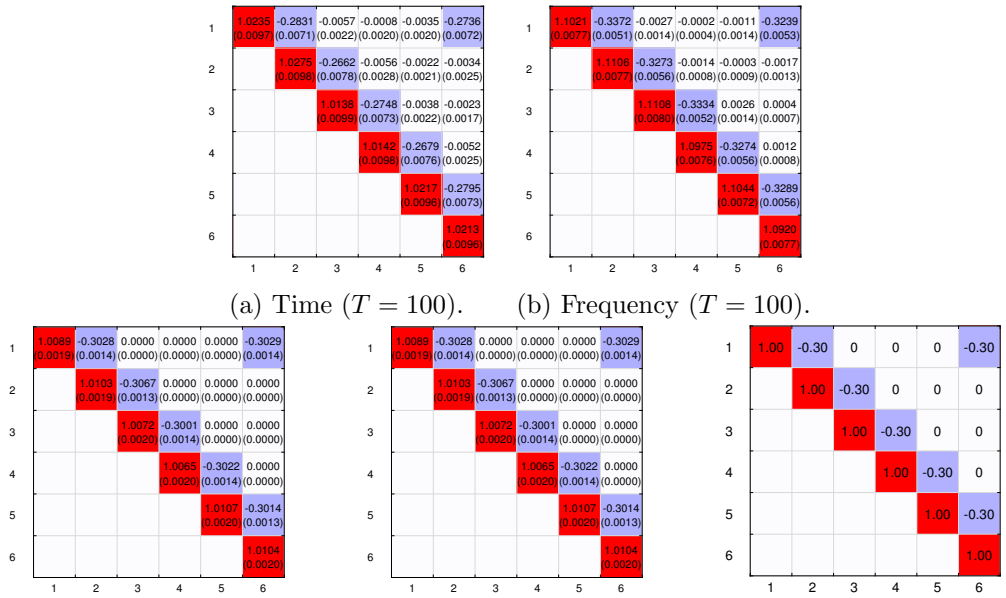


(a) Time ($T = 100$). (b) Frequency ($T = 100$).

(c) Time ($T = 1000$). (d) Frequency ($T = 1000$). (e) True.

FIGURE 3.21: Average values of the AR coefficient of lag 2 estimates for Model 5, $\hat{\mathbf{A}}_2^{(5)}$. Standard errors are in parentheses.

We can observe from the column $\hat{\boldsymbol{\Sigma}}_u^{-1}$ of Table 3.10 that the frequency domain method outperforms the time domain method, concerning the variance and MSE of the estimates, in the estimation of inverse covariances when the sample size of 200 or below, although the bias of the estimates is higher. The two methods perform similarly in the estimation when the sample size is 500 or above. Figure 3.22 shows the average inverse covariance estimates using the Frequency and Time methods when $T = 100$ and $T = 1000$. From this figure, the inverse covariance estimates obtained by the Frequency method deviate slightly higher form the actual values than that obtained by the Time method when $T = 100$, especially for the non-zero inverse covariances. Both methods obtain similar inverse covariance estimates as the sample size raises to 1000.



(a) Time ($T = 100$). (b) Frequency ($T = 100$).

(c) Time ($T = 1000$). (d) Frequency ($T = 1000$). (e) True.

FIGURE 3.22: Average values of the inverse covariance estimates for Model 5, $\hat{\mathbf{\Sigma}}_5^{-1}$. Standard errors are in parentheses.

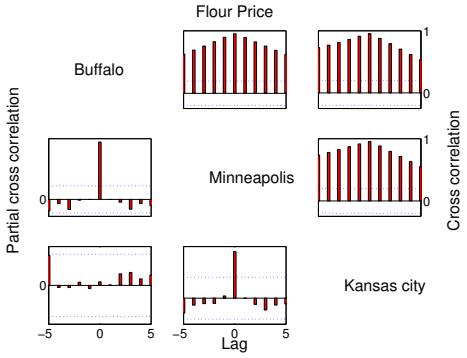
In summary, the frequency domain and the time domain methods perform similarly in the estimation of AR coefficients and inverse covariances, especially when the sample size is large. Their estimates have similar biases and sample variances. The time domain method consumes less CPU time in the estimation comparing to the frequency domain method. This is natural as the latter approach needs the evaluation of Fourier transforms.

3.2.2 Applications

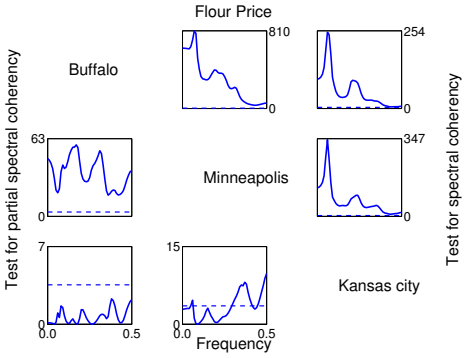
Flour price indices

We employ the introduced method to a monthly flour price indices data in Buffalo, Minneapolis, and Kansas City, over the period from August 1972 to November 1980, with a length of 100 months. This dataset has been studied by Tunnicliffe-Wilson et al. (2015) to investigate the dynamic interdependencies among the indices by fitting a parsimonious structural

vector autoregressive model. We utilize the time domain and frequency domain methods, described in Section 3.1.2, to identify partial correlation graphs of the three price series. With the determined partial correlation graph, we fit a sparse VAR model to the series using the alternating method to explore the dynamic interdependencies between the flour price indices. The 2-Stage approach (Davis et al., 2016) is also adopted as a comparison.



(a) Partial cross-correlations for the flour prices data. The blue dotted line represents an approximate 5% error bound of $\pm 2/\sqrt{T}$.



(b) Test statistics of spectral coherences (above diagonal) and partial spectral coherences (below diagonal) for the flour prices data.

FIGURE 3.23: Partial cross-correlations and Test statistics of spectral and partial spectral coherences for the flour prices data.

Figure 3.23(a) shows the cross-correlations (upper triangular part) and partial cross-correlations (lower triangular part) for the flour prices data. The blue dotted line represents an approximate 5% error bound of $\pm 2/\sqrt{T}$. This figure suggests the partial cross-correlation of the Buffalo and the Kansas City series is insignificant at all lags. Figure 3.23(b) depicts the test statistics of spectral (upper triangular part) and partial spectral (lower triangular part) coherencies for the flour prices data. The blue dotted line in each subplot is the corresponding critical value of the F distribution at 5% level of significance. This figure shows all coherences are significant, while the partial coherence of the Buffalo and the Kansas City series is insignificant at all frequencies. Based on this result, both methods identify the same partial correlation graph for the flour price indices (Figure 3.24).

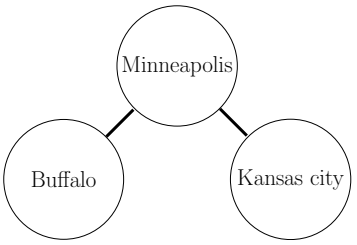
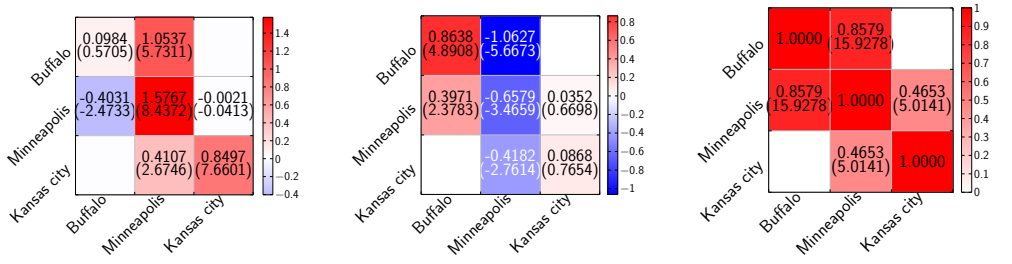


FIGURE 3.24: Partial correlations graph for the flour prices data.

With the identified partial correlation graph, we can determine the sparsity constraints, and estimate a VAR model using the alternating method. A model of lag order 2 is identified, and both time and frequency domain methods do not select more autoregressive coefficient pairs and inverse covariances to be zero (i.e. only the autoregressive coefficients and inverse innovation covariance of the case: Buffalo / Kansas city are restricted to zero). The heatmaps in Figure 3.25 visualize the estimated autoregressive coefficients and partial correlations of innovations. Figure 3.26(a) renders the determined VAR model by the mixed graph presented in Section 2.2.2. Figure 3.26(b) plots the directed acyclic graph representing the structural VAR model for the flour prices series suggested by Tunnicliffe-Wilson et al. (2015). In Figure 3.26(b), $X_{1,t}$, $X_{2,t}$ and $X_{3,t}$ represents the flour price indices at time t at Buffalo, Minneapolis, and Kansas city, respectively.



(a) AR coefficients of lag order 1.

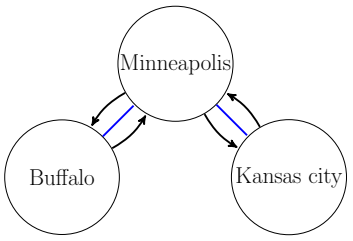
(b) AR coefficients of lag order 2.

(c) Partial correlations of innovations.

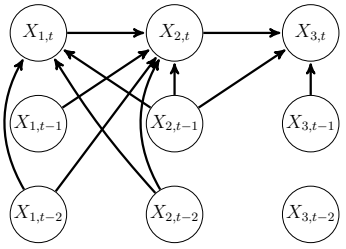
FIGURE 3.25: The autoregressive coefficient estimates and the estimated partial correlations of innovations using the time and frequency domain methods for the flour prices data (t-values are in parentheses).

As shown in Figure 3.25, some of the autoregressive coefficients are insignificant, and all partial correlations of innovations are significant. We

note that the estimate VAR model possesses similar dynamic inter-relation structure comparing to the structural VAR model identified by Tunnicliffe-Wilson et al. (2015). Both models suggest that the Buffalo and the Kansas city flour price indices are contemporaneously conditionally independent. We next consider the dependence between current and lagged variables. We can observe from the DAG in Figure 3.26(b) that there are no links in the directions $X_{1,t-1} \to X_{1,t}$, $X_{3,t-1} \to X_{2,t}$, $X_{3,t-2} \to X_{2,t}$, and $X_{3,t-2} \to X_{3,t}$. The autoregressive coefficients, determined by the introduced method, of the corresponding directions are also insignificant. For example, $X_{3,t-1} \to X_{2,t}$ corresponds to the autoregressive coefficient estimates with value $-0.0021(-0.0413)$ in the estimated VAR model (see Figure 3.25(a)).



(a) A mixed graph visualizing the estimated VAR model for the flour prices data.



(b) A DAG representing the structural VAR model identified by Tunnicliffe-Wilson et al. (2015) for the flour prices data.

FIGURE 3.26: Graphs for the flour prices data.

The 2-Stage method (Davis et al., 2016) obtains a sparse VAR model of order 6. This perhaps is because the method over-selects the autoregressive coefficients to be zero in the first stage of the 2-Stage approach. The method, in particular, constrains the autoregressive coefficient matrices to be diagonal for all lag order, leading to a less interpretable VAR model regarding the dynamic interdependencies structure.

Air pollution data in the Pearl River Delta region

We apply the proposed estimation method to an air pollution data in the Pearl River Delta region (PRDR)[1]. The government authorities have published the monthly time series data on few air pollutants in some air quality monitoring stations across the PRDR on a quarterly basis. The pollutants include sulphur dioxide (SO_2), nitrogen dioxide (NO_2), ozone (O_3) and respirable suspended particulates (RSP). Note that RSP is equivalent to particulate matter with a particle size less than 10 microns (PM_{10}). During the past decade, some of the monitoring stations in the region were under maintenance for an extended period, and some were closed and replaced by new ones. We thus select seven locations that have full data to study the interaction of RSP between the stations.

The data from January 2006 to December 2015, with a length of 120 months, are analyzed. Box-Cox transformations on each RSP series are first performed to stabilize the variance. Some of the transformed series possess decreasing trend, and all series possess seasonal pattern. Therefore, we detrend the transformed series that have a decreasing trend and then deseasonalize all the RSP series after treatments using harmonic regression described in McLeod & Gweon (2013). The time domain and frequency domain methods are applied to determine partial correlation graphs of the seven RSP series. We then determine sparse VAR models to investigate the inter-relationship of the RSP between monitoring stations further. We also implement the 2-Stage method (Davis et al., 2016) to the series as a comparison.

Figure 3.27 plots the partial cross-correlations for the air pollution data. The upper (lower) triangular part of the diagram shows the cross-correlations

[1] `http://www.epd.gov.hk/epd/english/resources_pub/publications/m_report.html`

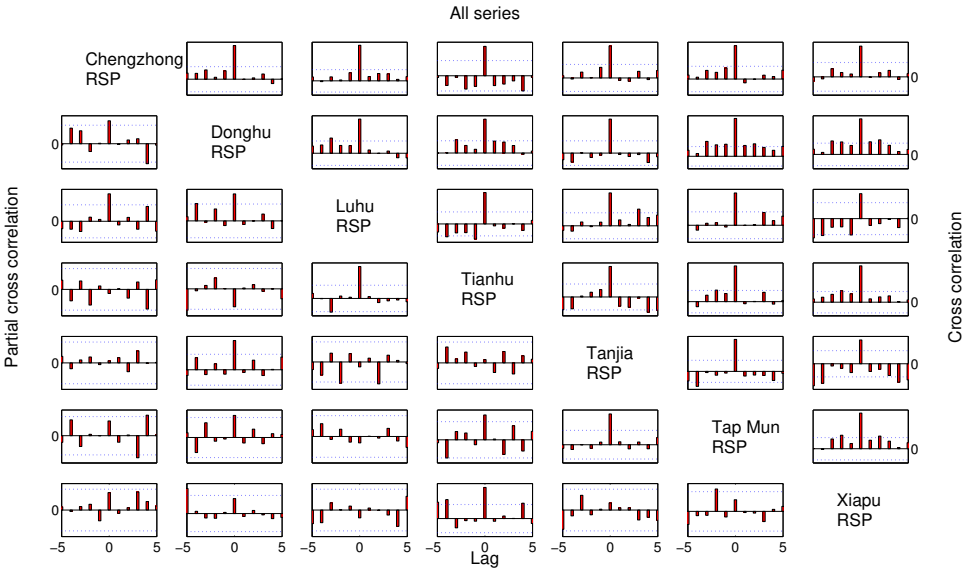


FIGURE 3.27: Partial cross-correlations for the PRDR air pollution data. The blue dotted line represents an approximate 5% error bound of $\pm 2/\sqrt{T}$.

(partial cross-correlations) described in Section 2.2.1. The blue dotted line in each subplot is the approximate 5% error bound of $\pm 2/\sqrt{T}$. Based on the significance of the partial cross-correlations, the partial correlation graph is identified and is shown in Figure 3.29(a). The nodes in the partial correlation graph are connected if the corresponding partial cross-correlation is significant. A blue dotted (bold red) line in the graph represents the corresponding partial cross-correlation is significant at non-zero (zero) lags.

Figure 3.28 depicts the test statistics of spectral and partial spectral coherencies. The upper (lower) triangular part of the plot shows the test statistics of spectral (partial spectral) coherencies described in Section 2.2.1. The blue dotted line in each subplot is the corresponding critical value of the F distribution at 5% significance level. We then determine a partial correlation graph based on the significance of the partial spectral coherencies and the graph is displayed in Figure 3.29(b). The nodes in the identified partial correlation graph are linked when the corresponding partial spectral coherence is significant. A blue dotted (purple dashed, bold red) line in the

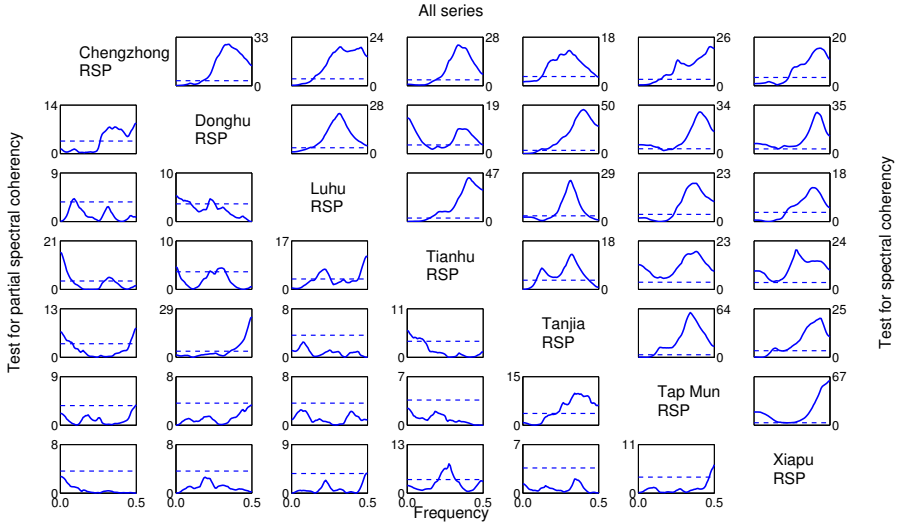
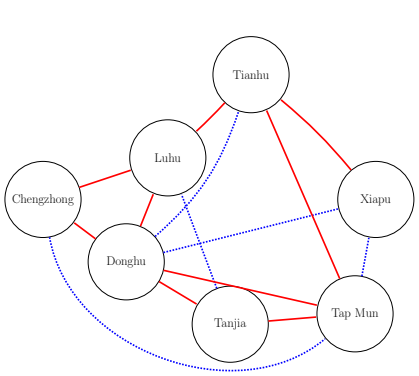
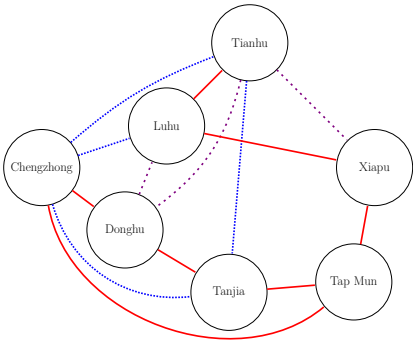


FIGURE 3.28: Test statistics of spectral coherencies (above diagonal, the blue dotted line represents a 95% quantile of the $F(2, 20)$ distribution) and partial spectral coherencies (below diagonal, the blue dotted line represents a 95% quantile of the $F(2, 12)$ distribution) for the PRDR air pollution data.



(a) Time domain method. The blue dotted (bold red) line indicates that the corresponding partial cross-correlation is significant at non-zero (zero) lags.



(b) Frequency domain method. The blue dotted (purple dashed, bold red) line indicates that the corresponding partial coherence is significant at low (mid, high) frequencies.

FIGURE 3.29: Partial correlation graph for the PRDR air pollution data. The figure displays the approximate geographical location and is not drawn to scale.

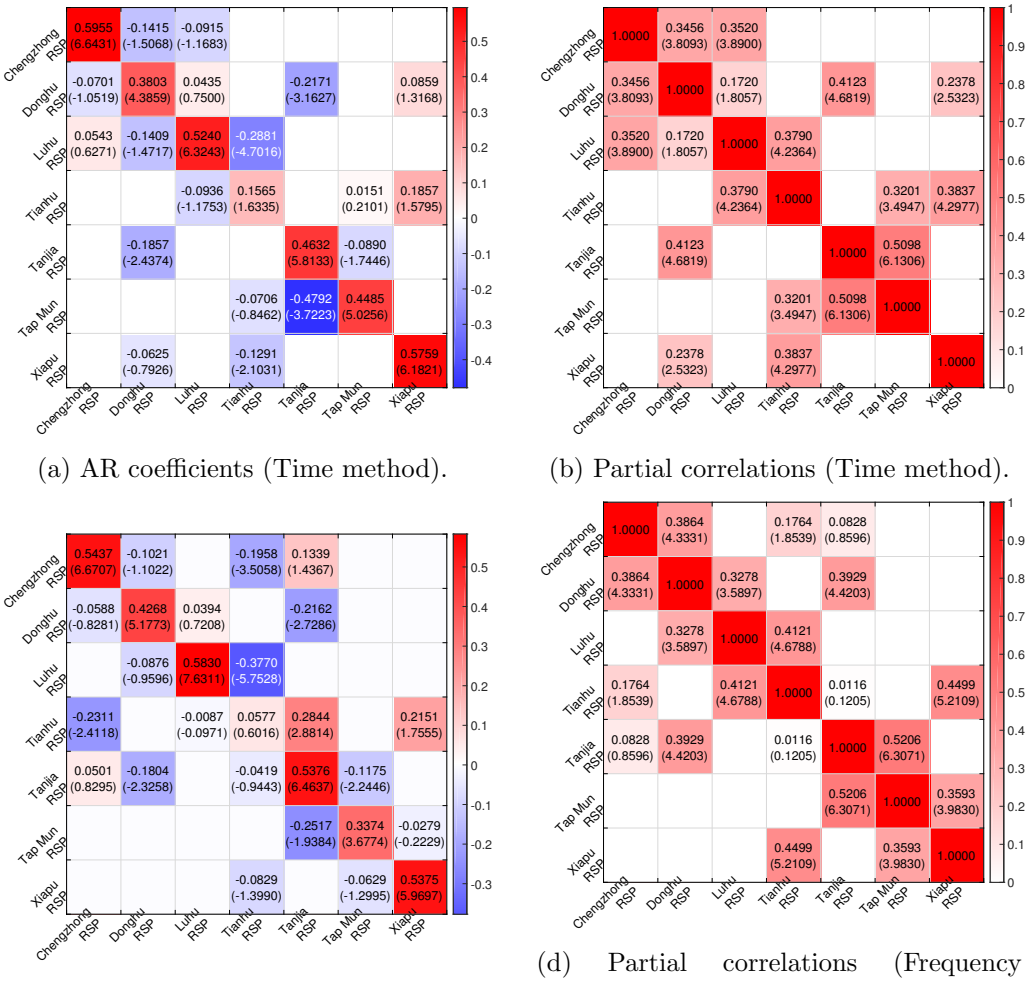
graph represents the corresponding partial coherency is significant at low (mid, high) frequencies.

As shown in Figure 3.29, the time domain and frequency domain methods identify similar, though not identical partial correlation graphs. The two methods agree on 10 out of 14 edges. The nodes between the cases:

Luhu / Tianhu, Chengzhong / Donghu, Donghu / Tanjia and Tanjia / Tap Mun are, in particular, connected by red edges. These cases indicate the corresponding partial cross-correlation is significant at zero lags or the corresponding partial spectral coherence is significant at high frequencies. Such observations seem to echo with the flight distances between the monitoring stations. For instance, the flight distances between the stations of the cases Luhu / Tianhu, Donghu / Tianhu and Tanjia / Tap Mun are 67 km, 61 km, and 73 km, respectively. Some of the partial cross-correlations and the partial spectral coherences are marginally significant at few lags or few frequencies, such as the case Donghu / Tianhu. Both the time and frequency domain methods in the estimation of a sparse VAR model on the RSP series identifies and restricts the corresponding autoregressive coefficients and inverse covariances to be zero.

Both the time and frequency domain methods determine a model of order 1. The heatmaps in Figure 3.30 visualize the autoregressive coefficient and partial correlation of innovations estimates. Figure 3.31 depicts the estimated VAR models by the mixed graph presented in Section 2.2.2. Each node represents the RSP series at a specific location among the seven monitoring stations. The estimated autoregressive coefficients and the partial correlation coefficients of the noise terms are printed next to the edges. The value in parenthesis is the t-value of the estimate of autoregressive coefficient or partial correlation. Some of the partial correlation coefficients are relatively small in magnitude, such as the partial correlation coefficient of the case Tianhu / Tanjia using the frequency domain method.

Following Tunnicliffe-Wilson et al. (2015), we fit a structural VAR (SVAR) model for the PRDR series. Figure 3.32 plots the directed acyclic graph (DAG) representing the determined SVAR model for the PRDR series. In Figure 3.32, $X_{1,t}$ represents the RSP series at time t at Chengzhong and



(a) AR coefficients (Time method).

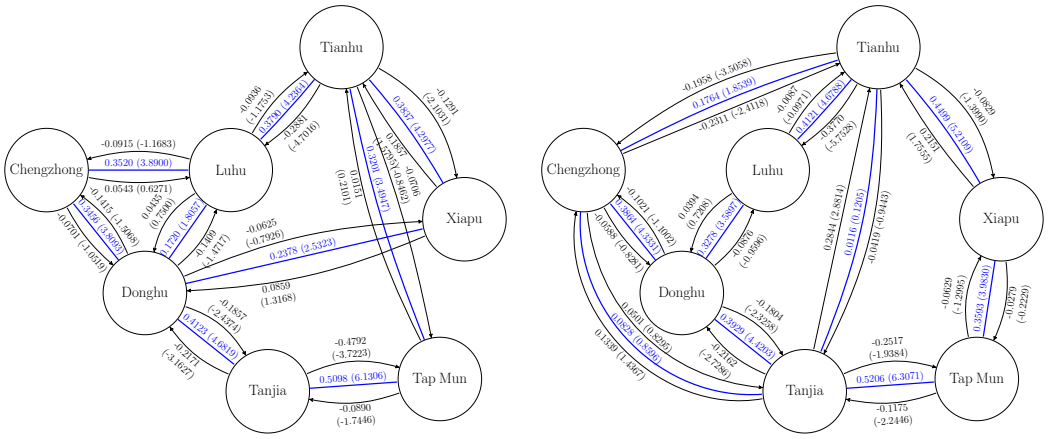
(b) Partial correlations (Time method).

(c) AR coefficients (Frequency method).

(d) Partial correlations (Frequency method).

FIGURE 3.30: The autoregressive coefficient estimates and the estimated partial correlations of innovations for the PRDR air pollution data (t-values are in parentheses).

others correspond to the labels at the bottom of the graph. We note that the estimated VAR model by the time domain method possesses similar dynamic inter-relation structure comparing to the SVAR model, especially for the contemporaneous dependence part. For the nodes that are connected by undirected edges in the mixed graph identified by the time domain method (Figure 3.31(a)), there are directed edges connecting the corresponding nodes of the current variables in the DAG (Figure 3.32). These cases are Chengzhong ($X_{1,t}$) / Donghu ($X_{2,t}$), Luhu ($X_{3,t}$) / Donghu ($X_{2,t}$) and Tap Mun ($X_{6,t}$) / Tanjia ($X_{5,t}$).



(a) Time domain method. (b) Frequency domain method.

FIGURE 3.31: A mixed graph visualizing the estimated VAR model for the PRDR air pollution data (the bold blue line represents the undirected edge determined by the inverse of noise covariance matrix, the black arrow is the directed edge characterized by the AR coefficient, and t-values in parentheses). The figure displays the approximate geographical location and is not drawn to scale.

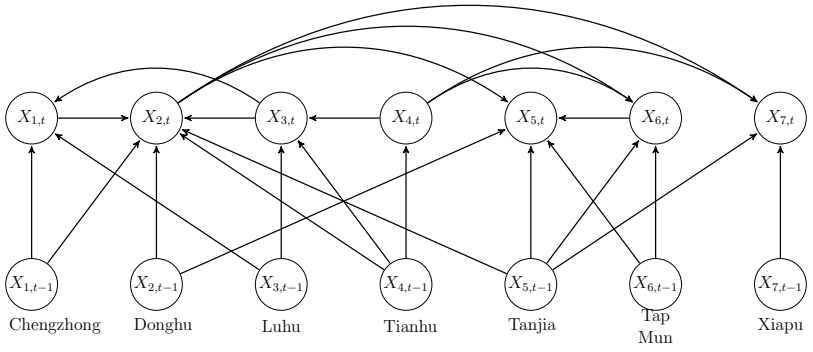
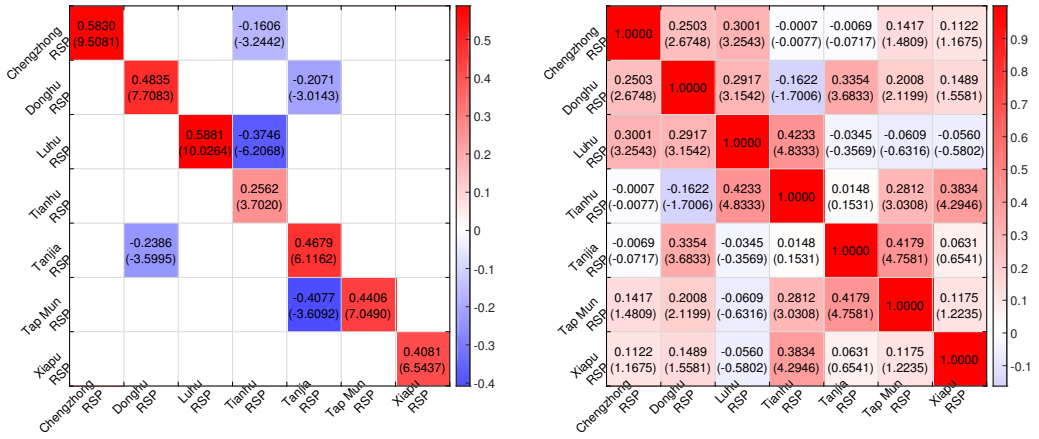


FIGURE 3.32: The DAG representing a SVAR for the PRDR series.

Figure 3.33 plots the AR coefficient and the partial correlation of noise estimates obtained by the 2-Stage method (Davis et al., 2016). Comparing to the AR coefficient estimates obtained by the alternating method, the 2-Stage approach achieves higher sparsity in the estimation of the AR coefficients. For the insignificant AR coefficient estimates determined by the alternating method, the 2-Stage approach shrinks most of these coefficients to zero. For the partial correlations, the 2-Stage method does not impose sparsity constraints on the inverse covariance matrix. Therefore, the partial correlations estimated by the 2-Stage approach are all non-zero.



(a) AR coefficients.

(b) Partial correlations of innovations.

FIGURE 3.33: The autoregressive coefficient estimates and the estimated partial correlations of innovations using the 2-Stage method for the PRDR air pollution data (t-values are in parentheses).

We note that there are some partial correlations, obtained by the 2-Stage method, are insignificant using a 5% level of significance; or critical value of $t_{0.025;\nu=107} = -1.9824$. These cases include Tanjia / Luhu, Tanjia / Tianhu, Tap Mun / Luhu, Xiapu / Luhu, and Xiapu / Tanjia. The partial correlations of innovations for these cases obtained by the time and frequency domain methods are zero or insignificant.

3.3 Summary

The present chapter introduced a constrained likelihood estimation method on sparse vector autoregressive (VAR) models, in which the autoregressive (AR) coefficients and the inverse covariance matrix are both restricted to be sparse. We have formulated the model estimation problem as a "biconcave" problem. This optimization problem is, in particular, concave when either the AR coefficients or the inverse noise covariance matrix is fixed. We have proposed an alternating maximization algorithm to solve the "biconcave" problem. The alternating method first estimates the AR coefficients with

fixed inverse covariance matrix followed by the estimation of the inverse covariance matrix with fixed AR coefficients alternately. We also studied the estimation performance of the introduced method and illustrated the method by real data examples.

In practice, we need to first determine the sparsity constraints before applying the estimation method. We adopted the same sparsity structure, determined by the conditional correlation graph, on both the AR coefficients and the inverse covariance in our implementation. The next chapter will present a penalized likelihood estimation method on sparse VAR models. The shrinkages on the AR coefficients and the inverse covariance are promoted by penalty terms. This means that the sparseness of the model is achieved in the estimation procedure without explicitly identifying the sparsity constraints.

Chapter 4

Penalized Likelihood Estimation Method

A fully parametrized vector autoregressive (VAR) model consists of a number of parameters which grows quadratically with the model dimension. Researchers have explored various methods to solve the over-parametrization problem. The methods include restricting the autoregressive (AR) coefficients to be zero based on the conditional correlation structure and the penalized likelihood estimation method by introducing penalty terms to the parameters to achieve model sparsity. The penalized likelihood method does not require the identification of the underlying conditional correlation structure from the sample before the estimation procedure. We, therefore, apply the penalized likelihood method for sparse VAR model estimation.

The current chapter begins with a brief introduction to penalized likelihood estimation. The problem formulation for the penalized likelihood estimation on VAR models is then presented. The AR coefficients and the off-diagonal elements of the inverse covariance matrix are particularly

penalized. The conditional maximum likelihood estimation on VAR models is equivalent to solve the multivariate regression problem. Researchers have studied the penalized likelihood estimation for multivariate regression (Rothman et al., 2008; Lee & Liu, 2012; Sofer et al., 2014). These work, in particular, solve the penalized likelihood estimation through alternating maximization. We present a different estimation method for the formulated problem, which is based on the local linear approximation (LLA) proposed by Zou & Li (2008). We carry out simulation experiments to investigate the finite sample properties of the penalized likelihood estimator. This chapter ends by applying the method to a real dataset for exemplification.

4.1 Problem Description

We adopt the penalized likelihood estimation on VAR models. The AR coefficients and the off-diagonal elements of the inverse noise covariance matrix are penalized for reducing model complexity. We first provide a brief introduction to the penalized likelihood method as a prologue to the discussion of the underlying problem. Suppose the penalized likelihood function is

$$l(\boldsymbol{\beta}) - n \sum_{j=1}^{p} p_\lambda(|\beta_j|), \tag{4.1}$$

where $l(\boldsymbol{\beta})$ is a likelihood function with a p-dimensional parameter $\boldsymbol{\beta}$, n is the sample size, and $p_\lambda(\cdot)$ is a penalty function. L_1 penalty (Tibshirani, 1996), smoothly clipped absolute deviation (SCAD) penalty (Fan & Li, 2001), and minimax concave penalty (MCP) (Zhang, 2010) are some of the penalty functions. The L_1 penalty is defined as $p_\lambda(|x|) = \lambda|x|$, where $\lambda > 0$.

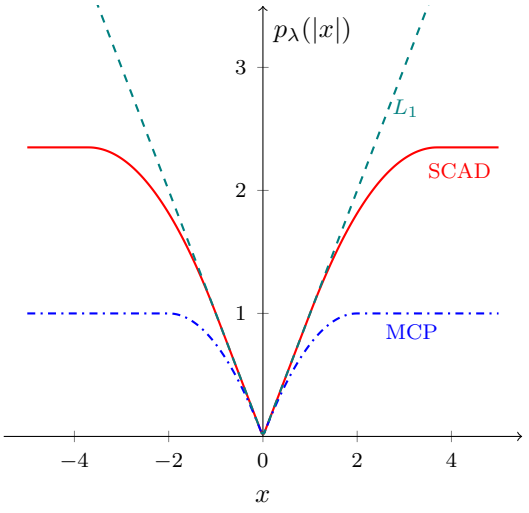


FIGURE 4.1: Some commonly used penalty functions.

The SCAD penalty is defined as

$$p_\lambda(|x|) = \lambda \int_0^{|x|} I\left(t \leq \lambda\right) + \frac{\max\left(0, a\lambda - t\right)}{(a-1)\lambda} I\left(t > \lambda\right)\, dt, \tag{4.2}$$

where $\lambda > 0$, $a > 2$, and $I(\cdot)$ is the indicator function. The MCP penalty is defined as

$$p_\lambda(|x|) = \lambda \int_0^{|x|} \max\left(0, 1 - \frac{t}{\gamma\lambda}\right) dt, \tag{4.3}$$

where $\lambda > 0$ and $\gamma > 0$. Figure 4.1 depicts these three penalty functions. The L_1 penalty function is a convex function while SCAD and MCP are non-convex functions. These penalty functions also have a singularity at the origin to ensure the penalized likelihood estimator possesses sparsity property. That is, the estimator shrinks the small estimates to zero to reduce model complexity. The penalized likelihood estimator with the L_1 penalty, however, is biased as discussed in Fan & Li (2001). Researchers, therefore, have adopted the non-convex penalties, such as SCAD and MCP, to alleviate the statistical bias issue.

The singularity and non-convexity encumber the maximization of the penalized likelihood function. Fan & Li (2001) proposed the local quadratic

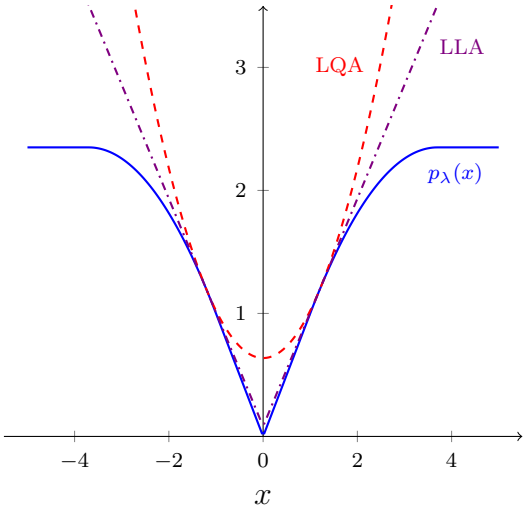


FIGURE 4.2: The local quadratic approximation (LQA) and local linear approximation (LLA) of a SCAD penalty, $p_\lambda(x)$, with $a = 3.7$, $\lambda = 1$ and $x^{(0)} = 1.2$.

approximation (LQA) for the penalty function by

$$p_\lambda(|x|) \approx p_\lambda(|x^{(0)}|) + \frac{p'_\lambda(|x^{(0)}|)\left(x^2 - (x^{(0)})^2\right)}{2|x^{(0)}|}, \tag{4.4}$$

for $x \approx x^{(0)}$. Figure 4.2 illustrates the LQA for a SCAD penalty function with $a = 3.7$, $\lambda = 1$ and $x^{(0)} = 1.2$. Having this local quadratic approximation, we can adopt the Newton-Raphson algorithm to maximize the penalized likelihood function. Specifically, we can solve the problem (4.1) by the Newton-Raphson algorithm iteratively and update the LQA for the penalty function at each iteration,

$$\boldsymbol{\beta}^{(k+1)} = \arg\max_{\boldsymbol{\beta}} \left\{ l(\boldsymbol{\beta}) - n \sum_{j=1}^{p} \frac{p'_\lambda(|\beta_j^{(k)}|)\beta_j^2}{2|\beta_j^{(k)}|} \right\}. \tag{4.5}$$

The quadratic function in (4.4), however, is not defined when $x^{(0)} = 0$. Fan & Li (2001) suggested to set the estimates to zero and exclude such estimates from the estimation once the estimates are close to zero.

Hunter & Li (2005) introduced a perturbed version of the LQA for the penalty function by,

$$\Phi_{x^{(0)},\lambda,\varepsilon}(x) = p_{\lambda,\varepsilon}(|x^{(0)}|) + \frac{p'_\lambda(|x^{(0)}|)\left(x^2 - (x^{(0)})^2\right)}{2(|x^{(0)}| + \varepsilon)} \quad \text{for } \varepsilon > 0,$$

where $p_{\lambda,\varepsilon}(|x|) = p_\lambda(|x|) - \varepsilon \int_0^{|x|} \frac{p'_\lambda(t)}{\varepsilon+t}\, dt$. The perturbed LQA method attenuates the issue of the LQA when the estimate is close to zero. Thus, the penalty part of the algorithm in (4.5) can be replaced and becomes

$$\boldsymbol{\beta}^{(k+1)} = \arg\max_{\boldsymbol{\beta}} \left\{ l(\boldsymbol{\beta}) - n \sum_{j=1}^{p} \frac{p'_\lambda(|\beta_j^{(k)}|)\left(\beta_j^2 - (\beta_j^{(k)})^2\right)}{2(|\beta_j^{(k)}| + \varepsilon)} \right\}. \tag{4.6}$$

Hunter & Li (2005) also showed the two algorithms (4.5) and (4.6) belong to the minorization-maximization (MM) algorithm. The local quadratic approximation function, in particular, acts as a surrogate function to minorize the objective function being maximized. We refer the reader to Lange (2013) for more details.

Zou & Li (2008) proposed the local linear approximation (LLA) to the penalty function by

$$p_\lambda(|x|) \approx p_\lambda(|x^{(0)}|) + p'_\lambda(|x^{(0)}|)(|x| - |x^{(0)}|),$$

for $x \approx x^{(0)}$. The LLA approach circumvents the difficulty in choosing an appropriate perturbation ε in the perturbed LQA method. Therefore, the algorithm is refined to be

$$\boldsymbol{\beta}^{(k+1)} = \arg\max_{\boldsymbol{\beta}} \left\{ l(\boldsymbol{\beta}) - n \sum_{j=1}^{p} p'_\lambda(|\beta_j^{(k)}|)|\beta_j| \right\}.$$

4.1.1 Problem Formulation

Recall from the problem (3.2) in Section 3.1.1 that the log-likelihood function of interest is

$$l(\mathbf{B}, \mathbf{\Theta}) = -\frac{KT}{2} \log 2\pi + \frac{T}{2} \log \det \mathbf{\Theta} - \frac{1}{2} \operatorname{trace} \left[(\mathbf{Y} - \mathbf{BZ})^\top \mathbf{\Theta} (\mathbf{Y} - \mathbf{BZ}) \right]. \tag{4.7}$$

Using the notation in Chapter 2, we consider the following penalized likelihood estimation problem,

$$\underset{\mathbf{B}, \mathbf{\Theta}}{\text{maximize}}\, Q(\mathbf{B}, \mathbf{\Theta}) = l(\mathbf{B}, \mathbf{\Theta}) - T \sum_{i,j} p_{\lambda_{b,ij}}(|b_{ij}|) - T \sum_{i \neq j} p_{\lambda_{\theta,ij}}(|\theta_{ij}|), \tag{4.8}$$

where $p_{\lambda_{b,ij}}(\cdot)$ and $p_{\lambda_{\theta,ij}}(\cdot)$ are some given penalty functions, and $\lambda_{b,ij}$, and $\lambda_{\theta,ij}$ are the tuning parameters. Note that we do not penalize the intercept terms.

Rothman et al. (2010) considered the penalized likelihood method for the multivariate regression problem, with the L_1 penalty, and called the method multivariate regression with covariance estimation (MRCE). Lee & Liu (2012) generalized the MRCE method by replacing the L_1 penalty with the weighted L_1 penalty. The authors proved the asymptotic properties of the proposed estimator. Sofer et al. (2014) introduced a two-stage procedure for the penalized likelihood estimation for multivariate regression. These work attempted to solve the estimation through alternating convex search (ACS) method as discussed in Chapter 3. The ACS method maximizes the penalized likelihood function for either fixed regression coefficients or inverse covariance alternately. We, instead, estimates both the regression coefficients and inverse covariance simultaneously.

4.1.2 Estimation Method

Denote $\boldsymbol{\beta} = \mathbf{vec}(\mathbf{B})$, $\boldsymbol{\theta} = \mathbf{vech}(\boldsymbol{\Theta})$, and $\boldsymbol{\omega} = (\boldsymbol{\beta}^\top, \boldsymbol{\theta}^\top)^\top$. With an initial estimate $\boldsymbol{\omega}^{(0)}$, we approximate the log-likelihood function in (4.7) by

$$\tilde{l}(\boldsymbol{\omega}|\boldsymbol{\omega}^{(0)}) = l(\boldsymbol{\omega}^{(0)}) + \nabla l\left(\boldsymbol{\omega}^{(0)}\right)^\top \left(\boldsymbol{\omega} - \boldsymbol{\omega}^{(0)}\right) + \frac{1}{2}\left(\boldsymbol{\omega} - \boldsymbol{\omega}^{(0)}\right)^\top \nabla^2 l(\boldsymbol{\omega}^{(0)}) \left(\boldsymbol{\omega} - \boldsymbol{\omega}^{(0)}\right),$$

where

$$\nabla l\left(\boldsymbol{\omega}\right) = \begin{pmatrix} \mathbf{vec}\left[\boldsymbol{\Theta}(\mathbf{Y} - \mathbf{BZ})\mathbf{Z}^\top\right] \\ \frac{1}{2}\mathbf{D}_K^\top \mathbf{vec}\left[T\boldsymbol{\Theta}^{-1} - (\mathbf{Y} - \mathbf{BZ})(\mathbf{Y} - \mathbf{BZ})^\top\right] \end{pmatrix} \text{ and}$$

$$\nabla^2 l(\boldsymbol{\omega}) = -T \begin{pmatrix} \frac{\mathbf{ZZ}^\top}{T} \otimes \boldsymbol{\Theta} & -\left[\frac{2}{T}\mathbf{Z}(\mathbf{Y} - \mathbf{BZ})^\top \otimes \mathbf{I}_K\right] \mathbf{D}_K \\ -\mathbf{D}_K^\top \left[\frac{2}{T}(\mathbf{Y} - \mathbf{BZ})\mathbf{Z}^\top \otimes \mathbf{I}_K\right] & \frac{1}{2}\mathbf{D}_K^\top \left(\boldsymbol{\Theta}^{-1} \otimes \boldsymbol{\Theta}^{-1}\right) \mathbf{D}_K \end{pmatrix},$$

with $\mathbf{D}_K$ is a $K^2 \times K(K+1)/2$ duplication matrix. Appendix A shows the derivations. Define

$$\tilde{Q}(\boldsymbol{\omega}|\boldsymbol{\omega}^{(k)}) = \tilde{l}(\boldsymbol{\omega}|\boldsymbol{\omega}^{(k)}) - T\sum_j p'_{\lambda_j}(|\omega_j^{(k)}|)|\omega_j|. \tag{4.9}$$

Here, we note that the off-diagonal elements of $\boldsymbol{\Theta}$ are not penalized, and the tuning parameters corresponding to the autoregressive coefficients may be different from that of the inverse covariance. To maximize (4.9), we consider the following sub-problem in each iteration,

$$\boldsymbol{\Delta}^{(k)} = \arg\max_{\boldsymbol{\omega}} \left\{\tilde{Q}(\boldsymbol{\omega}|\boldsymbol{\omega}^{(k)})\right\} - \boldsymbol{\omega}^{(k)}. \tag{4.10}$$

Then, we obtain the update

$$\boldsymbol{\omega}^{(k+1)} = \boldsymbol{\omega}^{(k)} + \alpha^{(k)}\boldsymbol{\Delta}^{(k)}, \tag{4.11}$$

where $\alpha^{(k)} = \max\left\{2^{-v} : \tilde{Q}(\boldsymbol{\omega}^{(k)} + 2^{-v}\boldsymbol{\Delta}^{(v)}|\boldsymbol{\omega}^{(k)}) > \tilde{Q}(\boldsymbol{\omega}^{(k)}|\boldsymbol{\omega}^{(k)}), v \in \mathbb{N}_0\right\}$. Therefore, the algorithm for the estimation is:

(1) Initialize the estimate $\boldsymbol{\omega}^{(0)}$ to be the maximum likelihood estimate in (2.3).

(2) Solve the sub-problem in (4.10) by the alternating direction method of multiplier (ADMM) algorithm; see Boyd et al. (2011) for a review on the ADMM algorithm.

(3) Find the step size $\alpha^{(k)}$ satisfying the specified condition.

(4) Repeat Step 2 and 3, until a convergence criterion is met, say $|\frac{\tilde{Q}(\boldsymbol{\omega}^{(k+1)}|\boldsymbol{\omega}^{(k)}) - \tilde{Q}(\boldsymbol{\omega}^{(k)}|\boldsymbol{\omega}^{(k)})}{\tilde{Q}(\boldsymbol{\omega}^{(k)}|\boldsymbol{\omega}^{(k)})}| < \epsilon$ for some positive tolerance ϵ.

The Hessian matrix $\nabla^2 l(\boldsymbol{\omega})$ may not be negative definite, we therefore replace it by $T\mathbf{I}(\boldsymbol{\omega})$, where $\mathbf{I}(\boldsymbol{\omega})$ is the Fisher information matrix, in the algorithm.

Suppose the tuning parameters for the autoregressive coefficients part are λ_b and that for the inverse covariance part are λ_θ. The tuning parameters $(\lambda_b, \lambda_\theta)$ are chosen as described below. We first select a λ_b that minimizes the Bayesian information criterion (BIC) among the estimated models with various λ_b and without penalizing the inverse covariance. With the selected λ_b, the penalizing parameter for the inverse covariance λ_θ is identified similarly. We count all the non-zero autoregressive coefficients and the non-zero inverse noise covariance at the upper triangular part of the matrix as the number of parameters in the computation of the information criteria. We next study the finite sample properties of the discussed method through simulation experiments.

4.2 Numerical Results

4.2.1 Simulation

We consider six stable VAR models in the simulation study. Three out of the six models are selected from those studied in Section 3.2.1. The structure of the AR coefficients matrix and that of the inverse covariance matrix in each of these three models is the same. We are intrigued by the estimation performance of the penalized likelihood estimation method when such structure is not identical. We, therefore, consider three other models in which the AR coefficients matrix $\mathbf{A}_l$ and the inverse covariance matrix $\mathbf{\Theta} = \mathbf{\Sigma}_u^{-1}$ have different structures. The inverse covariance matrices are positive definite. We undergo the simulation experiments by considering the following models:

Model 1. $\mathbf{y}_t^{(1)} = \mathbf{A}_1^{(1)}\mathbf{y}_{t-1}^{(1)} + \mathbf{u}_t^{(1)}, \quad \mathbf{u}_t^{(1)} \sim N(\mathbf{0}, \mathbf{\Sigma}_1),$

Model 2. $\mathbf{y}_t^{(2)} = \mathbf{A}_1^{(2)}\mathbf{y}_{t-1}^{(2)} + \mathbf{u}_t^{(2)}, \quad \mathbf{u}_t^{(2)} \sim N(\mathbf{0}, \mathbf{\Sigma}_2),$

Model 3. $\mathbf{y}_t^{(3)} = \mathbf{A}_1^{(3)}\mathbf{y}_{t-1}^{(3)} + \mathbf{u}_t^{(3)}, \quad \mathbf{u}_t^{(3)} \sim N(\mathbf{0}, \mathbf{\Sigma}_3),$

Model 4. $\mathbf{y}_t^{(4)} = \mathbf{A}_1^{(4)}\mathbf{y}_{t-1}^{(4)} + \mathbf{u}_t^{(4)}, \quad \mathbf{u}_t^{(4)} \sim N(\mathbf{0}, \mathbf{\Sigma}_4),$

Model 5. $\mathbf{y}_t^{(5)} = \mathbf{A}_1^{(5)}\mathbf{y}_{t-1}^{(5)} + \mathbf{A}_2^{(5)}\mathbf{y}_{t-2}^{(5)} + \mathbf{u}_t^{(5)}, \quad \mathbf{u}_t^{(5)} \sim N(\mathbf{0}, \mathbf{\Sigma}_5),$

Model 6. $\mathbf{y}_t^{(6)} = \mathbf{A}_1^{(5)}\mathbf{y}_{t-1}^{(6)} + \mathbf{A}_2^{(5)}\mathbf{y}_{t-2}^{(6)} + \mathbf{u}_t^{(6)}, \quad \mathbf{u}_t^{(6)} \sim N(\mathbf{0}, \mathbf{\Sigma}_1),$

where

$$\mathbf{A}_1^{(1)} = \begin{pmatrix} 0.8 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.3 & 0 & 0 \\ 0 & 0 & 0 & 0 & -0.3 & 0 \\ 0.6 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.6 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.8 \end{pmatrix}, \ \mathbf{\Sigma}_1^{-1} = \begin{pmatrix} 1 & 0.3 & 0.3 & 0.3 & 0.3 & 0.3 \\ 0.3 & 1 & 0 & 0 & 0 & 0 \\ 0.3 & 0 & 1 & 0 & 0 & 0 \\ 0.3 & 0 & 0 & 1 & 0 & 0 \\ 0.3 & 0 & 0 & 0 & 1 & 0 \\ 0.3 & 0 & 0 & 0 & 0 & 1 \end{pmatrix},$$

$$\mathbf{A}_1^{(2)} = \mathbf{A}_1^{(3)} = \begin{pmatrix} 0.4352 & -0.6552 & 0.4154 & 0.3930 & -0.5200 & 0.2256 \\ 0.1478 & -0.4932 & 0 & 0 & 0 & 0 \\ -0.7940 & 0 & -0.8933 & 0 & 0 & 0 \\ 0.5894 & 0 & 0 & -0.1478 & 0 & 0 \\ -0.8009 & 0 & 0 & 0 & -0.4169 & 0 \\ 0.4197 & 0 & 0 & 0 & 0 & -0.2439 \end{pmatrix}, \ \mathbf{\Sigma}_3^{-1} = \begin{pmatrix} 1 & 0.4 & 0.4 & 0.4 & 0.4 & 0.4 \\ 0.4 & 1 & 0 & 0 & 0 & 0 \\ 0.4 & 0 & 1 & 0 & 0 & 0 \\ 0.4 & 0 & 0 & 1 & 0 & 0 \\ 0.4 & 0 & 0 & 0 & 1 & 0 \\ 0.4 & 0 & 0 & 0 & 0 & 1 \end{pmatrix},$$

$$\mathbf{A}_1^{(4)} = \begin{pmatrix} 0.2177 & 0.3066 & 0 & 0 & 0 & 0.3775 \\ -0.6324 & -0.6650 & 0.0214 & 0 & 0 & 0 \\ 0 & -0.2749 & -0.7509 & 0.4482 & 0 & 0 \\ 0 & 0 & -0.3046 & -0.8066 & 0.9940 & 0 \\ 0 & 0 & 0 & -0.7313 & 0.5054 & 0.7959 \\ -0.0587 & 0 & 0 & 0 & -0.5140 & -0.9470 \end{pmatrix}, \ \boldsymbol{\Sigma}_2^{-1} = \boldsymbol{\Sigma}_4^{-1} = \begin{pmatrix} 1 & 0.4 & 0 & 0 & 0 & 0.4 \\ 0.4 & 1 & 0.4 & 0 & 0 & 0 \\ 0 & 0.4 & 1 & 0.4 & 0 & 0 \\ 0 & 0 & 0.4 & 1 & 0.4 & 0 \\ 0 & 0 & 0 & 0.4 & 1 & 0.4 \\ 0.4 & 0 & 0 & 0 & 0.4 & 1 \end{pmatrix},$$

$$\mathbf{A}_1^{(5)} = \begin{pmatrix} -0.6 & 0.4 & 0 & 0 & 0 & 0.4 \\ 0.4 & -0.6 & 0.4 & 0 & 0 & 0 \\ 0 & 0.4 & -0.6 & 0.4 & 0 & 0 \\ 0 & 0 & 0.4 & -0.6 & 0.4 & 0 \\ 0 & 0 & 0 & 0.4 & -0.6 & 0.4 \\ 0.4 & 0 & 0 & 0 & 0.4 & -0.6 \end{pmatrix}, \ \mathbf{A}_2^{(5)} = \begin{pmatrix} -0.3 & 0.2 & 0 & 0 & 0 & 0.2 \\ 0.2 & -0.3 & 0.2 & 0 & 0 & 0 \\ 0 & 0.2 & -0.3 & 0.2 & 0 & 0 \\ 0 & 0 & 0.2 & -0.3 & 0.2 & 0 \\ 0 & 0 & 0 & 0.2 & -0.3 & 0.2 \\ 0.2 & 0 & 0 & 0 & 0.2 & -0.3 \end{pmatrix} \text{ and}$$

$$\boldsymbol{\Sigma}_5^{-1} = \begin{pmatrix} 1 & -0.3 & 0 & 0 & 0 & -0.3 \\ -0.3 & 1 & -0.3 & 0 & 0 & 0 \\ 0 & -0.3 & 1 & -0.3 & 0 & 0 \\ 0 & 0 & -0.3 & 1 & -0.3 & 0 \\ 0 & 0 & 0 & -0.3 & 1 & -0.3 \\ -0.3 & 0 & 0 & 0 & -0.3 & 1 \end{pmatrix}.$$

Model 1 is a 6-dimensional VAR(1) model examined by Davis et al. (2016), except that we impose the same structure on the inverse innovation covariance matrix rather than the innovation covariance matrix. Model 2 is a 6-dimensional VAR(1) model with every node connected by directed edges to the first node in the causality graph, and the inverse noise covariance matrix has a Toeplitz structure. Models 3, 4 and 5 follow from three of the investigated models in Section 3.2.1. Model 6 is a 6-dimensional VAR model of lag 2 with Toeplitz AR coefficients matrices and an inverse covariance matrix of another structure.

We perform the simulation study with sample size T of 100, 200, 500 and 1000 over 500 replications using R (R Core Team, 2017). As a comparison to the estimation performance with various penalty functions, we carry out the simulation experiments with three penalty functions, including L_1, SCAD, and MCP. The penalty functions are the same for penalizing the AR coefficients and the inverse covariance matrix in each estimation procedure. We fix the penalty parameter as follows, $a = 3.7$ in (4.2) when the SCAD penalty is used or $\gamma = 2$ in (4.3) when the MCP is applied. The tuning parameters $(\lambda_b, \lambda_\theta)$ are chosen as described in Section 4.1.2. We also assume the lag order p is known in advance throughout the simulation experiments.

As a comparison to the introduced estimation method, we adopt the multivariate regression with covariance estimation (MRCE) method, proposed

by Rothman et al. (2010), in the simulation study. The MRCE method estimates the regression coefficients and the inverse covariance matrix through maximizing the L_1 penalized likelihood. This method, in particular, finds a solution by alternating maximization.

To compare the estimation performance, we compile the bias, variance and mean squared error (MSE) of the estimates introduced in Section 3.2.1. Besides these three metrics, we also compute the average number of coefficients that are set to zero correctly and incorrectly for both the AR coefficients and the inverse noise covariance matrix. For the inverse noise covariance matrix, only the upper triangular elements are considered in the calculation because of symmetry.

We also compile the divergence of the estimated model spectrum $\hat{\mathbf{f}}(\lambda)$ from the true spectrum $\mathbf{f}(\lambda)$ for comparison. The divergence is defined as

$$\frac{1}{2\pi}\int_0^{\pi} \text{trace}\left[\mathbf{f}(\lambda)\hat{\mathbf{f}}^{-1}(\lambda) - \mathbf{I}_K\right] - \log\det\left[\mathbf{f}(\lambda)\hat{\mathbf{f}}^{-1}(\lambda)\right]\, d\lambda.$$

This quantity measures the discrepancy between the true spectrum and the estimated model spectrum. It is positively valued unless the model spectrum and the true spectrum are identical which gives a zero value. For the computation of the spectrum, we refer the reader to Chapter 11 of Brockwell & Davis (1991). We delineate the results for Model 3 and 6 in this section, others are reported in Appendix B.

Tables 4.1 and 4.2 document the bias, variance and the mean squared error (MSE) of the estimates using three different penalties and the MRCE method. The column 'Size', 'Penalty', and 'Divergence' are, respectively, the sample size, the penalty or method implemented, and the average divergence value over 500 replications. Standard errors are in the parentheses.

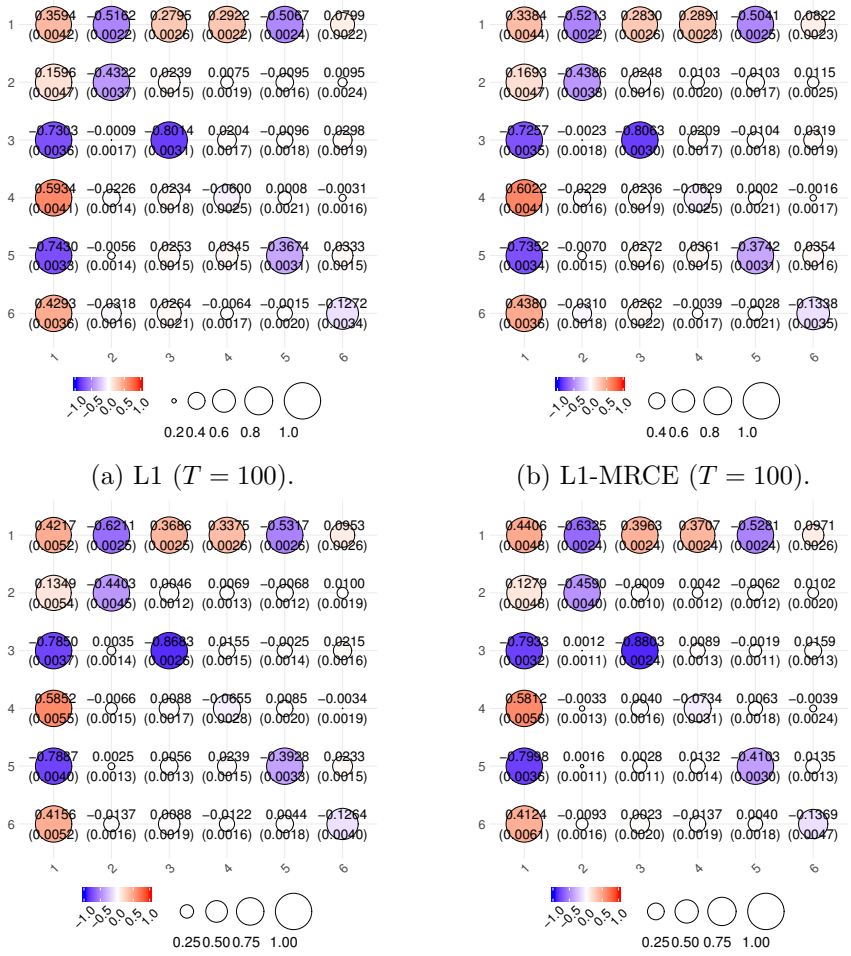
Model 3

TABLE 4.1: Simulation results for Model 3 over 500 replications. Zeros_C (Zeros_I) is the average number of zero coefficients correctly (incorrectly) estimated to be zero. Standard errors are in the parentheses.

			$\hat{\mathbf{A}}$					$\hat{\mathbf{\Sigma}}_y^{-1}$				
Size	Penalty	Divergence	Bias	Variance	MSE	Zeros_C	Zeros_I	Bias	Variance	MSE	Zeros_C	Zeros_I
100	L1	0.2544 (0.0033)	1.4898	0.1167	0.2433	11.8040 (0.1083)	0.6960 (0.0355)	0.7309	0.1873	0.2376	6.3240 (0.0830)	0 (0)
	L1-MRCE	0.2511 (0.0030)	1.5320	0.1213	0.2480	11.4060 (0.1093)	0.6180 (0.0327)	0.7263	0.1890	0.2348	6.5340 (0.0815)	0 (0)
	MCP	0.2260 (0.0030)	0.6057	0.1391	0.1772	16.4020 (0.0894)	1.1440 (0.0413)	0.6384	0.2583	0.2989	9.5000 (0.0375)	0.0040 (0.0028)
	SCAD	0.2261 (0.0029)	0.8292	0.1433	0.1947	13.7260 (0.0979)	0.7600 (0.0351)	0.5846	0.2460	0.2820	9.5140 (0.0418)	0.0020 (0.0020)
200	L1	0.1278 (0.0015)	1.0271	0.0558	0.1151	11.6120 (0.1041)	0.1440 (0.0167)	0.7084	0.0859	0.1251	6.3860 (0.0803)	0 (0)
	L1-MRCE	0.1277 (0.0015)	1.0517	0.0557	0.1164	11.5000 (0.1053)	0.1460 (0.0168)	0.6516	0.0915	0.1255	6.4500 (0.0825)	0 (0)
	MCP	0.1026 (0.0014)	0.3571	0.0578	0.0742	17.2500 (0.0736)	0.4580 (0.0287)	0.2873	0.0977	0.1059	9.7260 (0.0258)	0 (0)
	SCAD	0.1066 (0.0014)	0.5113	0.0587	0.0858	14.7720 (0.0927)	0.2540 (0.0214)	0.2391	0.0941	0.1001	9.8040 (0.0285)	0 (0)
500	L1	0.0531 (0.0006)	0.6554	0.0204	0.0448	12.1480 (0.1032)	0 (0)	0.5940	0.0336	0.0605	6.5100 (0.0757)	0 (0)
	L1-MRCE	0.0516 (0.0006)	0.6493	0.0205	0.0437	11.9140 (0.1008)	0 (0)	0.5616	0.0346	0.0585	6.5900 (0.0744)	0 (0)
	MCP	0.0329 (0.0004)	0.1114	0.0146	0.0159	18.4980 (0.0544)	0.0040 (0.0028)	0.1375	0.0337	0.0356	9.7860 (0.0212)	0 (0)
	SCAD	0.0372 (0.0005)	0.2299	0.0181	0.0242	16.1540 (0.0759)	0.0080 (0.0040)	0.0907	0.0343	0.0352	9.8920 (0.0181)	0 (0)
1000	L1	0.0281 (0.0003)	0.4609	0.0099	0.0226	12.4120 (0.0996)	0 (0)	0.4726	0.0170	0.0343	6.6420 (0.0731)	0 (0)
	L1-MRCE	0.0268 (0.0003)	0.4458	0.0099	0.0214	12.0460 (0.0984)	0 (0)	0.4454	0.0174	0.0326	6.7620 (0.0716)	0 (0)
	MCP	0.0154 (0.0002)	0.0416	0.0063	0.0064	19.1200 (0.0402)	0 (0)	0.0613	0.0157	0.0161	9.8780 (0.0167)	0 (0)
	SCAD	0.0172 (0.0002)	0.1325	0.0078	0.0097	17.3600 (0.0724)	0 (0)	0.0431	0.0157	0.0159	9.9560 (0.0100)	0 (0)

Table 4.1 is the simulation results for Model 3 using three different penalties and the MRCE method. The models estimated using the two non-convex penalties possess less discrepancy from the true spectrum comparing with that using the L_1 penalty. All methods improve in the estimation as the sample size raises. The two non-convex penalties perform more satisfactorily than the L_1 penalty, concerning the bias of the estimates. This is because of the statistical bias issue when using the L_1 penalty and the better ability in identifying the zero coefficients correctly. The two non-convex penalties, however, set the non-zero coefficients to zero erroneously in more experiments comparing to that using the L_1 penalty. This leads to a relatively higher variance of the estimates, especially when the sample size is below 200. The rise of sample size mitigates the misidentification of zero coefficients. Both the proposed method using the L_1 penalty and the MRCE method obtain estimates that are close to each other. The inverse

covariance estimates obtained by the MRCE method possess less bias than that using the proposed method with the L_1 penalty.



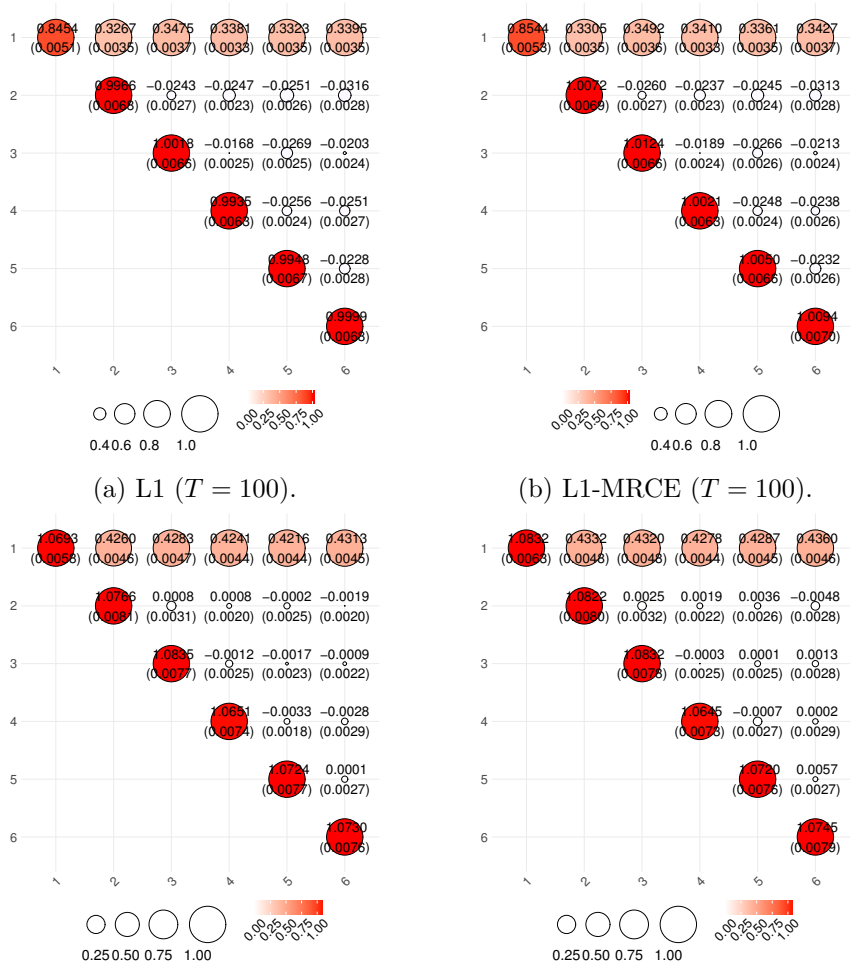
(a) L1 ($T = 100$). (b) L1-MRCE ($T = 100$).

(c) SCAD ($T = 100$). (d) MCP ($T = 100$).

FIGURE 4.3: Average values of the AR coefficient estimates for Model 3. Standard errors are in parentheses.

Figure 4.3 depicts the average AR estimates using the four studied methods, with a sample size of 100, by dot plots. The average estimate value characterizes the colour in the corresponding dot. The dot size is characterized by the proportion of experiments, out of 500 replications, that the corresponding estimate is not set to zero. According to Figure 4.3, the average AR estimates using the L_1 penalty deviate more from the true parameter values comparing to that using the two non-convex penalties. All four studied methods set the non-zero coefficients to be zero incorrectly in

some experiments, especially at the positions (1,6), (4,4), and (6,6). These coefficients are of small magnitude comparing to others.



(a) L1 ($T = 100$). (b) L1-MRCE ($T = 100$).

(c) SCAD ($T = 100$). (d) MCP ($T = 100$).

FIGURE 4.4: Average values of the inverse covariance estimates for Model 3. Standard errors are in parentheses.

Figure 4.4 visualize the average inverse covariance estimates using the four investigated methods by dot plots. We can observe from Figure 4.4(a) and Figure 4.4(b) that the inverse covariance estimates using the L_1 penalty possess higher bias relative to the estimates with the two non-convex penalties. All methods shrink the zero coefficients in most experiments, and the two non-convex penalties determine the zero coefficients in more replications.

Model 6

TABLE 4.2: Simulation results for Model 6 over 500 replications. Zeros_C (Zeros_I) is the average number of zero coefficients correctly (incorrectly) estimated to be zero. Standard errors are in the parentheses.

			$\hat{\mathbf{A}}$					$\hat{\mathbf{\Sigma}}_u^{-1}$				
Size	Penalty	Divergence	Bias	Variance	MSE	Zeros_C	Zeros_I	Bias	Variance	MSE	Zeros_C	Zeros_I
100	L1	0.4464 (0.0059)	4.0268	0.4481	0.8063	19.6960 (0.1772)	2.1580 (0.0956)	0.6680	0.2245	0.2564	6.6040 (0.0897)	0 (0)
	L1-MRCE	0.4480 (0.0061)	4.0448	0.4491	0.8069	19.9680 (0.1741)	2.2600 (0.1004)	0.6875	0.2491	0.2793	6.6220 (0.0933)	0 (0)
	MCP	0.5030 (0.0055)	2.1281	0.9355	1.0493	29.2760 (0.1220)	5.5180 (0.1109)	0.9113	0.3525	0.4385	9.3340 (0.0448)	0.0280 (0.0074)
	SCAD	0.4614 (0.0046)	2.2290	0.8470	0.9651	23.9380 (0.1542)	3.0660 (0.0840)	0.9558	0.3490	0.4443	9.3020 (0.0516)	0.0040 (0.0028)
200	L1	0.2140 (0.0019)	2.6637	0.2188	0.3723	19.4680 (0.1695)	0.3080 (0.0257)	0.6152	0.0876	0.1181	6.8420 (0.0785)	0 (0)
	L1-MRCE	0.2122 (0.0018)	2.6704	0.2174	0.3701	19.4800 (0.1661)	0.3380 (0.0273)	0.5390	0.0921	0.1170	6.7880 (0.0831)	0 (0)
	MCP	0.2050 (0.0022)	0.9213	0.3389	0.3645	31.4760 (0.1008)	1.8780 (0.0635)	0.4162	0.1131	0.1305	9.6020 (0.0326)	0 (0)
	SCAD	0.2021 (0.0019)	1.1555	0.3232	0.3611	26.7840 (0.1339)	0.8760 (0.0389)	0.3622	0.1114	0.1251	9.7020 (0.0339)	0 (0)
500	L1	0.0888 (0.0009)	1.7505	0.0798	0.1471	20.2320 (0.1574)	0.0040 (0.0028)	0.5498	0.0346	0.0579	6.7520 (0.0694)	0 (0)
	L1-MRCE	0.0877 (0.0008)	1.7400	0.0800	0.1463	20.3460 (0.1541)	0.0040 (0.0028)	0.5124	0.0354	0.0554	6.7520 (0.0720)	0 (0)
	MCP	0.0627 (0.0007)	0.2676	0.0805	0.0836	33.7660 (0.0703)	0.2040 (0.0197)	0.1752	0.0348	0.0378	9.8100 (0.0211)	0 (0)
	SCAD	0.0706 (0.0007)	0.5493	0.0952	0.1064	30.5340 (0.1011)	0.0780 (0.0127)	0.1232	0.0352	0.0368	9.8860 (0.0201)	0 (0)
1000	L1	0.0472 (0.0004)	1.2915	0.0382	0.0755	20.9220 (0.1503)	0 (0)	0.4646	0.0171	0.0342	6.9520 (0.0701)	0 (0)
	L1-MRCE	0.0467 (0.0004)	1.2941	0.0382	0.0755	21.1860 (0.1552)	0 (0)	0.4294	0.0175	0.0318	6.9720 (0.0698)	0 (0)
	MCP	0.0269 (0.0003)	0.0847	0.0288	0.0291	35.2560 (0.0422)	0.0020 (0.0020)	0.0840	0.0153	0.0160	9.9080 (0.0135)	0 (0)
	SCAD	0.0310 (0.0003)	0.2691	0.0371	0.0398	32.5780 (0.0808)	0.0020 (0.0020)	0.0634	0.0155	0.0160	9.9500 (0.0120)	0 (0)

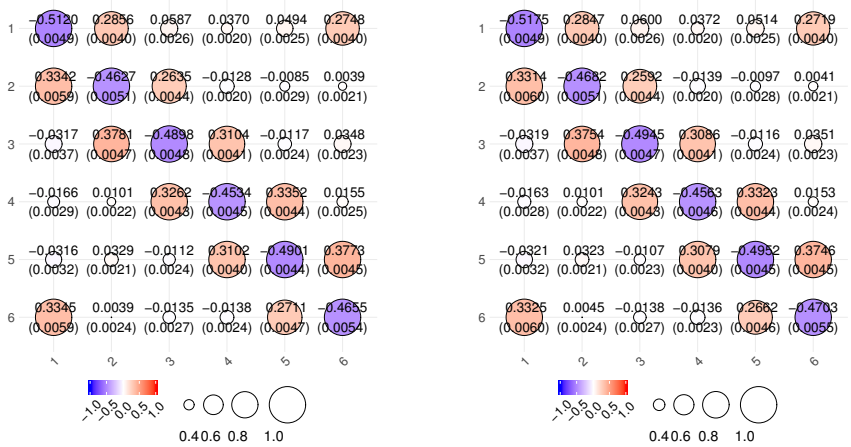
Table 4.2 reports the simulation results for Model 6 using three different penalties and the MRCE method with the L_1 penalty. The average divergence of the estimates using the two non-convex penalties is lower than that using the L_1 penalty when the sample size is 200 or above. For the experiments with a sample size of 100, the models estimated using the L_1 penalty have less discrepancy from the actual model spectrum, comparing to that using the two non-convex penalties. This difference is because the two non-convex penalties identify the non-zero lag 2 AR coefficients to be zero mistakenly in more experiments, comparing to that using the L_1 penalty; see Figure 4.6. The AR estimates that are penalized by the non-convex penalties possess less bias than the estimates that are penalized by the L_1 penalty. This is because of the unbiasedness property of the non-convex

penalties and the better ability in determining the zero coefficients. The two non-convex penalties, however, set the non-zero AR coefficients to zero erroneously in more experiments with a sample size of 100. This is more significant for the lag 2 AR coefficients, comparing to the estimates using the L_1 penalty; see Figure 4.6. Such misidentification leads to a higher variance of the estimates, especially when the sample size is low. These circumstances are alleviated as the sample size raises. Comparing the introduced method using the L_1 penalty with the MRCE method, both methods obtain AR coefficient estimates that are close to each other, and the inverse covariance estimates obtained by the MRCE method possess less bias with moderate to large sample size. The discussed method outperforms the MRCE method when the sample size is 100.

Figure 4.5 (4.6) delineates the average lag 1 (lag 2) AR coefficient estimates obtained by the four studied methods, when the sample size is 100, by dot plots. The average estimate value characterizes the colour of the corresponding dot. The proportion of experiments that an estimate is non-zero characterizes the corresponding dot size. According to Figures 4.5 and 4.6, the average AR estimates using the two non-convex penalties deviate less from the actual parameters, comparing to that using L_1 penalty.

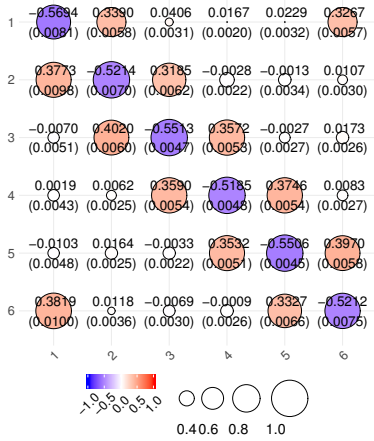
We can observe from Figure 4.6 that the discrepancy of the estimates from the actual parameters is more significant for the lag 2 AR coefficients. This is because the four studied methods misidentify the off-diagonal non-zero coefficients to be zero in some simulation experiments, especially at the positions (1,2) and (1,6). The two non-convex penalties shrink some of the non-zero lag 2 AR coefficients erroneously in more experiments relative to that using L_1 penalty.

Figure 4.7 displays the average inverse covariance estimates using the

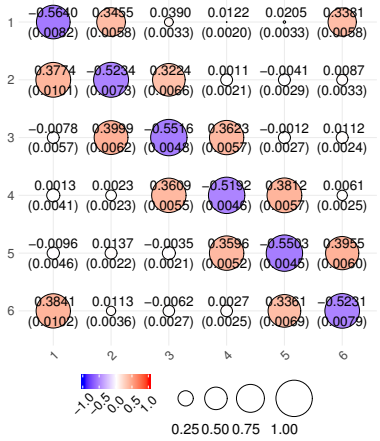


(a) L1 ($T = 100$).

(b) L1-MRCE ($T = 100$).



(c) SCAD ($T = 100$).

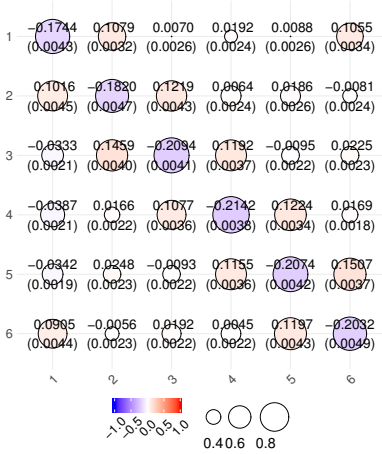


(d) MCP ($T = 100$).

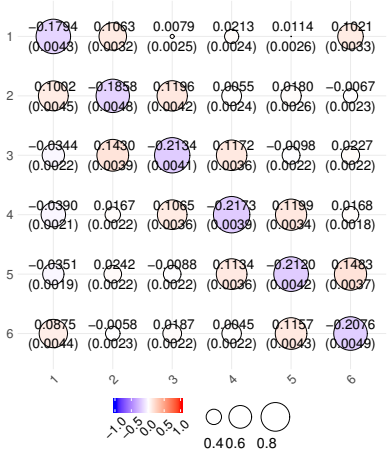
FIGURE 4.5: Average values of the lag 1 AR coefficient estimates for Model 6. Standard errors are in parentheses.

four investigated methods by dot plots. As shown in the figure, the non-zero inverse covariance estimates using the non-convex penalties possess slightly larger bias than that using the L_1 penalty. The two non-convex penalties, however, identify the zero inverse covariances correctly more often.

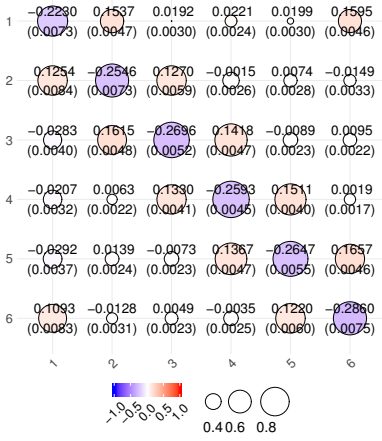
In summary, all methods improve in the estimation bias, variance, and MSE when the sample size T increases. The penalized estimates using the two non-convex penalties, in general, posses less bias and divergence than the estimates penalized by the L_1 penalty. This is because the estimation using the two non-convex penalties identify zero coefficients correctly in more cases. The non-zero estimates using the two non-convex penalties



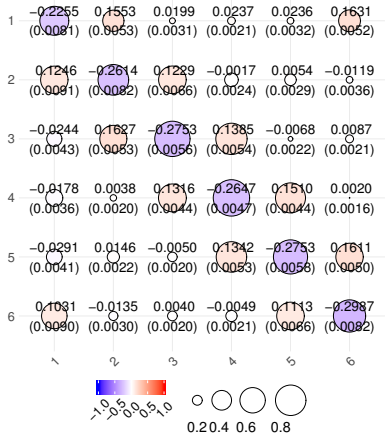
(a) L1 ($T = 100$).



(b) L1-MRCE ($T = 100$).



(c) SCAD ($T = 100$).

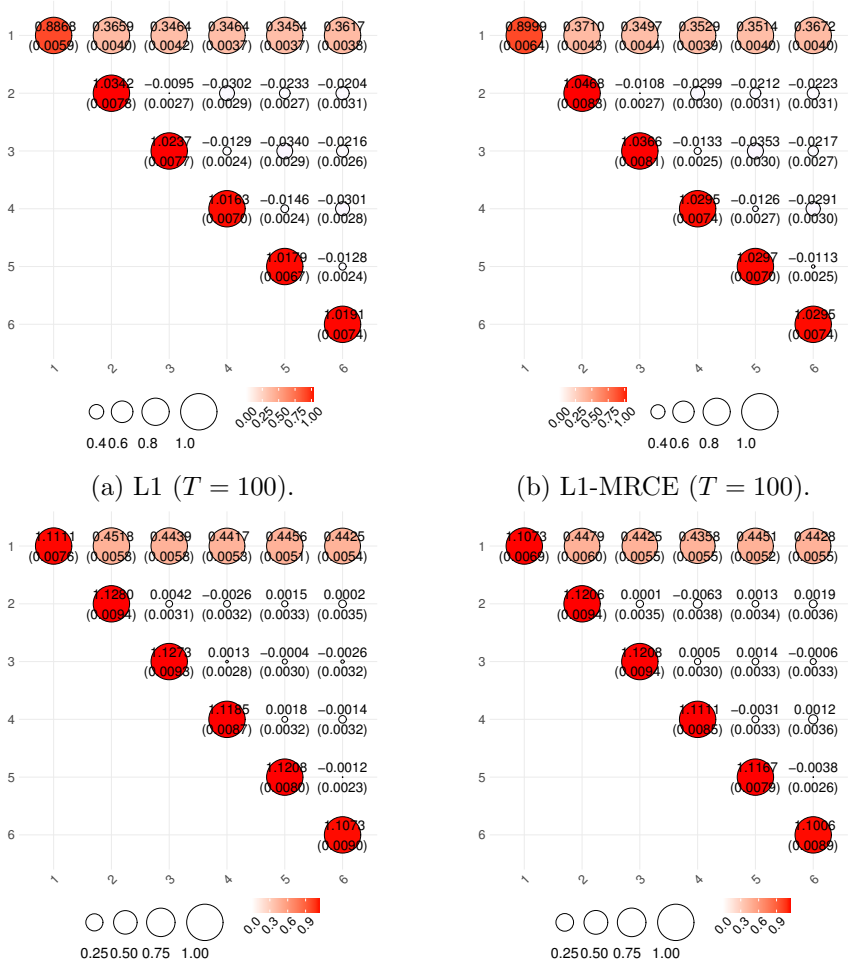


(d) MCP ($T = 100$).

FIGURE 4.6: Average values of the lag 2 AR coefficient estimates for Model 6. Standard errors are in parentheses.

have less bias comparing to the estimates penalized by the L_1 penalty. These circumstances are particularly more apparent with large sample size. The results align with the statistical bias issue when using the L_1 penalty.

A notable number of coefficients carrying marginally significant coefficients in few studied models are identified inaccurately to be zero. This indicates the performance of the penalization method is weakened when determining marginally significant coefficients, especially when the sample size is small. We demonstrate the penalized estimation method by an application in the next section.



(a) L1 ($T = 100$). (b) L1-MRCE ($T = 100$).

(c) SCAD ($T = 100$). (d) MCP ($T = 100$).

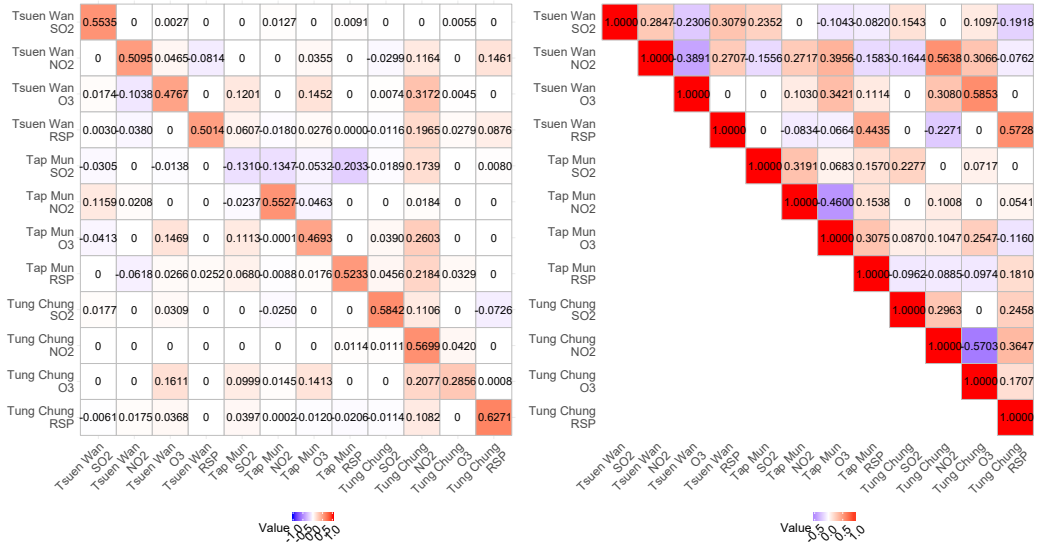
FIGURE 4.7: Average values of the inverse covariance estimates for Model 6. Standard errors are in parentheses.

4.2.2 Application

Air pollution data in Hong Kong

We employ the penalized likelihood estimation method to an air pollution data in Hong Kong. The data has been investigated by Hu et al. (2016) and consists of the daily average concentration of four air pollutants recorded at three air monitoring stations from September 2010 to September 2014. These stations are located at Tsuen Wan (TW), Tung Chung (TC) and Tap Mun (TM). The four air pollutants are sulphur dioxide (SO_2), nitrogen dioxide (NO_2), ozone (O_3) and respirable suspended particulates (RSP).

The series has a length of 1491 days. We adopt the same data treatment method suggested by the authors before applying the penalized likelihood estimation. The authors, in particular, first stabilize the variance of each series by Box-Cox transformation (Box & Cox, 1964) followed by deseasonalizing the transformed series by harmonic regression (McLeod & Gweon, 2013). The data is preprocessed since the marginal variance of the series changes over time and the series possesses seasonality with a period of about 365 days. The transformed SO_2 series at TC is further first-order differenced to achieve second-order stationarity.



(a) AR coefficients. (b) Partial correlations of innovations.

FIGURE 4.8: The autoregressive coefficient estimates and the estimated partial correlations of innovations using the penalized likelihood estimation method for the PRDR air pollution data.

With the preprocessed data, we first determine the lag order p of the VAR model by selecting a saturated VAR model that carries the minimum BIC value among the models with different lag orders, for instance, p ranges from 1 to 6. We proceed to the penalized likelihood estimation on VAR models, using the selected lag order, with SCAD penalties for both the AR coefficients and the inverse covariances. Similar to the procedure we implemented in the simulation studies (Section 4.2.1), we fix the penalty parameters a

to be 3.7 for all penalties. We pick the suitable tuning parameters $(\lambda_b, \lambda_\theta)$ that carry the least BIC value among various models with different tuning parameters.

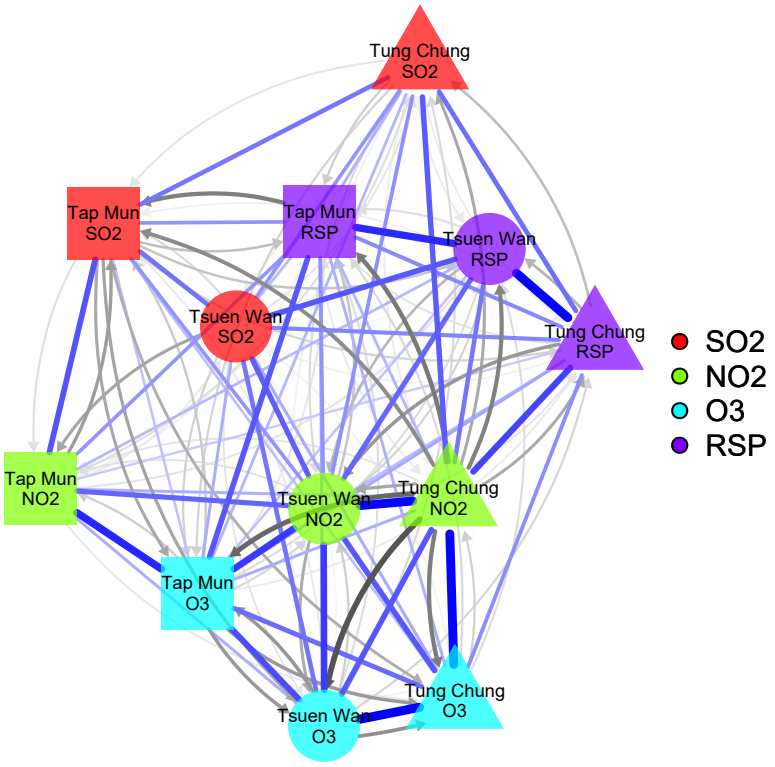
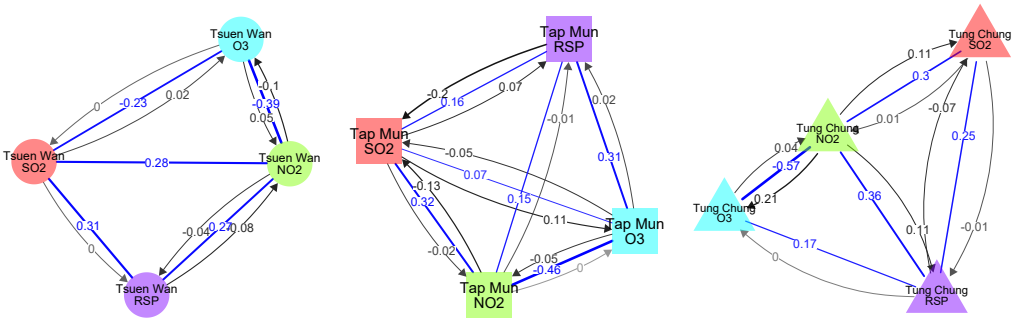


FIGURE 4.9: A mixed graph visualizing the estimated VAR model for the Hong Kong air pollution data. The blue line represents the undirected edge determined by the inverse of noise covariance matrix, the black arrow is the directed edge characterized by the AR coefficient.

Figures 4.8(a) and 4.8(b) report, respectively, the penalized AR coefficient estimates and the estimated partial correlation coefficients of the noise terms obtained by normalizing the inverse covariances. Figure 4.9 depicts the estimated sparse VAR model by the causality graph introduced in Section 2.2.2. Each node represents an air pollutant series at a particular station. The shape (colour) of a node is characterized by the location (pollutant) of the corresponding series. The opacity of an edge reflects the magnitude of the corresponding coefficient. Figure 4.9 is a complicated

graph. We, therefore, consider its subgraphs for interpretations. We generate two types of subgraphs which are grouped by location (Figure 4.10) and pollutant (Figure 4.11).

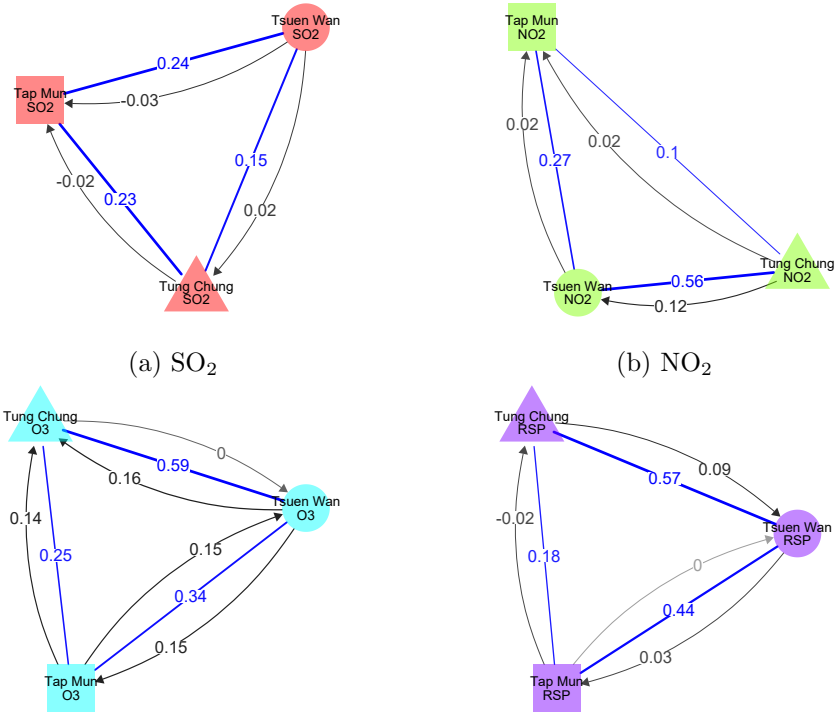


(a) Tsuen Wan (b) Tap Mun (c) Tung Chung

FIGURE 4.10: A mixed graph visualizing the estimated VAR model for the Hong Kong air pollution data grouped by location.

Figure 4.10 shows the interactions between pollutants at the same station. For each of the three air monitoring stations, there is an undirected edge connecting the nodes of NO_2 and O_3 with negative partial correlation coefficient. This perhaps indicates the reaction between NO_2 and O_3 in ambient air. Figures 4.10(a), 4.10(b) and 4.10(c) show an undirected edge links the vertices of NO_2 (SO_2) and RSP at each of the three locations. This probably reflects the formation of RSP from the atmospheric oxidation of gaseous pollutants, like SO_2 and NO_2. We can observe from Figures 4.10(b) and 4.10(c) that an undirected edge bridges the nodes of O_3 and RSP at each of the two corresponding air monitoring stations. A possible reason for such observation is that RSP can be formed by photochemical reactions, involving O_3, under sunlight. The partial correlation coefficient between the two pollutants at TW is less significant than that at the other two locations which can be a result of the relatively lower concentration of O_3 at TW.

Figure 4.11 visualizes the interaction of a pollutant between the three stations. The magnitudes of the partial correlation coefficients between



(a) SO_2 (b) NO_2

(c) O_3 (d) RSP

FIGURE 4.11: A mixed graph visualizing the estimated VAR model for the Hong Kong air pollution data grouped by pollutant.

locations of the same pollutant possibly parallel with the transmission distances between the stations, since the partial correlation coefficients reflect the contemporaneous conditional interdependencies among the components. For instance, the flight distances between TC–TW, TW–TM and TC–TM are 25km, 33km, and 58km, respectively; and the partial correlation coefficients of RSP between TC–TW, TW–TM, and TC–TM are, respectively, 0.57, 0.44 and 0.18. These observations are also evident for the pollutants NO_2 (Figure 4.11(b)) and O_3 (Figure 4.11(c)). Figure 4.11(c) displays the interaction of O_3 between the three air monitoring stations, which consists of a number of significant directed and undirected edges between the nodes. This may indicate O_3 is a regional air pollution problem in both short and long terms.

4.3 Summary

In the current chapter, we have discussed a penalized likelihood estimation method on sparse vector autoregressive (VAR) models. The autoregressive (AR) coefficients and the off-diagonal elements of the inverse covariance matrix are penalized for achieving parsimonious models. We have applied the local linear approximation (LLA) to the penalty function and obtained the penalized estimates iteratively. Simulation studies were conducted to investigate the finite sample properties of the penalized likelihood estimator. The studies suggest that the penalization method work satisfactorily in promoting model sparsity in the absence of prior information about the sparsity structure. We utilized the introduced penalized likelihood estimation method to a real data for illustration. The real data application demonstrates that the penalization method improves the model interpretability.

Chapter 5

Conclusions

In summary, we discussed the estimation of graphical time series models based on sparse Gaussian vector autoregressive (VAR) processes. Two estimation methods for the sparse VAR models are presented, namely the constrained likelihood estimation method and the penalized likelihood estimation method.

The constrained likelihood estimation method estimates sparse vector autoregressive models by considering the maximum likelihood estimation with sparsity constraints on both the autoregressive coefficients and the inverse noise covariance matrix as a biconcave problem. An alternating maximization method is utilized to solve the biconcave problem. Simulation experiments study the estimation performance of this alternating method and compare with other non-linear optimization methods. The simulation results reflect that the alternating method is more robust whereas the compared methods failed to converge in some cases. We also introduce a frequency domain method and a time domain method for identifying the

sparsity structure. In the application section, the proposed method is comparable to another graphical time series model based on the parsimonious structural vector autoregressive models.

The sparsity constraints in the constrained likelihood estimation method, however, are required to be identified before the estimation procedure. A penalized likelihood estimation of vector autoregressive models is proposed, in Chapter 4, to achieve model sparsity in the estimation. This penalization method implements penalty terms on the autoregressive coefficients and the off-diagonal elements of the inverse covariance matrix to achieve parsimonious model. The finite sample properties of the penalized likelihood estimator are investigated by performing simulation experiments. The simulation studies suggest that the penalization method work satisfactorily in achieving sparse VAR models without prior determination of the sparsity constraint structure. The application section illustrates that the penalization method augments the interpretability of high dimensional graphical time series models by promoting model sparsity in the estimation.

We discuss some possible future research below. In the constrained likelihood estimation method, the frequency domain method sets an AR coefficient to zero when the corresponding partial spectral coherencies are insignificant at all frequencies. Indeed, there are VAR models in which a non-zero AR coefficient corresponds to zero partial spectral coherencies. We consider the following three-dimensional VAR(1) process for illustration,

$$\begin{pmatrix} x_{1,t} \\ x_{2,t} \\ x_{3,t} \end{pmatrix} = \begin{pmatrix} 0.8 & 0.3 & 0.4 \\ 0 & 0.7 & 0.4 \\ 0 & 0.3 & 0.5 \end{pmatrix} \begin{pmatrix} x_{1,t-1} \\ x_{2,t-1} \\ x_{3,t-1} \end{pmatrix} + \begin{pmatrix} \varepsilon_{1,t} \\ \varepsilon_{2,t} \\ \varepsilon_{3,t} \end{pmatrix},$$

where $\boldsymbol{\varepsilon}_t = (\varepsilon_{1,t}, \varepsilon_{2,t}, \varepsilon_{3,t})^\top \sim N(\mathbf{0}, \boldsymbol{\Sigma})$ with $\boldsymbol{\Sigma}^{-1} = \begin{pmatrix} 1 & 0 & -1 \\ 0 & 3 & 1 \\ -1 & 1 & 2 \end{pmatrix}$. The partial

spectral coherencies of $x_{1,t}$ and $x_{2,t}$ given $x_{3,t}$ are zero at all frequencies, while the corresponding AR coefficient is non-zero. The partial spectral coherence method in identifying the sparsity structure probably sets such non-zero AR coefficient to zero erroneously. To mitigate such misidentification, we may utilize the partial directed coherence (Baccalá & Sameshima, 2001) by factorizing the inverse of the spectral density matrix (Amblard, 2015), instead of the partial spectral coherence to identify the possible constraint structure. The partial directed coherence is a frequency domain measure for Granger causality. The factorization may also determine the possible sparsity structures of the AR coefficients and the inverse covariance matrix simultaneously.

In the penalized likelihood estimation method, we select the tuning parameters $(\lambda_b, \lambda_\theta)$ by BIC and is intriguing to study the selection method using other metrics further, like the forecast error. We may determine the lag order of the model by implementing the group lasso penalty (Yuan & Lin, 2006). That is, we can consider the following penalized likelihood function:

$$Q(\mathbf{B}, \mathbf{\Theta}) = l(\mathbf{B}, \mathbf{\Theta}) - T \sum_{i,j} p_{\lambda_b}(|b_{ij}|) - T \sum_{i \neq j} p_{\lambda_\theta}(|\theta_{ij}|) - (1-\alpha)\sqrt{K}\lambda_b \sum_{l=1}^{p} \|\mathbf{A}_l\|_F,$$

where $l(\mathbf{B}, \mathbf{\Theta})$ is in (4.7). A similar study has been investigated by Nicholson et al. (2017). We may also investigate the asymptotic properties of the penalized likelihood estimator, including the rate of convergence, the sparsistency, and the asymptotic normality of the estimator.

Appendix A

Proofs

Lemma A.1. *For any positive semidefinite matrices* $\mathbf{A}$ *and* $\mathbf{B}$ *and* $0 < \alpha < 1$, $\det[\alpha\mathbf{A} + (1-\alpha)\mathbf{B}] \geq \det(\mathbf{A})^{\alpha}\det(\mathbf{B})^{1-\alpha}$ *with equality if and only if* $\mathbf{A} = \mathbf{B}$ *or* $\det[\alpha\mathbf{A} + (1-\alpha)\mathbf{B}] = 0$.

Proof. See Magnus & Neudecker (1999, Chapter 11). □

We recall Theorem 3.1 in Chapter 3:

Theorem. *The optimization problem in* (3.2) *with respect to* $\mathbf{B}$ *and* $\boldsymbol{\Theta}$ *is biconcave.*

Proof. To prove the problem in (3.2) is biconcave, we first show the feasible set $\mathcal{D}$ is biconvex followed by showing the objective function of the problem is biconcave. Let $\mathcal{S}$ be the set of lower triangular positions of $\boldsymbol{\Theta}$ that are not constrained to be zero having q elements (i.e. $\mathcal{S} = \{(i_1, j_1), \cdots, (i_k, j_k), \cdots, (i_q, j_q)\}$ with $1 \leq i_k \leq j_k \leq K$ for $k = 1, \cdots, q$). Define two $K \times q$ matrices

$$\mathbf{E}_1 = \begin{pmatrix} \mathbf{e}_{i_1} & \mathbf{e}_{i_2} & \cdots & \mathbf{e}_{i_q} \end{pmatrix} \quad \text{and} \quad \mathbf{E}_2 = \begin{pmatrix} \mathbf{e}_{j_1} & \mathbf{e}_{j_2} & \cdots & \mathbf{e}_{j_q} \end{pmatrix},$$

where $\mathbf{e}_{i_k}$ is a vector of zeros except the i_k-th entry being one for $k = 1, \cdots, q$. We express the optimization problem (3.2) as an unconstrained problem, following Dahl et al. (2005), with objective function $l(\boldsymbol{\beta}, \boldsymbol{\omega})$ given by

$$\begin{aligned} l(\boldsymbol{\beta}, \boldsymbol{\omega}) &= \frac{T}{2} \log\det \boldsymbol{\Theta}(\boldsymbol{\omega}) - \frac{1}{2} \operatorname{trace}\left[(\mathbf{Y} - \mathbf{BZ})^\top \boldsymbol{\Theta}(\boldsymbol{\omega}) (\mathbf{Y} - \mathbf{BZ}) \right] \\ &= \frac{T}{2} \log\det \boldsymbol{\Theta}(\boldsymbol{\omega}) - \frac{1}{2} \left[\mathbf{y} - \left(\mathbf{Z}^\top \otimes \mathbf{I}_K \right) \boldsymbol{\beta} \right]^\top \left[\mathbf{I}_T \otimes \boldsymbol{\Theta}(\boldsymbol{\omega}) \right] \left[\mathbf{y} - \left(\mathbf{Z}^\top \otimes \mathbf{I}_K \right) \boldsymbol{\beta} \right]. \end{aligned}$$

Here, the constant term is omitted and the inverse of innovation covariance matrix $\boldsymbol{\Theta}$ is parameterized as

$$\boldsymbol{\Theta}(\boldsymbol{\omega}) = \mathbf{E}_1 \mathbf{diag}\left(\boldsymbol{\omega}\right) \mathbf{E}_2^\top + \mathbf{E}_2 \mathbf{diag}\left(\boldsymbol{\omega}\right) \mathbf{E}_1^\top,$$

where $\boldsymbol{\omega} \in \mathbb{R}^q$ contains the non-zero element in the strict lower triangular part of $\boldsymbol{\Theta}$, and the non-zero elements on the diagonal are divided by 2, i.e.

$$\omega_k = \begin{cases} \theta_{i_k j_k}, & i_k \neq j_k \\ \dfrac{1}{2}\theta_{i_k j_k}, & i_k = j_k \end{cases} \quad \text{for } k = 1, \cdots, q.$$

We now prove that the feasible set $\mathcal{D}$ is biconvex based on the definition in Gorski et al. (2007). Let $\mathcal{B} = \{\boldsymbol{\beta} \in \mathbb{R}^{K(Kp+1)} \,|\, \mathbf{C}\boldsymbol{\beta} = \mathbf{0}\} \subseteq \mathbb{R}^{K(Kp+1)}$ and $\mathcal{W} = \{\boldsymbol{\omega} \in \mathbb{R}^q \,|\, \boldsymbol{\Theta}(\boldsymbol{\omega}) \succ 0\} \subseteq \mathbb{R}^q$ which are non-empty and convex. Let $\mathcal{D} = \{(\boldsymbol{\beta}, \boldsymbol{\omega}) \in \mathbb{R}^{K(Kp+1)} \times \mathbb{R}^q \,|\, \mathbf{C}\boldsymbol{\beta} = \mathbf{0},\ \boldsymbol{\Theta}(\boldsymbol{\omega}) \succ 0\} \subseteq \mathcal{B} \times \mathcal{W}$. Define $\mathcal{D}_{\boldsymbol{\omega}} = \{\boldsymbol{\beta} \in \mathcal{B} \,|\, (\boldsymbol{\beta}, \boldsymbol{\omega}) \in \mathcal{D}\}$ and $\mathcal{D}_{\boldsymbol{\beta}} = \{\boldsymbol{\omega} \in \mathcal{W} \,|\, (\boldsymbol{\beta}, \boldsymbol{\omega}) \in \mathcal{D}\}$.

For any $\boldsymbol{\beta}_1, \boldsymbol{\beta}_2 \in \mathcal{D}_{\boldsymbol{\omega}}$ and $\lambda \in [0, 1]$, $\lambda\mathbf{C}\boldsymbol{\beta}_1 + (1-\lambda)\mathbf{C}\boldsymbol{\beta}_2 \in \mathcal{D}_{\boldsymbol{\omega}}$ for every $\boldsymbol{\omega} \in \mathcal{W}$, since

$$\mathbf{C}\left[\lambda\boldsymbol{\beta}_1 + (1-\lambda)\boldsymbol{\beta}_2\right] = \lambda\mathbf{C}\boldsymbol{\beta}_1 + (1-\lambda)\mathbf{C}\boldsymbol{\beta}_2 = 0.$$

Therefore, $\mathcal{D}_{\boldsymbol{\omega}}$ is convex for every $\boldsymbol{\omega} \in \mathcal{W}$.

For any $\boldsymbol{\omega}_1, \boldsymbol{\omega}_2 \in \mathcal{D}_{\boldsymbol{\beta}}$ and $\lambda \in [0,1]$, $\lambda\boldsymbol{\omega}_1 + (1-\lambda)\boldsymbol{\omega}_2 \in \mathcal{D}_{\boldsymbol{\beta}}$ for every $\boldsymbol{\beta} \in \mathcal{B}$, since

$$\boldsymbol{\Theta}\left[\lambda\boldsymbol{\omega}_1 + (1-\lambda)\boldsymbol{\omega}_2\right] = \lambda\boldsymbol{\Theta}(\boldsymbol{\omega}_1) + (1-\lambda)\boldsymbol{\Theta}(\boldsymbol{\omega}_2) \succ 0.$$

Therefore, $\mathcal{D}_{\boldsymbol{\beta}}$ is convex for every $\boldsymbol{\beta} \in \mathcal{B}$. Hence, the set $\mathcal{D} \subseteq \mathcal{B} \times \mathcal{W}$ is biconvex.

We next show that the objective function is biconcave. Define $l_{\boldsymbol{\omega}}(\cdot) = l(\cdot, \boldsymbol{\omega})\colon \mathcal{D}_{\boldsymbol{\omega}} \to \mathbb{R}$ and $l_{\boldsymbol{\beta}}(\cdot) = f(\boldsymbol{\beta}, \cdot)\colon \mathcal{D}_{\boldsymbol{\beta}} \to \mathbb{R}$. For every fixed $\boldsymbol{\omega} \in \mathcal{W}$,

$$\begin{aligned} l_{\boldsymbol{\omega}}(\boldsymbol{\beta}) &= \frac{T}{2}\log\det\boldsymbol{\Theta}(\boldsymbol{\omega}) - \frac{1}{2}\left[\mathbf{y} - \left(\mathbf{Z}^\top \otimes \mathbf{I}_K\right)\boldsymbol{\beta}\right]^\top \left[\mathbf{I}_T \otimes \boldsymbol{\Theta}(\boldsymbol{\omega})\right]\left[\mathbf{y} - \left(\mathbf{Z}^\top \otimes \mathbf{I}_K\right)\boldsymbol{\beta}\right] \\ &= -\frac{1}{2}\boldsymbol{\beta}^\top\left[\mathbf{Z}\mathbf{Z}^\top \otimes \boldsymbol{\Theta}(\boldsymbol{\omega})\right]\boldsymbol{\beta} + \boldsymbol{\beta}^\top\left[\mathbf{Z} \otimes \boldsymbol{\Theta}(\boldsymbol{\omega})\right]\mathbf{y} - \frac{1}{2}\mathbf{y}^\top\left[\mathbf{I}_T \otimes \boldsymbol{\Theta}(\boldsymbol{\omega})\right]\mathbf{y} \\ &\quad + \frac{T}{2}\log\det\boldsymbol{\Theta}(\boldsymbol{\omega}) \end{aligned}$$

is a quadratic function of $\boldsymbol{\beta}$ and is strictly concave since $\mathbf{Z}\mathbf{Z}^\top \otimes \boldsymbol{\Theta}(\boldsymbol{\omega}) \succ 0$. Therefore, $l_{\boldsymbol{\omega}}(\cdot)$ is a concave function on $\mathcal{D}_{\boldsymbol{\omega}}$ for every fixed $\boldsymbol{\omega} \in W$.

Denote $\mathbf{S} = (\mathbf{Y} - \mathbf{B}\mathbf{Z})(\mathbf{Y} - \mathbf{B}\mathbf{Z})^\top$. For all $\boldsymbol{\omega}_1, \boldsymbol{\omega}_2 \in \mathcal{D}_{\boldsymbol{\beta}}$ with $\boldsymbol{\omega}_1 \neq \boldsymbol{\omega}_2$, $\lambda \in (0,1)$ and for every fixed $\boldsymbol{\beta} \in \mathcal{B}$,

$$\begin{aligned} l_{\boldsymbol{\beta}}(\lambda\boldsymbol{\omega}_1 + (1-\lambda)\boldsymbol{\omega}_2) &= \frac{T}{2}\log\det\boldsymbol{\Theta}(\lambda\boldsymbol{\omega}_1 + (1-\lambda)\boldsymbol{\omega}_2) \\ &\quad - \frac{1}{2}\operatorname{trace}\left[\mathbf{S}\boldsymbol{\Theta}(\lambda\boldsymbol{\omega}_1 + (1-\lambda)\boldsymbol{\omega}_2)\right] \\ &= \frac{T}{2}\log\det\left[\lambda\boldsymbol{\Theta}(\boldsymbol{\omega}_1) + (1-\lambda)\boldsymbol{\Theta}(\boldsymbol{\omega}_2)\right] \\ &\quad - \frac{1}{2}\left\{\lambda\operatorname{trace}\left[\mathbf{S}\boldsymbol{\Theta}(\boldsymbol{\omega}_1)\right] + (1-\lambda)\operatorname{trace}\left[\mathbf{S}\boldsymbol{\Theta}(\boldsymbol{\omega}_2)\right]\right\} \\ &> \frac{T}{2}\log\left\{\left[\det\boldsymbol{\Theta}(\boldsymbol{\omega}_1)\right]^\lambda\left[\det\boldsymbol{\Theta}(\boldsymbol{\omega}_2)\right]^{1-\lambda}\right\} \\ &\quad - \frac{1}{2}\left\{\lambda\operatorname{trace}\left[\mathbf{S}\boldsymbol{\Theta}(\boldsymbol{\omega}_1)\right] + (1-\lambda)\operatorname{trace}\left[\mathbf{S}\boldsymbol{\Theta}(\boldsymbol{\omega}_2)\right]\right\} \\ &= \lambda l_{\boldsymbol{\beta}}(\boldsymbol{\omega}_1) + (1-\lambda)l_{\boldsymbol{\beta}}(\boldsymbol{\omega}_2). \end{aligned}$$

Here, the inequality follows from Lemma A.1. Therefore, $l_{\boldsymbol{\beta}}(\cdot)$ is a strictly concave function on $\mathcal{D}_{\boldsymbol{\beta}}$ for every fixed $\boldsymbol{\beta} \in \mathcal{B}$. Hence, the optimization problem in (3.2) is biconcave. □

Recall the log-likelihood function (4.7) is

$$l(\mathbf{B}, \boldsymbol{\Theta}) = -\frac{KT}{2} \log 2\pi + \frac{T}{2} \log\det \boldsymbol{\Theta} - \frac{1}{2} \operatorname{trace}\left[(\mathbf{Y} - \mathbf{BZ})^\top \boldsymbol{\Theta} (\mathbf{Y} - \mathbf{BZ}) \right].$$

The gradient and the Hessian of the log-likelihood functions are

$$\nabla l(\boldsymbol{\beta}, \boldsymbol{\theta}) = \begin{pmatrix} \mathbf{vec}\left[\boldsymbol{\Theta}(\mathbf{Y} - \mathbf{BZ})\mathbf{Z}^\top \right] \\ \frac{1}{2}\mathbf{D}_K^\top \mathbf{vec}\left[T\boldsymbol{\Theta}^{-1} - (\mathbf{Y} - \mathbf{BZ})(\mathbf{Y} - \mathbf{BZ})^\top \right] \end{pmatrix} \text{ and}$$

$$\nabla^2 l(\boldsymbol{\beta}, \boldsymbol{\theta}) = \begin{pmatrix} \frac{\mathbf{ZZ}^\top}{T} \otimes \boldsymbol{\Theta} & -\left[\frac{2}{T}\mathbf{Z}(\mathbf{Y} - \mathbf{BZ})^\top \otimes \mathbf{I}_K \right] \mathbf{D}_K \\ -\mathbf{D}_K^\top \left[\frac{2}{T}(\mathbf{Y} - \mathbf{BZ})\mathbf{Z}^\top \otimes \mathbf{I}_K \right] & \frac{1}{2}\mathbf{D}_K^\top \left(\boldsymbol{\Theta}^{-1} \otimes \boldsymbol{\Theta}^{-1} \right) \mathbf{D}_K \end{pmatrix},$$

respectively, where $\boldsymbol{\beta} = \mathbf{vec}(\mathbf{B})$ and $\boldsymbol{\theta} = \mathbf{vech}(\boldsymbol{\Theta})$.

Proof. By matrix calculus, using the notation in Magnus & Neudecker (1999), we calculate the 1-st order differential of the log-likelihood function $l(\mathbf{B}, \boldsymbol{\Theta})$:

$$\begin{aligned} \mathsf{d}l(\mathbf{B}, \boldsymbol{\Theta}) &= \frac{T}{2} \operatorname{trace}\left(\boldsymbol{\Theta}^{-1} \mathsf{d}\boldsymbol{\Theta} \right) - \frac{1}{2} \operatorname{trace}\left(\mathbf{W} \mathsf{d}\boldsymbol{\Theta} \right) - \frac{1}{2} \operatorname{trace}\left(\boldsymbol{\Theta} \mathsf{d}\mathbf{W} \right) \\ &= \frac{1}{2} \operatorname{trace}\left[(T\boldsymbol{\Theta} - \mathbf{W}) \mathsf{d}\boldsymbol{\Theta} \right] - \frac{1}{2} \operatorname{trace}\left[\boldsymbol{\Theta}(-2(\mathsf{d}\mathbf{B})\mathbf{Z}(\mathbf{Y} - \mathbf{BZ})^\top) \right] \\ &= \frac{1}{2} \operatorname{trace}\left[(T\boldsymbol{\Theta}^{-1} - \mathbf{W}) \mathsf{d}\boldsymbol{\Theta} \right] + \operatorname{trace}\left[\mathbf{Z}(\mathbf{Y} - \mathbf{BZ})^\top \boldsymbol{\Theta} \mathsf{d}\mathbf{B} \right], \end{aligned} \tag{A.1}$$

where $\mathbf{W} = (\mathbf{Y} - \mathbf{BZ})(\mathbf{Y} - \mathbf{BZ})^\top$. By vectorization,

$$\begin{aligned}\mathsf{d}l(\boldsymbol{\beta}, \boldsymbol{\theta}) &= \frac{1}{2}\left[\mathbf{vec}\left(T\boldsymbol{\Theta}^{-1} - \mathbf{W}^\top\right)\right]^\top \mathsf{d}\,\mathbf{vec}\,(\boldsymbol{\Theta}) + \left[\mathbf{vec}\left(\boldsymbol{\Theta}(\mathbf{Y} - \mathbf{BZ})\mathbf{Z}^\top\right)\right]^\top \mathsf{d}\boldsymbol{\beta} \\ &= \frac{1}{2}\left[\mathbf{D}_K^\top \mathbf{vec}\left(T\boldsymbol{\Theta}^{-1} - \mathbf{W}^\top\right)\right]^\top \mathsf{d}\boldsymbol{\theta} + \left[\mathbf{vec}\left(\boldsymbol{\Theta}(\mathbf{Y} - \mathbf{BZ})\mathbf{Z}^\top\right)\right]^\top \mathsf{d}\boldsymbol{\beta},\end{aligned} \tag{A.2}$$

where $\mathbf{D}_K$ is a duplication matrix of dimension K. Therefore, the gradient is

$$\nabla l\,(\boldsymbol{\beta}, \boldsymbol{\theta}) = \begin{pmatrix} \mathbf{vec}\left[\boldsymbol{\Theta}(\mathbf{Y} - \mathbf{BZ})\mathbf{Z}^\top\right] \\ \frac{1}{2}\mathbf{D}_K^\top \mathbf{vec}\left[T\boldsymbol{\Theta}^{-1} - (\mathbf{Y} - \mathbf{BZ})(\mathbf{Y} - \mathbf{BZ})^\top\right] \end{pmatrix}. \tag{A.3}$$

We then compute the 2-nd order differential of the log-likelihood function $l(\mathbf{B}, \boldsymbol{\Theta})$:

$$\begin{aligned}\mathsf{d}^2 l(\mathbf{B}, \boldsymbol{\Theta}) &= \frac{1}{2}\operatorname{trace}\left[(T\mathsf{d}\boldsymbol{\Theta}^{-1} - \mathsf{d}\mathbf{W})\mathsf{d}\boldsymbol{\Theta}\right] + \operatorname{trace}\left[-\mathbf{Z}\mathbf{Z}^\top(\mathsf{d}\mathbf{B})^\top\boldsymbol{\Theta}\mathsf{d}\mathbf{B}\right] \\ &\qquad + \operatorname{trace}\left[\mathbf{Z}(\mathbf{Y} - \mathbf{BZ})^\top(\mathsf{d}\boldsymbol{\Theta})\mathsf{d}\mathbf{B}\right] \\ &= \frac{1}{2}\operatorname{trace}\left[-T\boldsymbol{\Theta}^{-1}(\mathsf{d}\boldsymbol{\Theta})\boldsymbol{\Theta}^{-1}\mathsf{d}\boldsymbol{\Theta} + 2(\mathsf{d}\mathbf{B})\mathbf{Z}(\mathbf{Y} - \mathbf{BZ})^\top\mathsf{d}\boldsymbol{\Theta}\right] \\ &\qquad - \operatorname{trace}\left[\mathbf{Z}\mathbf{Z}^\top(\mathsf{d}\mathbf{B})^\top\boldsymbol{\Theta}\mathsf{d}\mathbf{B}\right] + \operatorname{trace}\left[(\mathsf{d}\mathbf{B})\mathbf{Z}(\mathbf{Y} - \mathbf{BZ})^\top\mathsf{d}\boldsymbol{\Theta}\right] \\ &= -\frac{T}{2}\operatorname{trace}\left[\boldsymbol{\Theta}^{-1}(\mathsf{d}\boldsymbol{\Theta})\boldsymbol{\Theta}^{-1}\mathsf{d}\boldsymbol{\Theta}\right] + 2\operatorname{trace}\left[\mathbf{I}_K(\mathsf{d}\mathbf{B})\mathbf{Z}(\mathbf{Y} - \mathbf{BZ})^\top\mathsf{d}\boldsymbol{\Theta}\right] \\ &\qquad - \operatorname{trace}\left[\mathbf{Z}\mathbf{Z}^\top(\mathsf{d}\mathbf{B})^\top\boldsymbol{\Theta}\mathsf{d}\mathbf{B}\right].\end{aligned} \tag{A.4}$$

Thus, the Hessian matrix is

$$\nabla^2 l(\boldsymbol{\beta}, \boldsymbol{\theta}) = -T\begin{pmatrix} \frac{\mathbf{Z}\mathbf{Z}^\top}{T} \otimes \boldsymbol{\Theta} & -\left[\frac{2}{T}\mathbf{Z}(\mathbf{Y} - \mathbf{BZ})^\top \otimes \mathbf{I}_K\right]\mathbf{D}_K \\ -\mathbf{D}_K^\top\left[\frac{2}{T}(\mathbf{Y} - \mathbf{BZ})\mathbf{Z}^\top \otimes \mathbf{I}_K\right] & \frac{1}{2}\mathbf{D}_K^\top\left(\boldsymbol{\Theta}^{-1} \otimes \boldsymbol{\Theta}^{-1}\right)\mathbf{D}_K \end{pmatrix}. \tag{A.5}$$

□

Appendix B

Tables

TABLE B.1: Simulation results for Model 1 using the penalized likelihood estimation over 500 replications. Zeros_C (Zeros_I) is the average number of zero coefficients correctly (incorrectly) estimated to be zero. Standard errors are in the parentheses.

			$\hat{\mathbf{A}}$					$\hat{\mathbf{\Sigma}}_u^{-1}$				
Size	Penalty	Divergence	Bias	Variance	MSE	Zeros_C	Zeros_I	Bias	Variance	MSE	Zeros_C	Zeros_I
100	L1	0.2243 (0.0038)	0.7113	0.0472	0.1299	26.4720 (0.1037)	0.0820 (0.0123)	0.8352	0.1782	0.2445	7.5680 (0.0816)	0.0500 (0.0102)
	L1-MRCE	0.2225 (0.0037)	0.6592	0.0525	0.1158	24.7440 (0.1152)	0.0200 (0.0063)	0.7276	0.1830	0.2398	7.7140 (0.0754)	0.0540 (0.0109)
	MCP	0.1706 (0.0036)	0.2336	0.0511	0.0668	28.3800 (0.0723)	0.1660 (0.0167)	0.4356	0.2788	0.3010	9.3720 (0.0373)	0.2860 (0.0253)
	SCAD	0.1931 (0.0037)	0.4242	0.0527	0.0907	26.9780 (0.0910)	0.0920 (0.0129)	0.3396	0.2854	0.2982	8.9300 (0.0590)	0.0940 (0.0148)
200	L1	0.1078 (0.0017)	0.4936	0.0199	0.0597	26.9020 (0.0908)	0.0020 (0.0020)	0.7516	0.0840	0.1336	7.5740 (0.0772)	0 (0)
	L1-MRCE	0.0988 (0.0014)	0.4338	0.0218	0.0478	24.6420 (0.0996)	0 (0)	0.6708	0.0848	0.1281	7.7860 (0.0732)	0 (0)
	MCP	0.0609 (0.0011)	0.0916	0.0186	0.0212	28.9900 (0.0537)	0.0060 (0.0035)	0.2162	0.1001	0.1054	9.6900 (0.0263)	0.0140 (0.0053)
	SCAD	0.0738 (0.0013)	0.2229	0.0214	0.0344	27.9180 (0.0727)	0.0020 (0.0020)	0.1511	0.1031	0.1060	9.5240 (0.0425)	0.0020 (0.0020)
500	L1	0.0449 (0.0006)	0.3227	0.0074	0.0255	27.3160 (0.0805)	0 (0)	0.5810	0.0338	0.0642	7.6620 (0.0674)	0 (0)
	L1-MRCE	0.0392 (0.0005)	0.2754	0.0079	0.0189	24.9020 (0.0925)	0 (0)	0.5432	0.0341	0.0616	7.8640 (0.0657)	0 (0)
	MCP	0.0205 (0.0004)	0.0233	0.0062	0.0064	29.5280 (0.0339)	0 (0)	0.0923	0.0365	0.0375	9.8260 (0.0188)	0 (0)
	SCAD	0.0244 (0.0004)	0.1009	0.0072	0.0113	28.8940 (0.0484)	0 (0)	0.0729	0.0350	0.0357	9.8940 (0.0191)	0 (0)
1000	L1	0.0232 (0.0003)	0.2299	0.0036	0.0129	27.5160 (0.0765)	0 (0)	0.4360	0.0178	0.0349	7.6980 (0.0656)	0 (0)
	L1-MRCE	0.0195 (0.0003)	0.1841	0.0041	0.0087	24.2700 (0.0953)	0 (0)	0.4255	0.0176	0.0344	8.0540 (0.0620)	0 (0)
	MCP	0.0095 (0.0002)	0.0089	0.0027	0.0027	29.3420 (0.0384)	0 (0)	0.0456	0.0171	0.0173	9.8940 (0.0154)	0 (0)
	SCAD	0.0105 (0.0002)	0.0450	0.0033	0.0040	29.2580 (0.0405)	0 (0)	0.0376	0.0167	0.0169	9.9620 (0.0090)	0 (0)

TABLE B.2: Simulation results for Model 2 using the penalized likelihood estimation over 500 replications. Zeros_C (Zeros_I) is the average number of zero coefficients correctly (incorrectly) estimated to be zero. Standard errors are in the parentheses.

			$\hat{\mathbf{A}}$					$\hat{\mathbf{\Sigma}}_u^{-1}$				
Size	Penalty	Divergence	Bias	Variance	MSE	Zeros_C	Zeros_I	Bias	Variance	MSE	Zeros_C	Zeros_I
100	L1	0.2475 (0.0028)	1.1222	0.1023	0.1865	13.4620 (0.1156)	0.7720 (0.0385)	0.5863	0.2148	0.2359	4.8560 (0.0748)	0 (0)
	L1-MRCE	0.2460 (0.0027)	1.1057	0.1052	0.1853	13.2920 (0.1227)	0.7180 (0.0360)	0.4701	0.2304	0.2463	4.9420 (0.0740)	0 (0)
	MCP	0.2338 (0.0029)	0.6881	0.1298	0.1790	16.4640 (0.0857)	1.4320 (0.0455)	0.7139	0.2732	0.3221	8.4620 (0.0384)	0.0220 (0.0066)
	SCAD	0.2442 (0.0030)	0.8479	0.1300	0.1935	14.7100 (0.1014)	1.1140 (0.0421)	0.6147	0.2789	0.3161	7.9900 (0.0527)	0.0140 (0.0053)
200	L1	0.1299 (0.0016)	0.8309	0.0487	0.0951	13.8200 (0.1135)	0.1500 (0.0174)	0.6595	0.1017	0.1277	4.8580 (0.0777)	0 (0)
	L1-MRCE	0.1260 (0.0015)	0.7952	0.0487	0.0908	13.6220 (0.1074)	0.1080 (0.0139)	0.5657	0.1032	0.1221	4.8420 (0.0757)	0 (0)
	MCP	0.1064 (0.0015)	0.4097	0.0603	0.0802	17.0900 (0.0737)	0.6140 (0.0358)	0.3081	0.1081	0.1173	8.6560 (0.0322)	0 (0)
	SCAD	0.1147 (0.0016)	0.5768	0.0583	0.0943	15.2640 (0.0933)	0.3880 (0.0278)	0.2360	0.1110	0.1168	8.5860 (0.0395)	0 (0)
500	L1	0.0530 (0.0006)	0.5298	0.0179	0.0368	14.2360 (0.1063)	0 (0)	0.5790	0.0371	0.0583	5.0440 (0.0651)	0 (0)
	L1-MRCE	0.0513 (0.0005)	0.5085	0.0181	0.0355	14.1020 (0.1021)	0 (0)	0.5076	0.0385	0.0544	5.0960 (0.0642)	0 (0)
	MCP	0.0354 (0.0005)	0.1371	0.0181	0.0201	18.2160 (0.0578)	0.0320 (0.0088)	0.1320	0.0376	0.0393	8.8660 (0.0168)	0 (0)
	SCAD	0.0395 (0.0005)	0.2501	0.0214	0.0288	15.9700 (0.0815)	0.0260 (0.0071)	0.1129	0.0379	0.0392	8.9080 (0.0157)	0 (0)
1000	L1	0.0283 (0.0003)	0.3977	0.0086	0.0195	14.6520 (0.0967)	0 (0)	0.4737	0.0183	0.0330	5.0320 (0.0664)	0 (0)
	L1-MRCE	0.0268 (0.0003)	0.3730	0.0086	0.0181	14.2720 (0.0944)	0 (0)	0.4326	0.0184	0.0305	5.1440 (0.0662)	0 (0)
	MCP	0.0162 (0.0002)	0.0570	0.0071	0.0074	18.7220 (0.0484)	0 (0)	0.0541	0.0172	0.0176	8.8920 (0.0172)	0 (0)
	SCAD	0.0181 (0.0002)	0.1439	0.0085	0.0106	17.1100 (0.0706)	0 (0)	0.0448	0.0175	0.0178	8.9660 (0.0086)	0 (0)

TABLE B.3: Simulation results for Model 4 using the penalized likelihood estimation over 500 replications. Zeros_C (Zeros_I) is the average number of zero coefficients correctly (incorrectly) estimated to be zero. Standard errors are in the parentheses.

			$\hat{\mathbf{A}}$					$\hat{\mathbf{\Sigma}}_u^{-1}$				
Size	Penalty	Divergence	Bias	Variance	MSE	Zeros_C	Zeros_I	Bias	Variance	MSE	Zeros_C	Zeros_I
100	L1	0.2544 (0.0027)	1.6183	0.1029	0.2766	8.9060 (0.1047)	1.9120 (0.0413)	0.5356	0.2109	0.2298	4.9040 (0.0782)	0 (0)
	L1-MRCE	0.2527 (0.0027)	1.6013	0.1014	0.2727	8.9280 (0.1090)	1.9040 (0.0412)	0.4596	0.2248	0.2401	4.9720 (0.0759)	0 (0)
	MCP	0.2243 (0.0029)	0.7324	0.0858	0.1484	14.8200 (0.0753)	2.5440 (0.0353)	0.6894	0.2680	0.3160	8.4680 (0.0393)	0.0300 (0.0081)
	SCAD	0.2329 (0.0028)	0.9085	0.0929	0.1699	12.0960 (0.0923)	2.2000 (0.0376)	0.6545	0.2820	0.3256	7.9700 (0.0549)	0.0040 (0.0028)
200	L1	0.1401 (0.0015)	1.2950	0.0506	0.1703	9.2500 (0.1049)	1.8740 (0.0417)	0.6354	0.0962	0.1213	5.1300 (0.0740)	0 (0)
	L1-MRCE	0.1377 (0.0014)	1.2741	0.0503	0.1667	9.1780 (0.1056)	1.8520 (0.0407)	0.5381	0.1017	0.1193	5.1400 (0.0762)	0 (0)
	MCP	0.1080 (0.0013)	0.5355	0.0425	0.0844	15.6240 (0.0684)	2.3500 (0.0338)	0.2948	0.1045	0.1131	8.7240 (0.0270)	0 (0)
	SCAD	0.1164 (0.0014)	0.7156	0.0443	0.1033	13.4640 (0.0863)	2.2320 (0.0358)	0.2666	0.1086	0.1158	8.6260 (0.0369)	0 (0)
500	L1	0.0607 (0.0007)	0.8757	0.0241	0.0795	8.6480 (0.1004)	1.2700 (0.0382)	0.5393	0.0376	0.0568	5.1300 (0.0691)	0 (0)
	L1-MRCE	0.0591 (0.0007)	0.8602	0.0240	0.0775	8.6400 (0.0998)	1.2580 (0.0377)	0.4660	0.0380	0.0519	5.1660 (0.0693)	0 (0)
	MCP	0.0400 (0.0005)	0.3070	0.0176	0.0310	15.9760 (0.0651)	1.6140 (0.0312)	0.1247	0.0367	0.0382	8.8340 (0.0207)	0 (0)
	SCAD	0.0451 (0.0007)	0.4138	0.0208	0.0439	14.2040 (0.0812)	1.5760 (0.0369)	0.1041	0.0363	0.0375	8.9020 (0.0182)	0 (0)
1000	L1	0.0307 (0.0003)	0.6139	0.0118	0.0397	8.4400 (0.0990)	0.8640 (0.0303)	0.4787	0.0174	0.0327	5.0860 (0.0647)	0 (0)
	L1-MRCE	0.0302 (0.0003)	0.6153	0.0114	0.0396	8.5720 (0.0984)	0.8740 (0.0292)	0.4372	0.0179	0.0304	5.1920 (0.0646)	0 (0)
	MCP	0.0186 (0.0002)	0.1973	0.0072	0.0128	16.2580 (0.0523)	1.1700 (0.0266)	0.0538	0.0166	0.0169	8.9080 (0.0162)	0 (0)
	SCAD	0.0210 (0.0003)	0.2903	0.0083	0.0201	14.9520 (0.0711)	1.1460 (0.0272)	0.0370	0.0170	0.0172	8.9800 (0.0069)	0 (0)

TABLE B.4: Simulation results for Model 5 using the penalized likelihood estimation over 500 replications. Zeros_C (Zeros_I) is the average number of zero coefficients correctly (incorrectly) estimated to be zero. Standard errors are in the parentheses.

			$\hat{\mathbf{A}}$					$\hat{\mathbf{\Sigma}}_y^{-1}$				
Size	Penalty	Divergence	Bias	Variance	MSE	Zeros_C	Zeros_I	Bias	Variance	MSE	Zeros_C	Zeros_I
100	L1	0.6719 (0.0183)	4.6821	0.4800	1.0696	21.3420 (0.2521)	4.9060 (0.2168)	0.7824	0.3922	0.4465	4.2220 (0.1224)	0.6360 (0.0649)
	L1-MRCE	0.6509 (0.0166)	4.6499	0.4786	1.0548	21.4520 (0.2553)	4.9740 (0.2169)	0.8348	0.3881	0.4419	4.2780 (0.1172)	0.5860 (0.0614)
	MCP	0.5724 (0.0066)	2.5712	0.8554	1.0334	30.6000 (0.1215)	7.1600 (0.1367)	0.7134	0.4563	0.5177	8.0000 (0.0515)	0.8140 (0.0429)
	SCAD	0.5425 (0.0081)	2.8450	0.7382	0.9530	24.7340 (0.1703)	4.3160 (0.1390)	0.7547	0.4685	0.5382	7.0580 (0.0753)	0.4880 (0.0376)
200	L1	0.2570 (0.0028)	2.8538	0.2217	0.4364	20.1120 (0.1951)	0.5960 (0.0414)	0.5534	0.1256	0.1493	4.4420 (0.1047)	0 (0)
	L1-MRCE	0.2530 (0.0027)	2.8308	0.2201	0.4301	20.2640 (0.1920)	0.6020 (0.0420)	0.5074	0.1240	0.1455	4.5360 (0.1025)	0 (0)
	MCP	0.2200 (0.0026)	1.1313	0.3476	0.3884	31.7260 (0.1018)	2.1000 (0.0843)	0.4002	0.1278	0.1446	8.5120 (0.0365)	0.0580 (0.0116)
	SCAD	0.2205 (0.0022)	1.4357	0.3297	0.3887	26.4960 (0.1467)	0.9000 (0.0455)	0.3595	0.1360	0.1512	8.0900 (0.0542)	0.0200 (0.0063)
500	L1	0.1047 (0.0010)	1.8563	0.0819	0.1730	21.8260 (0.1907)	0.0020 (0.0020)	0.5340	0.0413	0.0600	5.2140 (0.0852)	0 (0)
	L1-MRCE	0.1039 (0.0010)	1.8405	0.0819	0.1712	21.9300 (0.1870)	0.0040 (0.0028)	0.5096	0.0422	0.0593	5.3100 (0.0865)	0 (0)
	MCP	0.0658 (0.0007)	0.3631	0.0926	0.0971	33.9960 (0.0719)	0.0820 (0.0129)	0.1737	0.0391	0.0421	8.8540 (0.0188)	0 (0)
	SCAD	0.0745 (0.0007)	0.6760	0.1038	0.1192	29.9660 (0.1206)	0.0160 (0.0063)	0.1464	0.0410	0.0434	8.8640 (0.0197)	0.0020 (0.0020)
1000	L1	0.0562 (0.0006)	1.4074	0.0396	0.0923	23.1960 (0.1962)	0 (0)	0.4633	0.0198	0.0343	5.5600 (0.0794)	0 (0)
	L1-MRCE	0.0551 (0.0006)	1.3890	0.0392	0.0905	23.2500 (0.1929)	0 (0)	0.4451	0.0196	0.0328	5.7180 (0.0772)	0 (0)
	MCP	0.0278 (0.0003)	0.1118	0.0362	0.0366	35.4420 (0.0394)	0 (0)	0.0752	0.0182	0.0188	8.9120 (0.0133)	0 (0)
	SCAD	0.0328 (0.0004)	0.3555	0.0442	0.0485	32.4480 (0.0905)	0 (0)	0.0684	0.0183	0.0189	8.9500 (0.0109)	0 (0)

Bibliography

Amblard, P.-O. (2015). A nonparametric efficient evaluation of partial directed coherence. *Biological Cybernetics*, *109*, 203–214.

Anderson, T. W. (2003). *An introduction to multivariate statistical analysis*. (3rd ed.). Hoboken, NJ: John Wiley and Sons.

Baccalá, L. A., & Sameshima, K. (2001). Partial directed coherence: a new concept in neural structure determination. *Biological Cybernetics*, *84*, 463–474.

Beltrão, K. I., & Bloomfield, P. (1987). Determining the bandwidth of a kernel spectrum estimate. *Journal of Time Series Analysis*, *8*, 21–38.

Box, G. E. P., & Cox, D. R. (1964). An analysis of transformations. *Journal of the Royal Statistical Society: Series B (Methodological)*, *26*, 211–252.

Boyd, S., Parikh, N., Chu, E., Peleato, B., & Eckstein, J. (2011). Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, *3*, 1–122.

Brillinger, D. R. (1981). *Time series: data analysis and theory*. (Expanded ed.). San Francisco, CA: Holden-Day.

Brillinger, D. R. (1996). Remarks concerning graphical models for time series and point processes. *Revista de Econometrica*, *16*, 1–23.

Brockwell, P. J., & Davis, R. A. (1991). *Time series: theory and methods*. (2nd ed.). New York, NY: Springer-Verlag.

Dahl, J., Roychowdhury, V., & Vandenberghe, L. (2005). Maximum likelihood estimation of Gaussian graphical models: Numerical implementation and topology selection. *UCLA preprint*, .

Dahl, J., Vandenberghe, L., & Roychowdhury, V. (2008). Covariance selection for nonchordal graphs via chordal embedding. *Optimization Methods and Software*, *23*, 501–520.

Dahlhaus, R. (2000). Graphical interaction models for multivariate time series. *Metrika*, *51*, 157–172.

Dahlhaus, R., Eichler, M., & Sandkühler, J. (1997). Identification of synaptic connections in neural ensembles by graphical models. *Journal of Neuroscience Methods*, *77*, 93–107.

Darroch, J. N., Lauritzen, S. L., & Speed, T. P. (1980). Markov fields and log-linear interaction models for contingency tables. *The Annals of Statistics*, *8*, 522–539.

d'Aspremont, A., Banerjee, O., & El Ghaoui, L. (2008). First-order methods for sparse covariance selection. *SIAM Journal on Matrix Analysis and Applications*, *30*, 56–66.

Davis, R. A., Zang, P., & Zheng, T. (2016). Sparse vector autoregressive modeling. *Journal of Computational and Graphical Statistics*, *25*, 1077–1096.

Dempster, A. P. (1972). Covariance selection. *Biometrics*, *28*, 157–175.

Edwards, D. (1995). *Introduction to graphical modelling*. New York, NY: Springer-Verlag.

Eichler, M. (2012). Graphical modelling of multivariate time series. *Probability Theory and Related Fields*, *153*, 233–268.

Fan, J., Feng, Y., & Wu, Y. (2009). Network exploration via the adaptive LASSO and SCAD penalties. *The Annals of Applied Statistics*, *3*, 521–541.

Fan, J., & Li, R. (2001). Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American Statistical Association*, *96*, 1348–1360.

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

Gorski, J., Pfeuffer, F., & Klamroth, K. (2007). Biconvex sets and optimization with biconvex functions: a survey and extensions. *Mathematical Methods of Operations Research*, *66*, 373–407.

Granger, C. W. J. (1969). Investigating causal relations by econometric models and cross-spectral methods. *Econometrica*, *37*, 424–438.

Hannan, E. J., & Quinn, B. G. (1979). The determination of the order of an autoregression. *Journal of the Royal Statistical Society. Series B (Methodological)*, *41*, 190–195.

Hastie, T., Tibshirani, R., & Wainwright, M. (2015). *Statistical learning with sparsity: the Lasso and generalizations*. Boca Raton, FL: CRC Press.

Hsu, N.-J., Hung, H.-L., & Chang, Y.-M. (2008). Subset selection for vector autoregressive processes using Lasso. *Computational Statistics and Data Analysis*, *52*, 3645–3657.

Hu, F., Lu, Z., Wong, H., & Yuen, T. P. (2016). Analysis of air quality time series of Hong Kong with graphical modeling. *Environmetrics*, *27*, 169–181.

Hunter, D. R., & Li, R. (2005). Variable selection using MM algorithms. *The Annals of Statistics*, *33*, 1617–1642.

Jung, A., Hannak, G., & Goertz, N. (2015). Graphical LASSO based model selection for time series. *IEEE Signal Processing Letters*, *22*, 1781–1785.

Koopmans, L. H. (1995). *The spectral analysis of time series*. (2nd ed.). San Diego, CA: Academic Press.

Lange, K. (2013). *Optimization*. (2nd ed.). New York, NY: Springer.

Lauritzen, S. L. (1996). *Graphical models*. Oxford, England: Clarendon Press.

Lauritzen, S. L., & Richardson, T. S. (2002). Chain graph models and their causal interpretations. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *64*, 321–348.

Lee, W., & Liu, Y. (2012). Simultaneous multiple response regression and inverse covariance matrix estimation via penalized Gaussian maximum likelihood. *Journal of Multivariate Analysis*, *111*, 241–255.

Li, L., & Toh, K.-C. (2010). An inexact interior point method for L_1-regularized sparse covariance selection. *Mathematical Programming Computation*, *2*, 291–315.

Lütkepohl, H. (2005). *New introduction to multiple time series analysis*. Berlin, Germany: Springer-Verlag.

Magnus, J. R., & Neudecker, H. (1999). *Matrix Differential Calculus with Applications in Statistics and Econometrics*. (Rev. ed.). Chichester, England: John Wiley and Sons.

Matsuda, Y., & Yajima, Y. (2004). On testing for separable correlations of multivariate time series. *Journal of Time Series Analysis*, *25*, 501–528.

McLeod, A. I., & Gweon, H. (2013). Optimal deseasonalization for monthly and daily geophysical time series. *Journal of Environmental Statistics*, *4*, 1–11.

Medkour, T., Walden, A. T., & Burgess, A. (2009). Graphical modelling for brain connectivity via partial coherence. *Journal of Neuroscience Methods*, *180*, 374–383.

Nicholson, W. B., Matteson, D. S., & Bien, J. (2017). VARX-L: Structured regularization for large vector autoregressions with exogenous variables. *International Journal of Forecasting*, *33*, 627–651.

Oxley, L., Reale, M., & Tunnicliffe-Wilson, G. (2004). Finding directed acyclic graphs for vector autoregressions. In J. Antoch (Ed.), *COMPSTAT 2004 - Proceedings in Computational Statistics* (pp. 1621–1628). Heidelberg, Germany: Physica-Verlag.

Pearl, J. (1995). Causal diagrams for empirical research. *Biometrika*, *82*, 669–688.

R Core Team (2017). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. URL: `https://www.R-project.org/`.

Ranganath, R., Tang, L., Charlin, L., & Blei, D. (2015). Deep exponential families. In L. Guy, & S. V. N. Vishwanathan (Eds.), *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics* (pp. 762–771). Proceedings of Machine Learning Research.

Ren, Y., Xiao, Z., & Zhang, X. (2013). Two-step adaptive model selection for vector autoregressive processes. *Journal of Multivariate Analysis*, *116*, 349–364.

Rothman, A. J., Bickel, P. J., Levina, E., & Zhu, J. (2008). Sparse permutation invariant covariance estimation. *Electronic Journal of Statistics*, *2*, 494–515.

Rothman, A. J., Levina, E., & Zhu, J. (2010). Sparse multivariate regression with covariance estimation. *Journal of Computational and Graphical Statistics*, *19*, 947–962.

Salakhutdinov, R., & Hinton, G. (2009). Deep Boltzmann machines. In D. David van, & W. Max (Eds.), *Proceedings of the Twelth International Conference on Artificial Intelligence and Statistics* (pp. 448–455). Proceedings of Machine Learning Research.

Schwarz, G. (1978). Estimating the dimension of a model. *The Annals of Statistics*, *6*, 461–464.

Sofer, T., Dicker, L., & Lin, X. (2014). Variable selection for high dimensional multivariate outcomes. *Statistica Sinica*, *24*, 1633–1654.

Song, S., & Bickel, P. J. (2011). Large vector auto regressions. *Arxiv preprint arXiv:1106.3915v1*, .

Songsiri, J. (2013). Sparse autoregressive model estimation for learning Granger causality in time series. In *Proceedings of the 38th IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 3198–3202).

Songsiri, J., Dahl, J., & Vandenberghe, L. (2009). Graphical models of autoregressive processes. In D. P. Palomar, & Y. C. Eldar (Eds.), *Convex Optimization in Signal Processing and Communications* (pp. 89–116). Cambridge University Press.

Sturm, J. F. (1999). Using SeDuMi 1.02, a Matlab toolbox for optimization over symmetric cones. *Optimization Methods and Software*, *11*, 625–653.

Tibshirani, R. (1996). Regression shrinkage and selection via the Lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, *58*, 267–288.

Tunnicliffe-Wilson, G., Reale, M., & Haywood, J. (2015). *Models for dependent time series*. Boca Raton, FL: CRC Press.

Tütüncü, R. H., Toh, K. C., & Todd, M. J. (2003). Solving semidefinite-quadratic-linear programs using SDPT3. *Mathematical Programming*, *95*, 189–217.

Valdés-Sosa, P. A., Sánchez-Bornot, J. M., Lage-Castellanos, A., Vega-Hernández, M., Bosch-Bayard, J., Melie-García, L., & Canales-Rodríguez, E. (2005). Estimating brain functional connectivity with sparse multivariate autoregression. *Philosophical Transactions of the Royal Society B: Biological Sciences*, *360*, 969–981.

Vandenberghe, L., Boyd, S., & Wu, S.-P. (1998). Determinant maximization with linear matrix inequality constraints. *SIAM Journal on Matrix Analysis Applications*, *19*, 499–533.

Wei, W. W. S. (2006). *Time Series Analysis: Univariate and Multivariate Methods*. (2nd ed.). Boston, MA: Pearson Education.

Wermuth, N. (1976). Model search among multiplicative models. *Biometrics*, *32*, 253–263.

Yuan, M., & Lin, Y. (2006). Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *68*, 49–67.

Zhang, C.-H. (2010). Nearly unbiased variable selection under minimax concave penalty. *The Annals of Statistics*, *38*, 894–942.

Zou, H., & Li, R. (2008). One-step sparse estimates in nonconcave penalized likelihood models. *The Annals of Statistics*, *36*, 1509–1533.