Title

Some contributions to uncertainty quantification and change point detection in dynamic systems

Author

HAN, YI

Publication Date

2024

Peer reviewed|Thesis/dissertation

Some Contributions to Uncertainty Quantification and Change Point Detection in Dynamic Systems

By

YI HAN

DISSERTATION

Submitted in partial satisfaction of the requirements for the degree of

DOCTOR OF PHILOSOPHY

in

Statistics

in the

OFFICE OF GRADUATE STUDIES

of the

UNIVERSITY OF CALIFORNIA

DAVIS

Approved:

Thomas C.M. Lee, Chair

Alexander Aue

Debashis Paul

Committee in Charge

2024

To my family for their endless love, support, and encouragement.

Contents

Abstract

Some contributions to uncertainty quantification and change point detection in dynamic systems

This dissertation makes significant contributions to important statistical machine learning problems, including uncertainty quantification and structural break detection in dynamic systems. It focuses on these two challenges in several specific settings and develops tailored solutions. Firstly, it addresses the problem of uncertainty quantification for line spectral estimation. By leveraging the generalized fiducial inference framework, a novel method is developed to quantify the uncertainty of spectral line estimators. This method is theoretically proven to possess desirable properties and demonstrates promising empirical performance. Additionally, the proposed method has been successfully applied to exoplanet detection applications, shedding light on a crucial topic in astronomy. Secondly, the dissertation tackles the challenge of breakpoint detection in non-stationary network vector autoregression models. Thirdly, it considers breakpoint detection in pairwise ranking problems. For the latter two problems, the minimum description length principle is invoked to derive a model selection criterion, which is shown to produce statistically consistent estimates for the number and locations of change points, as well as other model parameters. Furthermore, two practical algorithms are developed to optimize the criterion for these respective problems.

Acknowledgments

First and foremost, I would like to express my deepest gratitude to my advisor, Professor Thomas C. M. Lee, for his unwavering support, expert guidance, and invaluable mentorship throughout my doctoral studies. With his steadfast help, I have transitioned from a fresh graduate student with limited knowledge of academic research to an independent researcher capable of conducting basic research. Thomas not only provides valuable suggestions to refine my research approach but also offers unwavering support and compassion during times of confusion and frustration. His patience, kindness, and willingness to assist me in all aspects of my academic and personal life have been truly appreciated. Without the mentorship and encouragement of this exceptional scholar and supportive mentor, it would have been impossible for me to accomplish the completion of my PhD degree. I am eternally grateful for Thomas's significant role in my journey.

Besides my advisor, I am deeply grateful to Professor Alexander Aue, Professor Debashis Paul, Professor Ethan Anderes, and Professor Dalia Ghanem for serving on both my PhD dissertation defense committee and the qualifying exam committee. Their insightful feedback, constructive comments, and unwavering encouragement throughout my research have been instrumental in strengthening my thesis and shaping my growth as a PhD. I am truly honored to have benefited from their collective expertise and mentorship.

I am also deeply appreciative of the tireless efforts of our department staff - Pete Scully, Sara Driver, Olga Rodriguez, Andi Carr, Kimberly McMullen, and Nehad Ismail. Their unwavering commitment to providing an array of resources, from welcoming tea time snacks to state-of-the-art research facilities, has been invaluable in supporting me throughout my doctoral studies. Additionally, I fondly remember the annual New Year party hosted by our department, which always made me feel right at home within the academic community. The warmth and camaraderie fostered by these events will be greatly missed.

I would also thank my dear friends and classmates within the Department of Statistics. The memories we have built together over the past five years are truly unforgettable. Whether it was frantically working to meet deadlines, engaging in lively discussions about our research projects,

or gathering to enjoy leisure time. I am honored to have had the opportunity to share this journey with such a wonderful and supportive community of peers, and I will cherish these experiences long after the completion of my doctoral degree.

Last, but certainly not least, I would like to express my deepest gratitude to my family. To my parents, I am forever indebted for your unconditional love, support, and sacrifices that have enabled me to reach this milestone. Besides, Ranran, your constant encouragement and companionship have been invaluable. It is to this loving family that I proudly dedicate this dissertation, in the hopes of making you all proud.

CHAPTER 1

Introduction

In the analysis of many real-world complex problems, two key challenges are widely encountered. The first is quantifying the uncertainty of estimated parameters, and the second is detecting breakpoints in observed data. This dissertation focuses on addressing these two important problems.

Spectral analysis is a crucial topic that has garnered significant attention in the signal processing community due to its rich applications in areas such as speech coding, radar, sonar signal processing, and imaging systems. However, most published work in this field primarily focuses on point estimation of the spectral line numbers and frequencies, while relatively little attention has been given to the issue of uncertainty quantification. Chapter 2 of this dissertation examines the line spectral estimation problem from the perspective of uncertainty quantification for the numbers and amplitudes of the signals. Specifically, it develops a novel method for deriving a probability density function on the space of all potential signal combinations. This approach enables the calculation of point estimates and confidence intervals for quantities of interest. The proposed method is based on the relatively new methodology termed generalized fiducial inference (GFI), a modern update of Fisher's original fiducial idea, and is theoretically proven to possess desirable properties. Several simulated datasets and three real exoplanet datasets are used to illustrate the promising empirical performance of the proposed method.

Another critical problem addressed in this dissertation is the detection of breakpoints in observed multivariate time-series datasets. Breakpoints are time points that divide the series into smaller locally stationary time series, where the stationarity assumption holds.

In Chapter 2, the dissertation tackles the challenge of analyzing multivariate time series data in dynamic networks. It introduces a piecewise stationary network vector autoregressive (NAR)

model to capture evolving dynamics within the network over time. The identification of these segments, along with the determination of the NAR processes' autoregressive lag orders, are treated as unknowns. The minimum description length (MDL) principle is leveraged to develop a criterion for model selection that estimates these unknown parameters. A two-stage genetic algorithm is formulated to tackle this optimization challenge. The MDL criterion is proven to be consistent in identifying the number and positions of the breakpoints, and its validity is demonstrated through simulation studies and real data analysis related to yellow-cab pick-up data in Manhattan. To the best of our knowledge, this is one of the first works to consider breakpoint detection in NAR models.

In Chapter 3, the dissertation shifts its focus to breakpoint detection within the ranking problem domain. The Bradley-Terry-Luce model, which facilitates the estimation of item ranks based on pairwise comparison results, is foundational. Additionally, with the availability of covariate information for the items, the covariate-assisted ranking estimation (CARE) model was introduced to enhance ranking accuracy. Chapter 3 introduces the Piecewise Stationary CARE (PS-CARE) model, designed to partition the data into distinct stationary CARE phases, initially characterized by unknown numbers and positions of segments. To address this challenge, the minimum description length (MDL) principle is employed to derive a model selection criterion, crucial for accurately estimating these unknown parameters. The practical optimization of this criterion is achieved through the utilization of the PELT algorithm, effectively identifying change points—namely, the junctures where adjacent CARE segments converge temporally.

CHAPTER 2

Uncertainty Quantification for Sparse Estimation of Spectral Lines

Line spectral estimation is an important problem that finds many useful applications in signal processing. Many high-performance methods have been proposed for solving this problem: they select the number of spectral lines and provide point estimates of the frequencies and amplitudes of such spectral lines. This chapter studies the line spectral estimation problem from a different and equally important angle: uncertainty quantification. More precisely, this chapter develops a novel method that provides an uncertainty measure for the number of spectral lines and also offers point estimates and confidence intervals for other parameters of interest. The proposed method is based on the generalized fiducial inference framework and is shown to possess desirable theoretical and empirical properties. It has also been numerically compared with existing methods in the literature and applied for the detection of exoplanets.

2.1. Introduction

Spectral analysis is an important topic that attracts much attention in the signal processing community. It has rich applications in areas like speech coding [1], radar and sonar signal processing [2] [3], and imaging system [4], to name a few.

This chapter focuses on the sparse spectral line estimation problem as described, for example, in [5]. Let

$$\boldsymbol{Y} = [Y(t_1), Y(t_2), \cdots, Y(t_N)]^T \in \mathbb{C}^{N\times 1} \tag{2.1}$$

denote the complex-valued signal data vector, where the observed times $t_k \in \mathbb{R}^+$, $k \in 1, \ldots, N$, are not required to be regularly spaced. We shall focus on complex-valued signals, but our methodology can be naturally carried over to real-valued signals; see Section VI. We assume that $\boldsymbol{Y}$ satisfies the following model, sometimes known as the sinusoids-in-noise model [5], in which p

represents the true number of significant frequencies:

$$\boldsymbol{Y} = \sum_{l=1}^{p} \alpha_l \boldsymbol{a}(f_l) + \boldsymbol{\epsilon}, \tag{2.2}$$

where $\alpha_l \in \mathbb{C}$ are the complex amplitudes of the p sinusoidal components, $f_l \in \mathbb{R}$ are the true frequencies, and

$$\boldsymbol{a}(f) = [e^{i2\pi f t_1}, e^{i2\pi f t_2}, e^{i2\pi f t_3}, \cdots, e^{i2\pi f t_N}]^T \in \mathbb{C}^{N\times 1}.$$

Also, $\boldsymbol{\epsilon} \in \mathbb{C}^{N\times 1}$ is the noise vector, and we assume that its elements are i.i.d. and follow the complex normal distribution with mean 0 and variance σ^2, denoted as $\mathcal{CN}(0, \sigma^2)$.

With this setup, the problem is to use the observation signal vector $\boldsymbol{Y}$ to estimate the true frequencies f_l and their amplitudes $|\alpha_l|$. This problem has been studied for a long time, and different methods have been proposed. An earlier set of methods are non-parametric, including conventional periodogram-based methods and variants like the Daniell method [6] and the Welch method [7]. There are also correlogram-based, temporal windowing, and lag windowing methods. However, these methods may show low performance, such as limited resolving power.

The second set of methods is parametric and models the time series data with auto-regressive or auto-regressive moving-average processes [8] [9]. They provide accurate spectral estimation if the assumed model is appropriate for the observed time series. However, a drawback of these methods is that they typically require prior knowledge of the number of true frequencies p, which is often not practical.

The third set of methods is semi-parametric, which mostly performs sparse estimation. The performances of these methods are similar to those of parametric methods despite not requiring prior knowledge of p. Some of these methods perform sparse data recovery using mixed norm approximation [10], or atomic norm denoising [11]. Also, there are other sparse estimation methods that need other prior information, such as the noise variance as in [12]. One notable exception is the LIKES method (LiKelihood-based Estimation of Sparse), which does not require prior information [5].

Lastly, Bayesian methods have also been proposed [13] [14]. In addition to offering point estimates, the latter work also provides uncertainty quantification for some parameters of interest.

We need more notations to proceed. Assume $f_{\max}$ to be the upper bound of all the true frequencies $\{f_l\}$; i.e., $f_{\max} \geq f_l, l = 1, \ldots, p$. Let Δ be the step size or the distance between two adjacent grid points of a uniform grid covering the interval $[0, f_{\max}]$. This chapter only considers the positive frequencies for notation simplicity, but the discussion can be straightforwardly extended to negative frequencies. Finally, write

$$K = \lfloor \frac{f_{\max}}{\Delta} \rfloor \tag{2.3}$$

and

$$\boldsymbol{A} = [\boldsymbol{a}(0), \boldsymbol{a}(\Delta), \cdots, \boldsymbol{a}((K-1)\Delta))] \in \mathbb{C}^{N \times K}.$$

Using these notations, we can approximately re-express (2.2) as

$$\boldsymbol{Y} = \boldsymbol{A}\boldsymbol{\beta} + \boldsymbol{\epsilon}, \tag{2.4}$$

where $\boldsymbol{\beta} = [\beta_1, \cdots, \beta_K]^T$ is a sparse vector with mostly zero elements. Those non-zero elements of $\boldsymbol{\beta}$ equal to $\{\alpha_l\}$, while the indexes of these non-zero elements represent the corresponding frequencies in $\boldsymbol{A}$ that are equal to $\{f_l\}$.

The main idea of this so-called on-grid method is to use the grid that is closest to the true frequency to approximate it. Also, the problem of estimating $\{\alpha_l, f_l\}$ can be reformulated as estimating the sparse vector $\boldsymbol{\beta}$ in (2.4) and detecting the non-zero elements of the sparse vector. By choosing K to be sufficiently large and Δ to be sufficiently small, one can have the distance between the true frequencies and their closest grids be practically negligible. However, a very large value of K usually implies high-dimensional problems, so there is actually a trade-off between estimation accuracy and computational efficiency.

In practice, K is almost always much larger than N, which makes the estimation of $\boldsymbol{\beta}$ from $\boldsymbol{Y}$ in (2.4) a very challenging task. Different methods have been proposed to solve this problem,

where some require additional prior knowledge such as noise variance or the number of non-zero frequencies; e.g., see [12, 15, 16].

It is fair to say that most existing methods focus on estimating the amplitudes α_l and noise variance σ^2. At the same time, very little treatment has been given to the issue of uncertainty quantification. The main goal of this chapter is to construct confidence intervals for α_l, σ^2, as well as the number of the true frequencies, p. The proposed method is based on the relatively new methodology termed generalized fiducial inference (GFI) [17]. To the best of our knowledge, this is one of the first complete systematic analyses that capture these uncertainties in the line spectral estimation problem. It is also the first time that GFI is being applied to a complex-valued problem.

The rest of this chapter is organized as follows. Section 2.2 provides some background material and usage on GFI. Section 2.3 applies the methodology to the sparse line spectral estimation problem, and one relatively simple and fast algorithm to generate fiducial samples is proposed. The theoretical properties of the proposed solution are examined in Section 2.4, while its empirical properties are illustrated in Sections 2.5 and 2.6 by numerical simulations and a real data application. Lastly, concluding remarks are offered in Section 2.7, and technical details are provided in the appendix.

2.2. Methodology

2.2.1. A Brief History of Generalized Fiducial Inference. The idea of fiducial inference was first proposed by Fisher in 1930s [18] as an alternative to the Bayesian approach with the goal of constructing an appropriate statistical distribution on the estimator of an unknown parameter. One potential issue of the Bayesian approach is that, when inappropriate prior distributions are used, the performance and reliability of the approach could be affected. Fisher's fiducial method intends to avoid using the prior distribution; instead, it considers a switching mechanism between the model parameters and the observed data that is very similar to the idea of the method of maximum likelihood. In spite of Fisher's continuous endeavor to complete the framework of fiducial inference, it has not received much attention because it works well only for single-parameter problems but

fails in the context of multiple parameters. Interested readers are referred to [17], where a more detailed introduction about the history of fiducial inference is given.

In recent years, there has been a resurgent interest in reformulating the fiducial concept. These modifications include Dempster–Shafer theory [19], [20] and inferential models [21]. Motivated by generalized confidence intervals [22], [23] and the surrogate variable method for obtaining confidence intervals for variance components [24], GFI was developed in a series of papers published around 2010s, and summarized in [17]. It has been successfully applied to solve different uncertainty quantification problems, including wavelet regression [25], ultrahigh dimensional regression [26] and sparse additive models [27].

2.2.2. An Introduction to Generalized Fiducial Inference. As mentioned before, GFI utilizes the idea of a so-called switching principle that is similar to Fisher's celebrated maximum likelihood method. It first begins with expressing the relationship between the data $\boldsymbol{Y}$ and the parameter $\boldsymbol{\theta}$ with

$$\boldsymbol{Y} = \boldsymbol{G}(\boldsymbol{U}, \boldsymbol{\theta}), \tag{2.5}$$

where $\boldsymbol{G}(\cdot, \cdot)$ is sometimes known as the "structural equation." Also, $\boldsymbol{U}$ is the random component of the problem whose distribution is **completely known** and is independent of $\boldsymbol{\theta}$. For example, for the problem of estimating μ from $\{X_i\}_{i=1}^n$ with X_i's as i.i.d. $\mathcal{N}(\mu, \sigma^2)$, we write $X_i = \mu + \sigma Z_i$ with Z_i as i.i.d. $\mathcal{N}(0, 1)$, where the parameter $\boldsymbol{\theta} = \{\mu, \sigma\}$, data $\boldsymbol{Y} = \{X_i\}_{i=1}^n$ and random component $\boldsymbol{U} = \{Z_i\}_{i=1}^n$. Note that the distribution of $\boldsymbol{U}$ is completely known.

Similar to the main idea behind maximum likelihood estimation, with the switching principle, the roles of $\boldsymbol{\theta}$ and $\boldsymbol{Y}$ are switched in the GFI framework once the data are observed. That is, to treat the random data $\boldsymbol{Y}$ as deterministic and the deterministic parameter $\boldsymbol{\theta}$ as random. With this idea, we can define a set $\{\boldsymbol{\theta} : \boldsymbol{y} = \boldsymbol{G}(\boldsymbol{U}^*, \boldsymbol{\theta})\}$ as the inverse mapping of $\boldsymbol{G}$, where $\boldsymbol{U}^*$ is an independent copy of $\boldsymbol{U}$ and $\boldsymbol{y}$ is an observation of $\boldsymbol{Y}$. A method is provided by [28] to ensure the existence and uniqueness of this inverse mapping.

With the above setup, we can build a distribution of $\boldsymbol{\theta}$ from (2.5) in the following manner. For any observed data $\boldsymbol{y}$ and $\boldsymbol{u}$, we can adopt the method from [28] to identify one $\boldsymbol{\theta}$ that guarantees the existence of the inverse

$$\boldsymbol{H_y(u)} = \{\boldsymbol{\theta} : \boldsymbol{y} = \boldsymbol{G}(\boldsymbol{U}^*, \boldsymbol{\theta})\}. \tag{2.6}$$

Since the distribution of $\boldsymbol{U}$ is totally known and independent of $\boldsymbol{\theta}$, we can generate the random samples $\boldsymbol{U_1, U_2, \ldots}$ and use (2.6) to obtain the random samples for $\boldsymbol{\theta}$ via

$$\boldsymbol{\theta}_1 = \boldsymbol{H_y}(\boldsymbol{U}_1), \quad \boldsymbol{\theta}_2 = \boldsymbol{H_y}(\boldsymbol{U}_2), \ldots$$

In other words, GFI transfers the randomness in $\boldsymbol{U}_1, \boldsymbol{U}_2, \ldots$ to $\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \ldots$ via the inverse equation (2.6). We call these $\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, \ldots$ *fiducial samples*, which can be used to calculate point estimates and construct confidence intervals of $\boldsymbol{\theta}$ in a way similar to posterior samples in the Bayesian context. Notice that an explicit expression for $\boldsymbol{H_y}$ may not exist for certain problems, but next, we describe how the fiducial samples can still be generated without calculating an expression for $\boldsymbol{H_y}$.

Through (2.6) one can see that a density function $r(\boldsymbol{\theta}|\boldsymbol{y})$ is implicitly defined for $\boldsymbol{\theta}$. We refer $r(\boldsymbol{\theta}|\boldsymbol{y})$ as the *generalized fiducial density* (GFD) of $\boldsymbol{\theta}$, which plays a similar role as the posterior density in the Bayesian context. It is shown in [17] that under some mild smoothness assumptions on the likelihood function $f(\boldsymbol{y}, \boldsymbol{\theta})$ of $\boldsymbol{y}$, the GFD $r(\boldsymbol{\theta}|\boldsymbol{y})$ admits the following expression

$$r(\boldsymbol{\theta}|\boldsymbol{y}) = \frac{f(\boldsymbol{y}, \boldsymbol{\theta}) J(\boldsymbol{y}, \boldsymbol{\theta})}{\int_\Theta f\left(\boldsymbol{y}, \boldsymbol{\theta}'\right) J\left(\boldsymbol{y}, \boldsymbol{\theta}'\right) d\boldsymbol{\theta}'}, \tag{2.7}$$

where

$$J(\boldsymbol{y}, \boldsymbol{\theta}) = D\left(\frac{\boldsymbol{d}}{\boldsymbol{d\theta}} \boldsymbol{G}(\boldsymbol{U}, \boldsymbol{\theta})|_{\boldsymbol{U}=\boldsymbol{G}^{-1}(\boldsymbol{Y}, \boldsymbol{\theta})}\right) \tag{2.8}$$

with $D(\boldsymbol{A}) = |\det(\boldsymbol{A}^T \boldsymbol{A})|^{1/2}$ and $\boldsymbol{u} = \boldsymbol{G^{-1}(y, \theta)}$ as the value of $\boldsymbol{u}$ such that $\boldsymbol{y} = \boldsymbol{G(U, \theta)}$.

We note that although (2.7) provides an explicit formula to calculate the GFD, it may not be as straightforward as it looks: the denominator requires the calculation of an integral that is

intractable for some problems, and hence Monte Carlo or other numerical techniques are needed to sample from the GFD $r(\boldsymbol{\theta}|\boldsymbol{y})$.

2.2.3. Incorporating Model Selection in GFI. Up to now, our discussion on GFI assumes that the dimension of $\boldsymbol{\theta}$ is fixed and known. In other words, (2.7) cannot be used for model selection problems, where the size of $\boldsymbol{\theta}$ also needs to be chosen.

In the context of wavelet regression, [25] incorporated model selection in the GFI framework, which can be extended to more general situations. The idea is similar to penalized likelihood estimation, where a penalty term is added to the (log)-likelihood function to achieve a balanced trade-off between data fidelity and model complexity. Here we provide a brief description and refer the reader to [17] for further details.

Let $\mathcal{M}$ be the set of all possible models and $\boldsymbol{\theta}_M$ be the parameters of any model $M \in \mathcal{M}$. The GFD of $(\boldsymbol{\theta}_M, M)$ can be expressed as

$$r(\boldsymbol{\theta}_M, M|\boldsymbol{y}) = r(\boldsymbol{\theta}_M|\boldsymbol{y}, M) r(M|\boldsymbol{y}),$$

where the conditional GFD $r(\boldsymbol{\theta}_M|\boldsymbol{y}, M)$ of $\boldsymbol{\theta}_M$ (given M) can be calculated using (2.7), while the marginal GFD $r(M|\boldsymbol{y})$ of M admits the expression

$$r(M|\boldsymbol{y}) = \frac{\int r(\boldsymbol{\theta}_M|\boldsymbol{y}, M) e^{-q(M)} d\boldsymbol{\theta}_M}{\sum_{M' \in \mathcal{M}} \int r(\boldsymbol{\theta}_{M'}|\boldsymbol{y}, M') e^{-q(M')} d\boldsymbol{\theta}_{M'}}, \tag{2.9}$$

where $q(M)$ is the penalty associated with model M.

Different choices of $q(M)$ will lead to different penalty strengths, which will in turn affect the final results. In general, the stronger the penalty, the lesser the number of spectral lines we would expect to obtain. When $q(M)$ is suitably chosen, it leads to some well-known model selection methods commonly used in the signal processing and statistics communities. For example, if we set $q(M) = 2|M|$ with $|M|$ as the number of parameters in model M, we have the Akaike Information Criterion (AIC). Here we follow [26] and choose $q(M)$ as

$$q(M) = \frac{|M|}{2} \log N + \log_{e^{1/\gamma}} \binom{K}{|M|}, \tag{2.10}$$

where K is the number of parameters of the largest model in $\mathcal{M}$. Also, γ is a constant measuring the sparsity belief of the model. A natural choice is $\gamma = 1$, but other choices are also possible, and we note that there is not a universal choice of γ that is suitable for all different kinds of true models. In our work, we use $\gamma = 1$, which aligns (2.10) with the minimum description length principle [29] for high-dimensional problems [30]. This is a main reason behind our choice of $q(M)$, as the minimum description length principle is a well-studied model selection method that often produces excellent theoretical and empirical results.

2.3. GFI for Line Spectral Estimation

This section applies the above GFI methodology to the line spectral estimation problem represented by (2.4). We shall calculate the GFDs for this problem and devise a method for generating fiducial samples. To the best of our knowledge, this is the first time that GFI is being applied to a problem with complex-valued coefficients and responses.

Let M_0 be the true model and M be any candidate model such that $|M| < K$. Given M, the structural equation (2.5) for model (2.4) is:

$$\boldsymbol{y} = \boldsymbol{A}_M \boldsymbol{\beta}_M + \sigma \boldsymbol{U}, \tag{2.11}$$

where $\boldsymbol{A}_M$ and $\boldsymbol{\beta}_M$ represent, respectively, the design matrix and the parameter vector of model M. Also, σ is the standard deviation of the error term and $\boldsymbol{U}$ is a standard multivariate complex normal variable; i.e. $\boldsymbol{U} \sim \mathcal{CN}(\boldsymbol{0}, \boldsymbol{I}_N)$. To calculate the GFD of $\boldsymbol{\theta} = (\sigma, \boldsymbol{\beta})^T$ given M for (2.11), we first compute (2.8)

$$\begin{aligned} J(\boldsymbol{y}, \boldsymbol{\theta}) &= D\left(\frac{\boldsymbol{d}}{\boldsymbol{d\theta}} \boldsymbol{G}(\boldsymbol{U}, \boldsymbol{\theta})|_{\boldsymbol{U}=\boldsymbol{G}^{-1}(\boldsymbol{y},\boldsymbol{\theta})}\right) = D\left(\boldsymbol{A}_M, \frac{\boldsymbol{y} - \boldsymbol{A}_M \boldsymbol{\beta}_M}{\sigma}\right) \\ &= \left[\det\left\{\begin{pmatrix} \boldsymbol{A}_M^H \\ \frac{\boldsymbol{y}^H - \boldsymbol{\beta}_M^H \boldsymbol{A}_M^H}{\sigma} \end{pmatrix} \begin{pmatrix} \boldsymbol{A}_M & \frac{\boldsymbol{y} - \boldsymbol{A}_M \boldsymbol{\beta}_M}{\sigma} \end{pmatrix}\right\}\right]^{\frac{1}{2}} = \sigma^{-1} |\det(\boldsymbol{A}_M^H \boldsymbol{A}_M)|^{\frac{1}{2}} \mathrm{RSS}_M^{\frac{1}{2}}, \end{aligned}$$

where $\boldsymbol{A}_M^H$ is the conjugate transpose of a $\boldsymbol{A}$ and RSS_M is the residual sum of squares of model M.

Next we calculate the GFD of $\boldsymbol{\theta}$ given M using (2.7):

$$r(\boldsymbol{\theta}|\boldsymbol{y}, M) = \frac{c_N \sigma^{-1} \mathrm{RSS}_M^{\frac{1}{2}} (\frac{1}{\sigma^{-2n}}) e^{\frac{-1}{\sigma^2}(\boldsymbol{y}-\boldsymbol{A}_M\boldsymbol{\beta}_M)^H(\boldsymbol{y}-\boldsymbol{A}_M\boldsymbol{\beta}_M)}}{\int_\Theta c_N \sigma^{-1} \mathrm{RSS}_M^{\frac{1}{2}} (\frac{1}{\sigma^{-2n}}) e^{\frac{-1}{\sigma^2}(\boldsymbol{y}-\boldsymbol{A}_M\boldsymbol{\beta}_M)^H(\boldsymbol{y}-\boldsymbol{A}_M\boldsymbol{\beta}_M)} d\boldsymbol{\theta}'},$$

where $c_N = \frac{1}{\pi^N} |\det(\boldsymbol{A}_M^H \boldsymbol{A}_M)|^{\frac{1}{2}}$.

Let K be the length of $\boldsymbol{\beta}_M$. So the numerator of (2.9) can be calculated as:

$$\begin{aligned} r(M|\boldsymbol{y}) \propto & \int \sigma^{-1} |\det(\boldsymbol{A}_M^H \boldsymbol{A}_M)|^{1/2} \mathrm{RSS}_M^{1/2} (\frac{1}{\pi^N \sigma^{-2N}}) \\ & \cdot e^{-\frac{1}{\sigma^2}(\boldsymbol{y}-\boldsymbol{A}_M\boldsymbol{\beta}_M)^H(\boldsymbol{y}-\boldsymbol{A}_M\boldsymbol{\beta}_M)} d\boldsymbol{\theta} e^{-q(M)} \\ = & e^{-q(M)} \cdot \int \pi^{-N} \sigma^{-2N+1} |\det(\boldsymbol{A}_M^H \boldsymbol{A}_M)|^{1/2} \mathrm{RSS}_M^{1/2} d\sigma \\ & \cdot \int e^{-\frac{1}{\sigma^2}(\boldsymbol{y}-\boldsymbol{A}_M\boldsymbol{\beta}_M)^H(\boldsymbol{y}-\boldsymbol{A}_M\boldsymbol{\beta}_M)} d\boldsymbol{\beta}_M, \end{aligned}$$

where the last term is $\int e^{-\frac{1}{\sigma^2}(\boldsymbol{y}-\boldsymbol{A}_M\boldsymbol{\beta}_M)^H(\boldsymbol{y}-\boldsymbol{A}_M\boldsymbol{\beta}_M)} d\boldsymbol{\beta}_M = \pi^{|M|} \sigma^{2|M|} \det(\boldsymbol{A}_M^H \boldsymbol{A}_M)^{-1} \exp(-\frac{\mathrm{RSS}_M}{\sigma^2})$. Therefore

(2.12)

$$\begin{aligned} r(M|\boldsymbol{y}) \propto & \int \pi^{-N} \sigma^{-(2N+1)} |\det(\boldsymbol{A}_M^H \boldsymbol{A}_M)|^{1/2} \pi^{|M|} \sigma^{2|M|} \cdot \frac{1}{\det(\boldsymbol{A}_M^H \boldsymbol{A}_M)} e^{-\frac{1}{\sigma^2}\mathrm{RSS}_M} d\sigma e^{-q(M)} \\ = & \pi^{|M|-N} |\det(\boldsymbol{A}_M^H \boldsymbol{A}_M)|^{-1/2} \mathrm{RSS}_M^{1/2} e^{-q(M)} \cdot \int \sigma^{2|M|-2N-1} e^{-\frac{1}{\sigma^2}\mathrm{RSS}_M} d\sigma \\ = & \pi^{|M|-N} |\det(\boldsymbol{A}_M^H \boldsymbol{A}_M)|^{-1/2} \cdot \mathrm{RSS}_M^{\frac{1}{2}+|M|-N} \cdot \Gamma(N-|M|) \cdot e^{-q(M)}. \end{aligned}$$

2.3.1. Generating Fiducial Samples. This subsection presents a method for generating fiducial samples for the line spectral estimation problem that this chapter considers. The idea is to first generate a candidate model M, then given M, generate $\boldsymbol{\theta} = (\sigma, \boldsymbol{\beta})$.

First of all, due to the large number of columns of $\boldsymbol{A}$ in the line spectral estimation context, we are facing an extremely large number of potential models in the model set $\mathcal{M}$; i.e., the cardinality of $\mathcal{M}$ equals 2^K, which is often intractable. Therefore, for various practical considerations, we only consider models from a subset $\mathcal{M}^*$ of $\mathcal{M}$. We delay our discussion of how to choose $\mathcal{M}^*$ to Appendix 2.8.1. In principle, an ideal $\mathcal{M}^*$ should include all the models that have a non-negligible

value of $r(M|\boldsymbol{Y})$, while at the same time excluding other models that have a zero or near-zero $r(M|\boldsymbol{Y})$ value.

Suppose now we have a good $\mathcal{M}^*$. For each $M \in \mathcal{M}^*$, we compute (see (2.12))

$$R(M) = \pi^{|M|-N} |\det(\boldsymbol{A}_M^H \boldsymbol{A}_M)|^{-1/2} \cdot \mathrm{RSS}_M^{\frac{1}{2}+|M|-N} \cdot \Gamma(N-|M|) \cdot e^{-q(M)},$$

where $e^{-q(M)}$ is given by (2.10). The generalized fiducial probability $r(M|\boldsymbol{y})$ (2.12) can then be well approximated by

$$r(M|\boldsymbol{y}) \approx \frac{R(M)}{\sum_{M^* \in \mathcal{M}^*} R(M^*)}. \tag{2.13}$$

We can then sample a candidate model $M \in \mathcal{M}^*$ from (2.13).

Once a model M is generated, we set up the corresponding design matrix $\boldsymbol{A}_M$. Then we estimate the parameters $\boldsymbol{\beta}_M$ of the generated model M using maximum likelihood and obtain the estimate $\hat{\boldsymbol{\beta}}_{\mathrm{ML}}$ and the corresponding residual sum of squares RSS_M. As $\boldsymbol{A}_M^H \boldsymbol{A}_M$ is of full rank (i.e., not in a high-dimensional setting), these two quantities can be calculated using classical regression formulae: $\hat{\boldsymbol{\beta}}_{\mathrm{ML}} = (\boldsymbol{A}_M^H \boldsymbol{A}_M)^{-1} \boldsymbol{A}_M^H \boldsymbol{y}$ and $\mathrm{RSS}_M = \boldsymbol{y}^H (I - \boldsymbol{A}_M (\boldsymbol{A}_M^H \boldsymbol{A}_M)^{-1} \boldsymbol{A}_M^H) \boldsymbol{y}$. Then, using the properties of the complex normal distribution, σ and $\boldsymbol{\beta}$ can be sampled using the following distributional results:

$$\frac{2\mathrm{RSS}_M}{\sigma^2} \sim \chi^2_{2(N-|M|)} \tag{2.14}$$

and

$$\boldsymbol{\beta} \sim \mathcal{CN}(\hat{\boldsymbol{\beta}}_{\mathrm{ML}}, \sigma^2 (\boldsymbol{A}_M^H \boldsymbol{A}_M)^{-1}), \tag{2.15}$$

where $\chi^2_{2(N-|M|)}$ is the chi-square distribution with $2(N-|M|)$ degrees of freedom.

To sum up, a fiducial sample for $(M, \sigma, \boldsymbol{\beta})$ can be generated by the following steps.

(1) Sample a model M from $\mathcal{M}^*$ using (2.13).

(2) Fit M using maximum likelihood and obtain $\hat{\boldsymbol{\beta}}_{\mathrm{ML}}$ and RSS_M.

(3) Sample σ^2 using (2.14).

(4) Sample $\boldsymbol{\beta}$ using (2.15), where the σ^2 obtained from the above step is used in the RHS of (2.15).

By repeating the above steps, one can generate enough samples of $(M, \sigma, \boldsymbol{\beta})$ for forming point estimates and constructing confidence intervals. Notice that (2.13) only needs to be calculated once, so it is fast to generate an M. Also, notice that no costly procedures are required to generate σ or $\boldsymbol{\beta}$ so overall the whole sample method is fast.

2.3.2. Point Estimates and Confidence Intervals. Repeating the above procedure, we obtain multiple fiducial samples for $(M, \sigma, \boldsymbol{\beta})$, which can be used to perform statistical inference in a similar manner as with posterior samples in the Bayesian context. For the case of σ, we can use the mean or the median of its fiducial samples as a point estimate, and the $(\alpha/2, 1-\alpha/2)$ quantiles of the fiducial samples to be its $100(1-\alpha)\%$ confidence intervals.

The situation is less straightforward for M, as its domain $\mathcal{M}$ is a discrete space with 2^K elements and it is not entirely clear what would be a universally accepted definition for a "confidence interval" for a model. However, the fiducial samples of M could still provide valuable information on uncertainties. For example, for any M the samples can be used to approximate the generalized fiducial probability $r(M|\boldsymbol{y})$ as in (2.13), which is a numerical measure indicating how likely (or unlikely) M is the true model.

The fiducial samples can also provide uncertainty information for p, the number of significant frequencies. For example, the generalized fiducial probability for $p = l$ can be approximated by the sum of the generalized fiducial probabilities $r(M|\boldsymbol{y})$ of all models M with $p = l$.

One can also construct confidence intervals for $\boldsymbol{\beta}$ in the following manner. First, notice that, unless the GFD $r(M|\boldsymbol{y})$ has all its mass at one model, the fiducial samples will contain different models. In other words, for any $l = 1, \ldots, K$, β_l may be declared insignificant by some of the fiducial samples. These insignificant fiducial samples for β_l are zero, which will have an adverse effect when calculating averages or quantiles with those non-zero β_l values. We follow [26] to handle this issue: for each β_l, we count the percentage of non-zero fiducial sample values. If it is

more than 50%, we claim that this specific β_l is significant and use all the non-zero fiducial sample values to obtain point estimates and confidence intervals, in the same manner as for σ.

2.4. Theoretical Properties

This section investigates the theoretical properties of the above proposed GFI-based method under the situation that K is diverging and the size of the true model is fixed.

First, some notations. Recall M is any candidate model and M_0 is the true model. Let $\boldsymbol{P}_M$ be the projection matrix of $\boldsymbol{A}_M$; i.e., $\boldsymbol{P}_M = \boldsymbol{A}_M(\boldsymbol{A}_M^H\boldsymbol{A}_M)^{-1}\boldsymbol{A}_M^H$. Define $\Delta_M = \|\boldsymbol{\mu} - \boldsymbol{P}_M\boldsymbol{\mu}\|^2$, where $\boldsymbol{\mu} = E(\boldsymbol{Y}) = \boldsymbol{A}_{M_0}\boldsymbol{\beta}_{M_0}$.

Throughout this section, we assume that the following identifiability condition holds:

$$\lim_{n\to\infty} \min\left\{ \frac{\Delta_M}{|M_0|\log(K)} : M_0 \not\subset M, |M| \le b|M_0| \right\} = \infty \tag{2.16}$$

for some fixed constant $b > 1$. This b ensures that we only consider models whose size is comparable to the true model. This assumption is an identifiability condition because it guarantees the uniqueness of the true model among all the models that have a comparable size to the true model. To be more specific, this condition guarantees that if the true model $M_0 \not\subset M$, the residuals will become unbounded as $n \to \infty$. The restriction $|M| \le b|M_0|$ is imposed because in practice only those models with sizes comparable to the true model will be considered. Overall, this assumption means the true model is identifiable if no model other than the true model of comparable size can predict the response almost equally well, which ensures the true model can be differentiated from the other models.

THEOREM 2.4.1. *Assume condition* (2.16). *If* $N \to \infty$, $K \to \infty$, $|M_0|\log(K) = o(N)$, $\frac{\log(|M_0|)}{\log(K)} \to \delta$ *and* $\frac{\log(N)}{\log(K)} \to \eta$, *then there exists* $\gamma > \frac{1+\delta}{1-\delta} - \frac{5\eta}{2(1-\delta)}$ *such that*

$$\max_{M\neq M_0, M\in\mathcal{M}^*} \frac{r(M)}{r(M_0)} \xrightarrow{P} 0. \tag{2.17}$$

Moreover, Suppose there exists a procedure for obtaining $\mathcal{M}^*$ that satisfies:

$$P\left(M_0 \in \mathcal{M}^*\right) \to 1 \text{ and } \log\left(\left|\mathcal{M}_j^*\right|\right) = o(j\log(N)), \tag{2.18}$$

where $\mathcal{M}_j^*$ denotes the set of all sub-models in $\mathcal{M}^*$ of size j, we have

$$r(M_0) \xrightarrow{P} 1.$$

Theorem 2.4.1 implies that, under some regularity conditions, the true model M_0 has the highest generalized fiducial probability amongst all the candidate models. Assumption (2.18) guarantees the true model in the candidate set and the candidate set not to be too large. The proof of this theorem is provided in the appendix.

2.5. Simulation Results

Two simulation experiments were conducted to evaluate the practical performance of the proposed GFI method under the line spectral model (2.1).

2.5.1. Confidence Intervals and Widths. In the first experiment, we follow the experimental setting of [5], where

- Number of spectral lines: $p = 3$.
- Parameters: $f_1 = 0.4230$, $f_2 = 0.6875$, $f_3 = f_2 + \delta_f$, $\alpha_1 = 5e^{i2\pi u_1}$, $\alpha_2 = 5e^{i2\pi u_2}$ and $\alpha_1 = 10e^{i2\pi u_3}$, where u_1, u_2, u_3 are randomly chosen from Unif$(0,1)$. See below for a discussion on the values used for δ_f.
- Number of observations: $N = 50$.
- Sampling times: $t_1 = 0$[sec], $t_N = 50$[sec], and $\{t_k\}_{k=2}^{49}$ are uniformly randomly selected (real numbers) from the interval $(0, 50)$.
- Noise: $\boldsymbol{\epsilon}$ is sampled from a complex normal distribution $\mathcal{CN}(0, \sigma^2 \boldsymbol{I}_N)$.

The signal-to-noise ratio (SNR) is defined as

$$SNR = 10\log_{10}\left(\frac{|\alpha_1|^2 + |\alpha_2|^2 + |\alpha_3|^2}{\sigma^2}\right) = 10\log_{10}(\frac{150}{\sigma^2}).$$

As in here we have $\min_k(t_{k+1} - t_k) < 0.5$[sec], we can set $f_{\max} = 1$[Hz]. For the choices of K and Δ in (2.3), we adopted the suggestion by [5] and set

$$\Delta = \frac{1}{c(t_N - t_1)}$$

with $c = 20$, which gives $\Delta = 1 \times 10^{-3}$[Hz]. As we chose $f_{\max} = 1$[Hz], using (2.3) we have $K = 1000$.

For the frequency separation δ_f between f_2 and f_3, we considered three values: $\delta_f = \{0.01, 0.015, 0.1\}$. The first two values are considered "high-resolution" cases, and the last is a "normal" case. We also considered two SNRs=$\{5, 10\}$. Therefore, we have six different scenarios in this first simulation experiment. For each scenario, we generated 1,000 data sets and applied the proposed GFI method to each of them, where the number of fiducial samples for each data set was 10,000.

Recall that, unlike many traditional methods, the proposed GFI method also provides the generalized fiducial probabilities $r(M|\boldsymbol{Y})$ for all the candidate models, which in turn can be used to generate the corresponding generalized fiducial probabilities for the number of frequencies p; see Section 2.3.2. These probabilities provide valuable information about how certain or uncertain we are with the estimated results. For the six scenarios, Table 2.1 lists the percentages of times that different frequency numbers p were selected by the fiducial samples. As expected, the percentages for choosing the correct p are higher when the separation δ_f and/or the SNR are higher. Also, given the high percentages for selecting the true $p = 3$, one may conclude that the GFI estimation results are reliable.

For each data set, we also applied the LIKES method of [31] and the so-called Oracle method that has the knowledge of the true p and f_l's. Of course, the Oracle results cannot be obtained in practice as such knowledge is not available, but they are used here for benchmark comparisons. Table 2.2 provides the empirical coverage rates of the confidence intervals from Oracle and the GFI method for σ^2 (note that LIKES does not produce confidence intervals for σ^2). One can observe that the GFI results are comparable to those from Oracle.

TABLE 2.1. Percentages of times that different frequency numbers p were selected by GFI in the six different scenarios. Recall that the true $p = 3$.

estimated number of frequencies:		1	2	3	4
δ_f	SNR	percentages selected			
0.01	5	2%	12.5%	83.3%	2.2%
0.01	10	0%	7.5%	92.1%	0.4%
0.015	5	1.1%	11.2%	86.5%	1.2%
0.015	10	0%	6.2%	93.2%	0.6%
0.1	5	0.9%	10.3%	87.6%	1.2%
0.1	10	0%	4.5%	94.4%	0.1%

We also constructed confidence intervals for the amplitudes α_l's using all three methods: GFI, LIKES, and Oracle. Note that for GFI and LIKES, we only used those results where the true number of frequencies was selected. The empirical coverage rates of these confidence intervals are reported in Table 2.3. One can see that the GFI results are slightly worse than those from Oracle, but in general, are superior to those from LIKES. Also, very often, GFI produced higher empirical coverage rates with narrower confidence intervals.

2.5.2. Comparison with Bayesian Approach. In this second experiment, the simulation setting is similar to [14]:

- Number of spectral lines: $p = 3$.
- Parameters: $f_1 = 0.4230$, $f_2 = 0.6875$, $f_3 = 0.7875$, $\alpha_1 = 1 + 0.1e^{i2\pi u_1}$, $\alpha_2 = 1 + 0.1e^{i2\pi u_2}$ and $\alpha_3 = 5 + 0.1e^{i2\pi u_3}$, where u_1, u_2, u_3 are randomly chosen from Unif$(0, 1)$.
- Other quantities such as the sampling times are the same as in the first experiment.

As in [14] we use these two metrics to measure the quality of the estimation results: the normalized mean-squared-error (NMSE) of $\hat{\boldsymbol{A}}$ (only with those columns selected by the methods) and the mean-squared-error (MSE) of $\boldsymbol{f} = (f_1, f_2, f_3)$, defined respectively as

$$\text{NMSE}(\hat{\boldsymbol{A}}) = 20\log(\|\boldsymbol{A}\boldsymbol{\beta} - \hat{\boldsymbol{A}}\hat{\boldsymbol{\beta}}\|_F / \|\hat{\boldsymbol{A}}\hat{\boldsymbol{\beta}}\|_F)$$

TABLE 2.2. Empirical coverage rates of the confidence intervals for σ^2 obtained by the proposed GFI method and Oracle. The numbers in the parentheses are the average widths of the intervals.

(δ_f, SNR)	**method**	**90% CI**	**95% CI**	**99% CI**
(0.1, 5)	GFI	91.0% (1.81)	93.8% (1.14)	98.4% (2.88)
	Oracle	88.7% (1.67)	94.4% (2.00)	99.0% (2.66)
(0.1, 10)	GFI	91.6% (0.97)	96.0% (1.16)	99.6% (1.53)
	Oracle	89.7% (0.94)	94.7% (1.12)	99.1% (1.49)
(0.015, 5)	GFI	90.8% (1.81)	96.0% (2.23)	98.5% (2.87)
	Oracle	88.3% (1.68)	94.5% (2.01)	99.3% (2.67)
(0.015, 10)	GFI	88.6% (0.95)	93.8% (1.14)	98.0% (1.51)
	Oracle	89.0% (0.95)	94.8% (1.13)	98.7% (1.50)
(0.01, 5)	GFI	90.6% (1.82)	95.8% (2.76)	98.8% (2.86)
	Oracle	90.5% (1.68)	95.0% (2.00)	98.9% (2.66)
(0.01, 10)	GFI	88.8% (0.96)	96.0% (1.15)	99.6% (1.52)
	Oracle	89.5% (0.94)	94.0% (1.13)	98.5% (1.49)

and

$$\text{MSE}(\hat{\boldsymbol{f}}) = 20\log(\|\hat{\boldsymbol{f}} - \boldsymbol{f}\|_2),$$

where $\|\cdot\|_F$ is the Frobenius norm for matrices and $\|\cdot\|_2$ is the L_2 norm for vectors. Following [14], $\text{MSE}(\hat{\boldsymbol{f}})$ is calculated only when both the model order p is correctly estimated and $\text{MSE}(\hat{\boldsymbol{f}}) \leq 0(dB)$. In addition, we also approximated the probability that the correct model order p is selected; i.e., $P(\hat{p} = 3)$.

For each simulated data set, we applied the GFI method and the MVALSE method of [14] and calculated the above metrics. Figure 2.1 summarizes the results when the number of observations is fixed at $N = 75$ with changing SNRs $= \{-5, 0, 5, 10\}$. One can observe that when SNR $= 10$, both methods give comparable results, while GFI is better for the remaining

TABLE 2.3. Empirical coverage rates of the confidence intervals for the frequency amplitudes obtained by the proposed GFI method, LIKES, and Oracle. The numbers in the parentheses are the average widths of the intervals.

	method	**90% CI**	**95% CI**	**99% CI**
(δ_f, SNR) = (0.1, 5)	GFI	90.2%, 87.9%, 90.1% (2.4, 2.4, 2.4)	95.5%, 93.8%, 94.6% (2.9, 2.8, 2.9)	98.4%, 98.4%, 99.2% (3.9, 3.8, 3.8)
	LIKES	84.9%, 84.9%. 85.8% (6.5, 6.4, 6.8)	90.2%, 92.0%, 89.3% (6.1, 6.1, 6.6)	94.9%, 95.6%, 94.5% (8.6, 8.6, 8.9)
	Oracle	88.3%, 89.4%, 90.2% (2.3, 2.3, 2.3)	94.6%, 95.4%, 95.7% (2.8, 2.8, 2.8)	99.2%, 98.3%, 99.1% (3.6, 3.6, 3.7)
(δ_f, SNR) = (0.1, 10)	GFI	91.0%, 91.5%, 91.7% (1.4 1.4, 1.4)	96.6%, 95.8%, 95.8% (1.7, 1.7, 1.7)	98.4%, 98.6%, 99.3% (2.3, 2.2, 2.3)
	LIKES	83.4%, 84.4%, 84.2% (2.2, 2.2, 2.2)	89.6%, 91.0%, 90.6% (2.6, 2.6, 2.6)	96.4%, 97.0%, 95.6% (3.4, 3.4, 3.5)
	Oracle	87.8%, 89.8%, 90.0% (1.3, 1.3, 1.3)	95.4%, 94.8%, 93.8% (1.6, 1.6, 1.6)	99.2%, 99.0%, 99.0% (2.1, 2.1, 2.1)
(δ_f, SNR) = (0.015, 5)	GFI	89.5%, 88.9%, 89.3% (2.4, 2.7, 2.6)	95.2%, 94.0%, 95.1% (2.8, 3.0, 2.8)	98.6%, 97.6%, 98.2% (3.7, 4.1, 3.9)
	LIKES	85.8%, 84.7%, 81.3% (6.3, 6.4, 6.6)	92.8%, 90.2%, 89.9% (6.6, 6.8, 6.8)	95.1%, 95.8%, 94.7% (8.8, 8.7, 8.9)
	Oracle	90.3%, 90.0%, 89.5% (2.3, 2.4, 2.4)	95.3%, 95.1%, 95.1% (2.8, 2.9, 2.9)	98.9%, 98.8%, 97.8% (3.6, 3.7, 3.8)
(δ_f, SNR) = (0.015, 10)	GFI	91.5%, 91.4%, 91.5% (1.5, 1.7, 1.6)	96.1%, 95.9%, 96.0% (1.8, 1.9, 1.9)	99.2%, 99.5%, 99.2% (2.2, 2.6, 2.4)
	LIKES	84.2%, 84.4%, 85.2% (2.2, 2.2, 2.2)	89.0%, 85.6%, 88.2% (2.6, 2.6, 2.6)	95.6%, 96.4%, 96.4% (3.4, 3.4, 3.5)
	Oracle	91.1%, 89.7%, 90.8% (1.3, 1.4, 1.4)	95.2%, 95.6%, 94.5% (1.8, 1.9, 1.9)	98.8%, 98.8%, 98.8% (2.1, 2.1, 2.1)
(δ_f, SNR) = (0.01, 5)	GFI	90.8%, 89.6%, 87.1% (2.4, 3.0, 3.1)	95.4%, 94.4%, 92.8% (2.9, 3.7, 3.7)	98.3%, 98.7%, 98.0% (3.9, 4.9, 4.8)
	LIKES	86.9%, 85.9%, 85.6% (6.3, 6.3, 6.1)	90.3%, 93.0%, 89.4% (6.9, 6.9, 6.4)	95.0%, 95.9%, 95.7% (8.8, 8.2, 8.4)
	Oracle	90.0%, 91.5%, 88.9% (2.3, 2.9, 2.9)	93.9%, 94.8%, 94.1% (2.8, 3.5, 3.5)	99.4%, 99.1%, 98.7% (3.6, 4.6, 4.6)
(δ_f, SNR) = (0.01, 10)	GFI	91.8%, 91.3%, 89.2% (1.5, 1.9, 1.9)	96.7%, 95.1%, 93.5% (1.7, 2.2, 2.2)	99.2%, 98.9%, 98.1% (2.3, 2.9, 2.9)
	LIKES	85.2%, 85.4%, 86.2% (2.2, 2.2, 2.2)	90.4%, 91.4%, 90.8% (2.6, 2.6, 2.6)	97.0%, 96.2%, 96.6% (3.4, 3.4, 3.5)
	Oracle	89.5%, 88.7%, 91.3% (1.3, 1.6, 1.6)	94.1%, 94.6%, 93.8% (1.6, 2.0, 2.0)	98.8%, 99.2%, 98.7% (1.7, 2.2, 2.2)

SNRs. Similarly, Figure 2.2 presents the results when SNR $= 2$ is fixed for different values of $N = \{25, 50, 75, 100, 125\}$. The results suggest that GFI is superior.

To sum up, results from these two sets of numerical experiments suggest that the proposed GFI method produces highly reliable results, and compares favorably with some of the leading

methods in the literature. This agrees largely with the authors' experience in applying GFI to other problems. A thorough theoretical study is underway to identify those conditions under which GFI is expected to produce reliable results.

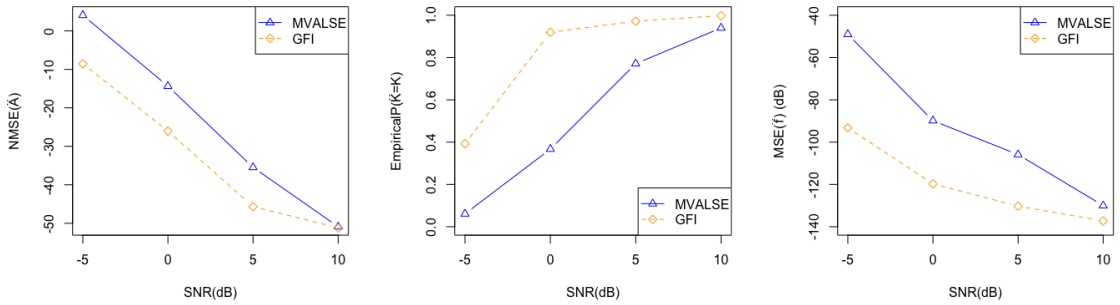


FIGURE 2.1. Empirical performances of the MVALSE method [14] and the proposed GFI method with different SNRs and $N = 75$.

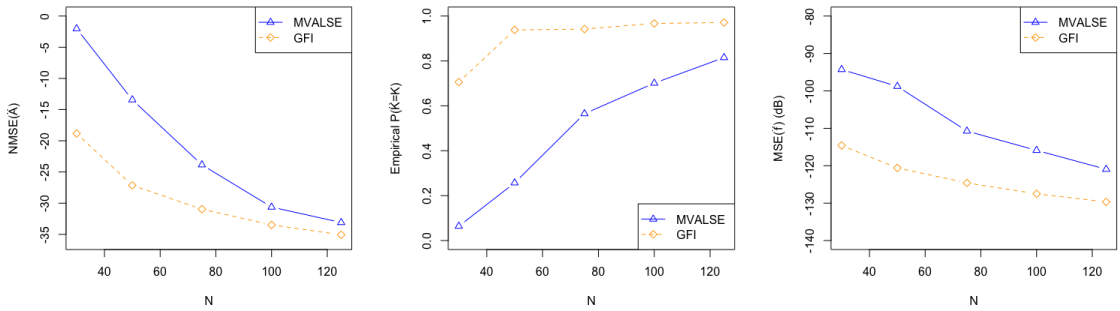


FIGURE 2.2. Empirical performances of the MVALSE method [14] and the proposed GFI method with different N and SNR$= 2$.

2.6. Real Data Example: Radial Velocity Analysis

2.6.1. Background. The detection of extrasolar planets, also known as exoplanets, has always been a challenging and fascinating area in astronomy. Until the end of 2021, a total of 1274 exoplanets have been discovered. Popular techniques for exoplanet detection include radial velocity analysis, the transit method, direct imaging, gravitational microlensing, and astrometry minuscule

movements; e.g., [32], [33]. Among these techniques, radial velocity analysis is one of the most commonly used.

Radial velocity refers to the speed at which an object (in this case an exoplanet) moves away from Earth (or approaches it, with a negative radial velocity). Orbiting exoplanets cause the stars to wobble in space, which in turn changes the color of the light astronomers observe. This permits an analysis of the Doppler shifts to confirm if there is any exoplanet revolving around a star. In order to do so, the radial velocity frequencies and amplitudes of the stars, need to be estimated. Notice that the radial velocity measurements are often obtained at non-uniformly spaced time intervals due to hardware and practical constraints, which limits the applications of many spectral analysis methods designed for equally-spaced data.

Here we apply the proposed GFI method to estimate the radial velocity frequencies and amplitudes of three different stars: HD 63454 [34], HD 208487 [35], and GJ 876 [36]. We note that the model we use ((2.2) and (2.4)) is simpler than those that are based on Keplerian's planetary motion, which also consider eccentricity and periastron parameters of the orbital planets thus more accurate; e.g., [32]. We also note that our model is complex-valued while the radial velocity data are real-valued. To circumvent this issue, we follow [32] and require both positive and negative frequencies in the model to represent a real-valued component. Below we compare our results with those reported in [32].

2.6.2. HD 63454. The radial velocity data set of star HD 63454 contains 26 samples spanning 350 days. The sampling pattern and the radial velocity measurements are shown in Figures 2.3(a) and (b), respectively. The proposed GFI method was applied to the data set and the results are shown in Table 2.4, where f represents its orbital frequency (in cycles day^{-1}) and β represents the corresponding amplitude. As the results suggest, only one exoplanet was detected whose estimated frequency was 0.3549 cycles day^{-1} (i.e., an orbital period of 2.8176 days), which is the same as in [32]. The GFI estimated amplitude is smaller than the one reported by [32] but the corresponding GFI confidence interval does cover it, so overall the GFI results are consistent with those in [32] for HD 63454. Table 2.5 shows the percentages that different numbers of exoplanets were selected.

One can see that for this star the proposed method is highly confident (99.9%) that there is only one exoplanet.

TABLE 2.4. Estimated results for star HD 63454 obtained from different methods. The GFI 95% confidence interval is given in parentheses.

	[32]		GFI	
Exoplanet No	$\hat{f}$	$\hat{\beta}$	$\hat{f}$	$\hat{\beta}$
1	0.3549	0.0634	0.3549	0.0482 (0.0255, 0.0708)

TABLE 2.5. Percentages that different numbers of exoplanets were selected for the three stars.

Number of exoplanets detected	1	2	3
HD 63454	99.9%	0.1%	0
HD 208478	93.9%	6.1%	0
GJ 876	0.03%	15.02%	84.95%

2.6.3. HD 208487. The data set for star HD 208487 contains 31 samples spanning 2250 days. The sampling pattern and the radial velocity measurements are displayed in, respectively, Figures 2.3(c) and (d). The GFI method only detected one exoplanet with an estimated orbital frequency of 0.0078 cycles day^{-1}, while 3 detected exoplanets were reported in [32]; see Table 2.6. However, as noted in both [32] and [35], there is no convincing evidence to support the claim of the existence of the two additional exoplanets for this star system, so the GFI method provided reasonable results for HD 208487. Table 2.5 also provides strong evidence (around 94%) that there is only one exoplanet for this star.

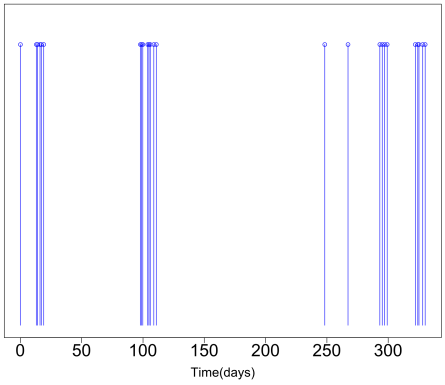
TABLE 2.6. Similar to Table 2.4 but for star HD 208487.

	[32]		GFI	
Exoplanet No	$\hat{f}$	$\hat{\beta}$	$\hat{f}$	$\hat{\beta}$
1	0.0078	19.9	0.0078	17.5 (13.7, 21.3)
2	0.0690	12.18	-	-
3	0.0408	4.96	-	-

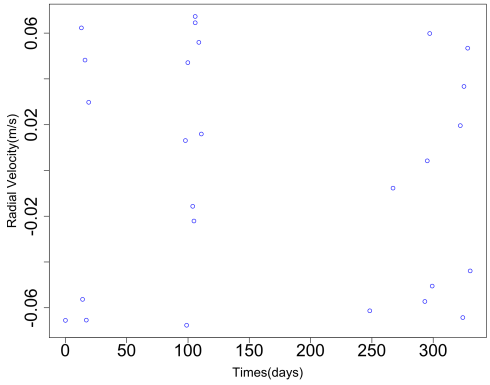
2.6.4. GJ 876. The last data set is for star GJ 876. It consists of 100 samples spanning 2000 days; see Figures 2.3(e) and (f) for the sampling pattern and the radial velocity measurements, respectively. The results are shown in Table 2.7. The GFI method detected 3 exoplanets with orbital frequencies 0.0165, 0.0332 and 0.0666 cycles day^{-1}. A previous study by [36] also detected 3 exoplanets, but with a different orbital frequency (0.516 cycles day^{-1}) for the last one. The method of [32] detected 5 exoplanets. However, [32] also suggested that there is no concrete evidence to support the existence of the additional 2 exoplanets. In any case, all these methods agreed on the first 2 exoplanets in this star system. The uncertainty information in Table 2.5 also suggests that there are three exoplanets, but with lower confidence (around 85%). This indicates that for this star, the true number of exoplanets is more challenging to estimate, as can be seen from the very different results obtained from previous studies.

2.7. Concluding Remarks

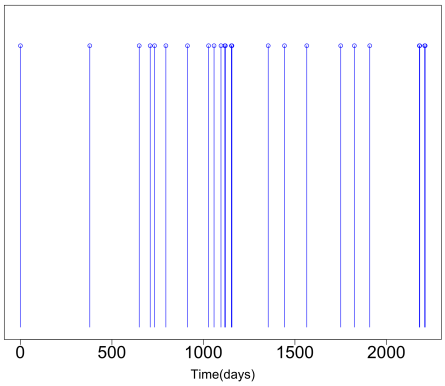
This chapter developed a new method to perform statistical inference on the line spectral estimation problem. The proposed method is based on the approach of generalized fiducial inference.



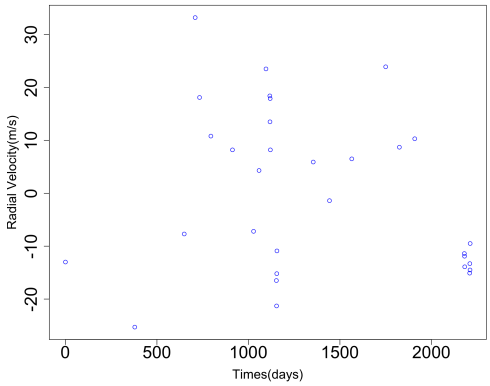
(a): HD 63454, t_k



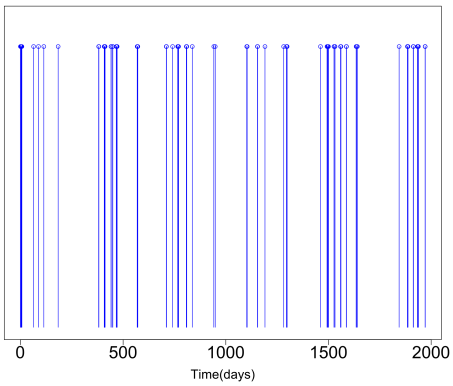
(b) HD 63454, $Y(t_k)$



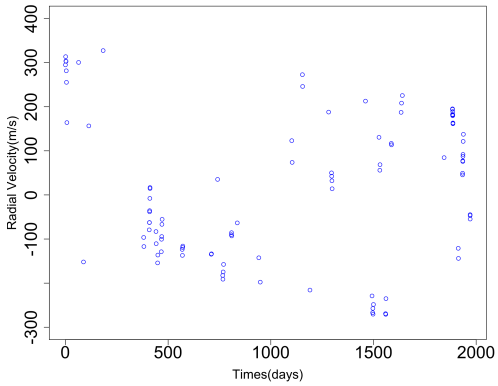
(c): HD 208487, t_k



(d): HD 208487, $Y(t_k)$



(e): GJ 876, t_k



(f): GJ 876, $Y(t_k)$

FIGURE 2.3. Sampling times t_k's (left column) and radial velocity measurements $Y(t_k)$'s (right column) for stars HD 63454 (top row), HD 208487 (middle row), and GJ 876 (bottom row).

TABLE 2.7. Similar to Table 2.4 but for star GJ 876.

	[32]		[36]		GFI	
Exoplanet No	$\hat{f}$	$\hat{\beta}$	$\hat{f}$	$\hat{\beta}$	$\hat{f}$	$\hat{\beta}$
1	0.0164	215.09	0.0164	212.60	0.0165	201.03 (181.44, 220.61)
2	0.0331	82.62	0.0331	88.36	0.0332	69.74 (46.99, 92.49)
3	0.0011	9.61	0.516	6.40	0.0666	31.86 (10.44, 53.27)
4	0.0066	9.95	-	-	-	-
5	0.0168	23.57	-	-	-	-

In greater detail, a procedure was developed to generate fiducial samples from a so-called generalized fiducial density for a set of candidate models. This generalized fiducial density plays a similar role as the posterior density in the Bayesian context. Its samples (i.e., fiducial samples) can be used to perform statistical inferences such as forming point estimates and confidence intervals. The proposed method was shown to enjoy desirable asymptotic properties under some regularity conditions. Through numerical experiments, it was also demonstrated that the proposed method possesses promising empirical properties and often outperforms existing methods in the literature. Lastly, the proposed method was applied to analyze three radial velocity data sets in the context of exoplanet detection and yielded similar results as those reported in the astronomy literature.

Recall that our method is an example of semi-parametric method, which can be further divided into three categories [37]: on-grid, off-grid, and gridless. The on-grid methods require a pre-selected grid and the true frequencies to be one of the grid values. Our method can be classified as on-grid. However, as mentioned in Section 2.1 and demonstrated by the simulation experiments in Section 2.5, our method can handle the situation when some of the true frequencies do not fall on the grid, by suitably choosing the values of K and Δ. For off-grid methods, they also require a

grid, which is estimated jointly with the sparse signals. Consequently, more variables are needed to be estimated, which increases the dimension of the problem. The last category of gridless methods does not require any grid when compared with the first two categories. However, they are typically designed for equally spaced sampled data, which may restrict their applicability. We believe that GFI can be applied to these methods, but the form of the structural equation (2.5) will need to be formulated differently. Overall, we are confident that the GFI approach can be applied to these off-grid and gridless methods. The main challenge will be the development of a practical algorithm for generating the fiducial samples. These are left for future work.

2.8. Supplementary materials

2.8.1. Obtaining $\mathcal{M}^*$. This appendix presents our method for obtaining $\mathcal{M}^*$. Recall that an ideal $\mathcal{M}^*$ should only contain those models that have a non-negligible value of $r(M|\boldsymbol{Y})$. Our method consists of two stages. The first stage applies a fast algorithm to traverse the space of $\mathcal{M}$ to obtain a set of non-negligible models, where the true model will be included with high probability. In the second stage, we obtain more models by data perturbation and add these models to $\mathcal{M}^*$. Notice that we are not choosing the models by comparing their values of $r(M|\boldsymbol{Y})$ with a threshold.

Stage 1: In the context of ultra-high dimensional regression, the lasso algorithm [38] [39] has been applied by [26] to obtain a $\mathcal{M}^*$. The idea is that, by changing the lasso tuning parameter, a sequence of models (also known as a solution path) will be generated, and all these models form $\mathcal{M}^*$. We shall follow this idea in the first stage of our method. This is a variable selection problem that is studied widely [40]. However, due to the complex-valued coefficients, the lasso algorithm cannot be directly applied, as it does not guarantee to select both the real and imaginary parts of a complex coefficient simultaneously. To circumvent this, one can use for example the complex lasso [41]. Below, however, we shall re-express the problem and apply the group lasso algorithm of [42].

First, express the lasso problem as:

$$\min_{\boldsymbol{\beta}}(\frac{1}{2}\|\boldsymbol{X}\boldsymbol{\beta}-\boldsymbol{y}\|_2^2+\lambda\|\boldsymbol{\beta}\|_1^*), \tag{2.19}$$

where $\boldsymbol{X}=\Re(\boldsymbol{X})+i\Im(\boldsymbol{X})\in\mathbb{C}^{N\times K}$, $\boldsymbol{y}=\Re(\boldsymbol{y})+i\Im(\boldsymbol{y})\in\mathbb{C}^N$, $\boldsymbol{\beta}=\Re(\boldsymbol{\beta})+i\Im(\boldsymbol{\beta})\in\mathbb{C}^K$, and

$$\|\boldsymbol{\beta}\|_1^*=\sum_{j=1}^{n}\sqrt{\Re(\beta))_j^2+\Im(\beta)_j^2},$$

with $\Re(\beta),\Im(\beta)\in\mathbb{R}$ and $j=1,\ldots,n$. Minimizing (2.19) with different values of λ will give different models. However, as suggested before, there is no guarantee that all the resulting models are legitimate in the sense that the corresponding estimates in $\Re(\boldsymbol{\beta})$ and $\Im(\boldsymbol{\beta})$ are both zeros or non-zeros.

Now we can re-express

$$\begin{aligned}\|\boldsymbol{X}\boldsymbol{\beta}-\boldsymbol{y}\|_2^2=&\|\Re(\boldsymbol{X})\Re(\boldsymbol{\beta})-\Im(\boldsymbol{X})\Im(\boldsymbol{\beta})-\Re(\boldsymbol{y})\|_2^2\\&+\|\Re(\boldsymbol{X})\Im(\boldsymbol{\beta})+\Im(\boldsymbol{X})\Re(\boldsymbol{\beta})-\Im(\boldsymbol{y})\|_2^2\\=&\left\|\begin{pmatrix}\Re(\boldsymbol{X})&-\Im(\boldsymbol{X})\\\Im(\boldsymbol{X})&\Re(\boldsymbol{X})\end{pmatrix}\begin{pmatrix}\Re(\boldsymbol{\beta})\\\Im(\boldsymbol{\beta})\end{pmatrix}-\begin{pmatrix}\Re(\boldsymbol{y})\\\Im(\boldsymbol{y})\end{pmatrix}\right\|_2^2\end{aligned}$$

and (2.19) becomes

$$\min_{\boldsymbol{\beta}}(\frac{1}{2}\|\tilde{\boldsymbol{X}}\tilde{\boldsymbol{\beta}}-\tilde{\boldsymbol{y}}\|_2^2+\lambda\|\tilde{\boldsymbol{\beta}}\|_{2,1}) \tag{2.20}$$

with $\tilde{\boldsymbol{X}}=\begin{pmatrix}\Re(\boldsymbol{X})&-\Im(\boldsymbol{X})\\\Im(\boldsymbol{X})&\Re(\boldsymbol{X})\end{pmatrix}$, $\tilde{\boldsymbol{y}}=\begin{pmatrix}\Re(\boldsymbol{y})\\\Im(\boldsymbol{y})\end{pmatrix}$, $\tilde{\boldsymbol{\beta}}=\begin{pmatrix}\Re(\boldsymbol{\beta})\\\Im(\boldsymbol{\beta})\end{pmatrix}$ and $\|\boldsymbol{\beta}\|_{2,1}=\sum_{j=1}^{n}\sqrt{\Re(\beta)_j^2+\Im(\beta)_j^2}$. With the above, we can apply the group lasso algorithm to (2.20) to generate different models with different values of λ. In practice, we observe that the true model was almost always included as one of the models generated by this algorithm.

Stage 2: To achieve theoretical guarantee, in the second stage, we apply the adaptive group lasso algorithm to generate more models, which was shown by [43] that the true model will be selected consistently. We can also obtain more models by applying the group lasso algorithm

to various re-sampled data sets [44], so that in practice most non-negligible models are included in $\mathcal{M}^*$. We can yet further enrich $\mathcal{M}^*$ by adding solutions from other methods to $\mathcal{M}^*$, such as SPICE [45] and GIST [46]. By doing so, we expect the size of $\mathcal{M}^*$ to be much smaller than the size of $\mathcal{M}$ (which is 2^K) and yet $\sum_{M\in\mathcal{M}^*} r(M|\boldsymbol{Y})$ is close to 1.

Lastly, we note that before we generate the fiducial samples, the model parameters will be re-fitted using maximum likelihood, and therefore the parameter estimation bias from group lasso will not be carried over.

2.8.2. Proof and Technical Details. This appendix proves Theorem 2.4.1. When compared to earlier theoretical results in GFI, a major difference is that the current work considers complex-valued coefficients and responses. We begin by presenting three lemmas.

2.8.2.1. *Lemmas.*

LEMMA 2.8.1. *If* $\log j/\log p \to \delta$ *as* $p \to \infty$, *then* $\log \binom{p}{j} = j\log p(1-\delta)(1+o(1))$.

PROOF. First, calculate $\binom{p}{j} = \frac{p!}{j!(p-j)!} = \frac{p(p-1)\cdots(p-j+1)}{j!} = \frac{p^j(1-\frac{1}{p})(1-\frac{2}{p})\cdots(1-\frac{j-1}{p})}{j!}$ and we have

$$(1-\frac{j-1}{p})^{j-1} < (1-\frac{1}{p})(1-\frac{2}{p})\cdots(1-\frac{j-1}{p}) < (1-\frac{1}{p})^{j-1}.$$

By sterling's formula,

$$\sqrt{2\pi}j^{j+1/2}e^{\frac{-j+1}{12j+1}} < j! < \sqrt{2\pi}j^{j+1/2}e^{\frac{-j+1}{12j}}$$

so we have

$$\begin{aligned}
\log\tbinom{p}{j} \leq & j\log p + (j-1)\log\left(1-\frac{1}{p}\right) - \log j! \\
\leq & j\log p + (j-1)\log\left(1-\frac{1}{p}\right) - \left(j+\frac{1}{2}\right)\log j + j - \frac{1}{12j+1} - \log\sqrt{2\pi} \\
\leq & j\log p - \left(j+\frac{1}{2}\right)\log j + j \\
= & j\log p\left[1 - \frac{\left(j+\frac{1}{2}\right)\log j}{j\log p} + \frac{1}{\log p}\right] \\
= & j\log p(1-\delta)(1+o(1))
\end{aligned}$$

and

$$\begin{aligned}
\log \tbinom{p}{j} \geq & j \log p + (j-1) \log \left(1 - \frac{j-1}{p}\right) - \left(j + \frac{1}{2}\right) \log j + j - \frac{1}{12j} - \log \sqrt{2\pi} \\
\geq & j \log p + (j-1) \log \left(1 - \frac{j-1}{p}\right) - \left(j + \frac{1}{2}\right) \log j - \log \sqrt{2\pi} \\
= & j \log p \left(1 + \frac{(j-1) \log \left(1 - \frac{j-1}{p}\right)}{j \log p}\right) - j \log p \left(\frac{\left(j + \frac{1}{2}\right) \log j}{j \log p} - \frac{\log \sqrt{2\pi}}{j \log p}\right) \\
= & j \log p (1-\delta)(1 + o(1)),
\end{aligned}$$

which completes the proof. □

LEMMA 2.8.2. *Let χ_j^2 be a chi-square random variable with j degrees of freedom. If $c \to \infty$ and $\frac{J}{c} \to 0$, then*

$$P(\chi_j^2 > c) = \frac{1}{\Gamma(\frac{j}{2})} (\frac{c}{2})^{k/2-1} e^{-c/2} (1 + o(1))$$

uniformly over $j \leq J$.

PROOF. The pdf of χ_j^2 is $f(x) = \frac{1}{2^{\frac{j}{2}} \Gamma(\frac{j}{2})} x^{\frac{j}{2}-1} e^{-\frac{x}{2}}$, so

$$\begin{aligned}
P\left(x_j^2 > c\right) &= \int_c^\infty \frac{\left(\frac{1}{2}\right)^{\frac{j}{2}}}{\Gamma\left(\frac{j}{2}\right)} x^{\frac{i}{2}-1} e^{-\frac{x}{2}} dx \\
&= \frac{\left(\frac{1}{2}\right)^{\frac{j}{2}}}{\Gamma\left(\frac{j}{2}\right)} \int_c^{+\infty} x^{\frac{j}{2}-1} e^{-\frac{x}{2}} dx.
\end{aligned}$$

Now calculate

$$\begin{aligned}
\int_c^\infty x^{\frac{j}{2}-1} e^{-\frac{x}{2}} dx = & \int_c^\infty x^{\frac{j}{2}-1} (-2) de^{-\frac{x}{2}} \\
= & (-2) x^{\frac{j}{2}} - e^{-\frac{x}{2}} \Big|_c^\infty - \int_c^\infty e^{-\frac{x}{2}} (-2) \left(\frac{j}{2} - 1\right) x^{\frac{j}{2}-2} dx \\
= & (j-2) \int_c^{+\infty} x^{\frac{j}{2}-2} e^{-\frac{x}{2}} dx + 2c^{\frac{j}{2}-1} e^{-\frac{c}{2}}.
\end{aligned}$$

Therefore

$$F_j(c) = P\left(X_j^2 > c\right) = \frac{\left(\frac{1}{2}\right)^{\frac{j}{2}-1}}{\Gamma\left(\frac{j}{2}\right)} c^{\frac{j}{2}-1} e^{-\frac{c}{2}} + \frac{\left(\frac{1}{2}\right)^{\frac{j}{2}}}{\Gamma\left(\frac{j}{2}\right)} (j-2) \int_c^\infty x^{\frac{j}{2}-2} e^{-\frac{x}{2}} dx$$
$$= \frac{\left(\frac{1}{2}\right)^{\frac{j}{2}-1}}{\Gamma\left(\frac{j}{2}\right)} c^{\frac{j}{2}-1} e^{-\frac{c}{2}} + F_{j-2}(c).$$

So if j is even,

$$F_j(c) = \frac{1}{\Gamma\left(\frac{j}{2}\right)} \left(\frac{c}{2}\right)^{\frac{j}{2}-1} \cdot e^{-\frac{c}{2}} \left[1 + \sum_{i=1}^{\frac{j-2}{2}} \left(\frac{\left(\frac{j}{2}-1\right) \cdots \left(\frac{j}{2}-i\right)}{(c/2)^i} \right)\right]$$

and if j is odd,

$$F_j(c) = \frac{1}{\Gamma\left(\frac{j}{2}\right)} \left(\frac{c}{2}\right)^{\frac{j}{2}-1} \cdot e^{-\frac{c}{2}} \left[1 + \sum_{i=1}^{\frac{j-3}{2}} \left(\frac{\left(\frac{j}{2}-1\right) \cdots - \left(\frac{j}{2}-i\right)}{\left(\frac{c}{2}\right)^i} \right)\right] + F_1(m),$$

where

$$F_1(c) = P(\chi_1^2 \geq c) \approx 2\frac{\exp\left(-\frac{c}{2}\right)}{\sqrt{2\pi c}} = \frac{1}{\Gamma\left(\frac{j}{2}\right)} \left(\frac{c}{2}\right)^{\frac{j}{2}-1} e^{-\frac{c}{2}} \frac{2T\left(\frac{j}{2}\right)}{\sqrt{2\pi}\left(\frac{c}{2}\right)^{\frac{j-1}{2}}}.$$

Now when $c \to \infty$, we have

$$F_j(c) = \frac{1}{\Gamma\left(\frac{j}{2}\right)} \left(\frac{c}{2}\right)^{\frac{j}{2}-1} e^{-\frac{c}{2}} [1 + R(j,c)].$$

Finally, it is straightforward to see that $R(j,c) \leq R(J,c) \to 0$ as $c \to \infty$, which completes the proof. □

LEMMA 2.8.3. *Let χ_j^2 be a chi-square random variable with j degree of freedom. Let $c_j = 2j\log p + \log(j \log p)$. If $p \to \infty$, then for any $J \leq p$,*

$$\sum_{j=1}^{J} \binom{p}{j} P(\chi_j^2 > c_j) \to 0.$$

PROOF. Here we can directly apply Lemma 2.8.2. Let $q_j = \sqrt{\frac{c_j}{(j\log p)^2}}$, by using $\binom{p}{j} \le p^j$, we have

$$\begin{aligned}
\binom{p}{j} P\left(x_j^2 > c_j\right) &= \binom{p}{j} \frac{\left(\frac{c_j}{2}\right)^{\frac{j}{2}-1} e^{-\frac{c_j}{2}}}{\Gamma\left(\frac{j}{2}\right)} (1+O(1)) \\
&\le (c_j)^{\frac{j}{2}-1} \frac{p^j e^{-\frac{1}{2}\cdot 2j(\log p + \log(j\log p))}}{2^{\frac{j}{2}-1}\Gamma\left(\frac{j}{2}\right)} (1+o(1)) \\
&= (c_j)^{\frac{j}{2}-1} (1+o(1)) \cdot \frac{p^j e^{-j\log p - j\log(j\log p)}}{2^{\frac{j}{2}}-1\Gamma\left(\frac{j}{2}\right)} \\
&= (c_j)^{\frac{j}{2}-1} (1+o(1)) \cdot \frac{(j\log p)^{-j}}{2^{\frac{j}{2}-1}\Gamma(\frac{j}{2})}.
\end{aligned}$$

Let

$$q_j = \sqrt{\frac{c_j}{(j\log p)^2}} \le \frac{(c_j)^{\frac{j}{2}-1}}{(j\log p)^j}(1+o(1)) \le \frac{q_j^j}{c_j}(1+o(1))$$

and therefore

$$\sum_{j=1}^{J} \binom{p}{j} P(\chi_j^2 > c_j) \le \sum_{j=1}^{J} \frac{q_j^j}{c_j}(1+o(1)) \overset{q_j\to 0}{\longrightarrow} 0,$$

which completes the proof. □

2.8.2.2. *Proof of Theorem 2.4.1.* Denote $\mathcal{M}$ as the collection of models for which (2.16) holds. We shall prove that $\max_{\mathcal{M}} \frac{r(M)}{r(M_0)} \xrightarrow{P} 0$. Without loss of generality, we assume $\sigma^2 = 1$. We write $|M_0| = m_0$, $|M| = m$, where $m_0 = o(N)$ and $m = o(N)$. For simplicity, we can rewrite

$$\frac{r(M)}{r(M_0)} = \exp\{-T_1 - T_2 - T_3\},$$

where

$$T_1 = (N - m - \frac{1}{2}) \log \frac{\mathbf{RSS}_M}{\mathbf{RSS}_{M_0}},$$

$$\begin{aligned}
T_2 = &\frac{m-m_0}{2}\log n + (m-m_0)\log \pi \mathbf{RSS}_{M_0} \\
&+ \log\frac{\Gamma(N-m_0)}{\Gamma(N-m)} + \gamma \log\binom{K}{m} - \gamma\log\binom{K}{m_0}
\end{aligned}$$

and

$$T_3 = -\frac{1}{2}\log\frac{[\det(A_{M_0}^H A_{M_0})]}{[\det(A_M^H A_M)]}.$$

Next we consider the following two cases:

Case 1: $M_0 \not\subset M$.

Let $\mathcal{M}_{|} = \{M : |M| = j, M \in \mathcal{M}\}$. Notice that $\text{RSS}_{M_0} = (N - m_0)(1 + o_p(1)) = N(1 + o_p(1))$,

$$\text{RSS}_M - \text{RSS}_{M_0} = \Delta_M + 2\boldsymbol{\mu}^H(\boldsymbol{I} - \boldsymbol{P}_M)\boldsymbol{\epsilon} - \boldsymbol{\epsilon}^H\boldsymbol{P}_M\boldsymbol{\epsilon} + \boldsymbol{\epsilon}^H(\boldsymbol{I} - \boldsymbol{P}_{M_0})\boldsymbol{\epsilon}, \tag{2.21}$$

where $\boldsymbol{\mu} = \boldsymbol{A}_{M_0}\boldsymbol{\beta}_{M_0}$, $\Delta_M = ||(\boldsymbol{I} - \boldsymbol{P}_M)\boldsymbol{\mu}||^2$ and $\boldsymbol{\epsilon}^H\boldsymbol{P}_{M_0}\boldsymbol{\epsilon} = m_0(1 + o_p(1))$.

First consider the second term in (2.21) and denote $\boldsymbol{Z}_M = \boldsymbol{\mu}^H(\boldsymbol{I} - \boldsymbol{P}_M)\boldsymbol{\epsilon}/\sqrt{\Delta_M}$, we have

$$\boldsymbol{\mu}^H(\boldsymbol{I} - \boldsymbol{P}_M)\boldsymbol{\epsilon} = \sqrt{\Delta_M}\boldsymbol{Z}_M,$$

where $\boldsymbol{Z}_M \sim \mathcal{CN}(0, 1)$. Let $c_j = j\{\log K + \log j \log K\}$. For simplicity, we denote $c_{|M|}$ by c_m. Then by Lemma 2.8.3

$$\begin{aligned} P(\max_{\mathcal{M}} |\boldsymbol{Z}_M/\sqrt{c_m}| > 1) &\leq \sum_{j=1}^{bm_0} \sum_{\mathcal{M}_j} P(\boldsymbol{Z}_M^2 > c_j) \\ &= \sum_{j=1}^{bm_0} \tbinom{K}{j} P(\frac{\chi_2^2}{2} > c_j) \leq \sum_{j=1}^{bm_0} \tbinom{K}{j} P(\chi_j^2 > 2c_j) \to 0. \end{aligned}$$

Therefore,

$$|\boldsymbol{\mu}^H(\boldsymbol{I} - \boldsymbol{P}_M)\boldsymbol{\epsilon}| \leq \sqrt{\Delta_M}|\boldsymbol{Z}_M| \leq \sqrt{\Delta_M}\sqrt{c_M}(1 + o_p(1))$$

uniformly over $\mathcal{M}$. Since $c_m = o(m_0 \log K)$ and the identifiability condition (2.16) states $m_0 \log(K) = o_p(\Delta_M)$ uniformly over $\mathcal{M}$ s.t. $M_0 \not\subset M$,

$$|\boldsymbol{\mu}^H(\boldsymbol{I} - \boldsymbol{P}_M)\boldsymbol{\epsilon}| = o_p(\Delta_M).$$

Next, we consider the third term in (2.21). By Lemma 2.8.3 again, we have

$$P(\max_{\mathcal{M}} \boldsymbol{\epsilon}^H \boldsymbol{P}_M \boldsymbol{\epsilon}/c_m > 1) \leq \sum_{j=1}^{km_0} \sum_{\mathcal{M}_j} P(\boldsymbol{\epsilon}^H \boldsymbol{P}_M \boldsymbol{\epsilon} > c_j)$$
$$\leq \sum_{j=1}^{km_0} \binom{K}{j} P(\chi_j^2 > 2c_j) \to 0.$$

So $\boldsymbol{\epsilon}^H \boldsymbol{P}_M \boldsymbol{\epsilon} \leq c_m(1 + o_p(1))$, and $\boldsymbol{\epsilon}^H \boldsymbol{P}_M \boldsymbol{\epsilon} = o_p(\Delta_M)$ uniformly over $\mathcal{M}$ s.t. $M_0 \not\subset M$. Therefore

$$\text{RSS}_M - \text{RSS}_{M_0} = \Delta(M)(1 + o_p(1))$$

and we have

$$\log(\frac{\text{RSS}_M}{\text{RSS}_{M_0}}) = \log(1 + \frac{\text{RSS}_M - \text{RSS}_{M_0}}{\text{RSS}_{M_0}})$$
$$= \log(1 + \frac{\Delta(M)}{N}(1 + o_p(1)))$$

uniformly over all $M \in \mathcal{M}$ s.t. $M_0 \not\subset M$. Thus

$$T_1 = (N - m - \frac{1}{2}) \log(\frac{\text{RSS}_M}{\text{RSS}_{M_0}})$$
$$= (N - m - \frac{1}{2}) \log(1 + \frac{\Delta(M)}{N}(1 + o_p(1)))$$
$$= \frac{2N(o_p(1) + 1)}{2} \cdot \frac{\Delta(M)}{N}(1 + o_p(1)) = \Delta(M)(1 + o_p(1))$$

uniformly for $M \in \mathcal{M}$ such that $M_0 \not\subset M$.

Also,

$$(m - m_0) \log(\pi \text{RSS}_{M_0}) + \log \frac{\Gamma(N - m_0)}{\Gamma(N - m)}$$
$$= (m - m_0) \log N(1 + o_p(1)) + (m - m_0) \log N(1 + o_p(1))$$
$$= 2(m - m_0) \log N(1 + o_p(1)).$$

Finally, we have

$$T_2 = \frac{5}{2}(m - m_0) \log N(1 + o_p(1)) - \gamma \log \binom{p}{m_0} + \gamma \log \binom{K}{m}$$
$$\geq \frac{-5}{2} m_0 \log N(1 + o_p(1)) - \gamma m_0 \log K.$$

Case 2: Let $\mathcal{M}^* = \{M \in \mathcal{M}, M_0 \subset M, M \neq M_0\}$ and $\mathcal{M}_j^* = \{M, |M| = j, M_0 \subset M\}$. First notice that when $M_0 \subset M$, we have $\text{RSS}_{M_0} - \text{RSS}_M = \frac{1}{2}\chi^2_{2(m-m_0)}(M)$, where $\chi^2_{2(m-m_0)}(M)$ is a chi-square distribution with $2(m-m_0)$ degrees of freedom depending on M. Write $c_j = j\{\log K + \log(j \log K)\}$. Then by Lemma 2.8.3 we have

$$\begin{aligned}
&P(\max_{1\le j\le bm_0-m_0} \max_{M\in\mathcal{M}_j^*} \chi_j^2(M)/2c_j \ge 1) \\
&\le \sum_{j=1}^{bm_0-m_0} P(\max_{M\in\mathcal{M}_j^*} \ge 2c_j) \\
&= \sum_{j=1}^{bm_0-m_0} \tbinom{K-m_0}{j} P(\chi_j^2(M) \ge 2c_j) \\
&\le \sum_{j=1}^{bm_0-m_0} \tbinom{K}{j} P(\chi_j^2 \ge 2c_j) \to 0,
\end{aligned}$$

which implies

$$\chi^2_{2(m-m_0)}(M) \le 2c_{2(m-m_0)}(1+o_p(1)).$$

Notice that $2c_{2(m-m_0)} = o(N)$, and therefore

$$\begin{aligned}
&(N-m-\frac{1}{2})\log(\frac{\text{RSS}_m}{\text{RSS}_{M_0}}) \\
=&-(N-m-\frac{1}{2})\log(1+\frac{\frac{1}{2}\chi^2_{2(m-m_0)}(M)}{\text{RSS}_{M_0}-\frac{1}{2}\chi_{2(m-m_0)^2(M)}}) \\
\ge&(N-m-\frac{1}{2})\log(\frac{\chi_{2(m-m_0)^2(M)}}{2\text{RSS}_{M_0}-\chi^2_{2(m-m_0)}(M)}) \\
\ge&-\frac{c_{2(m-m_0)}}{2}(1+o_p(1)) \\
\ge&-2(m-m_0)[1+\frac{\log\{(bm_0-m_0)\log K\}}{\log K}]\log K(1+o_p(1)) \\
\ge&-2(m-m_0)(1+\delta)\log K(1+o_p(1))
\end{aligned}$$

uniformly over $\mathcal{M}^*$. Consequently, we have shown that

$$T_1 \ge -2(m-m_0)(1+\delta)\log K(1+o_p(1))$$

uniformly over $\mathcal{M}^*$. By Lemma 2.8.1, for $m_0 < m < bm_0$, we have

$$\log \binom{K}{m} = (1-\delta)m \log K(1+o(1))$$

uniformly over $\mathcal{M}^*$. So

$$T_2 = \frac{5}{2}(m-m_0)\log N(1-o_p(1)) + \gamma(1-\delta)(m-m_0)\log K(1+o(1))$$

uniformly over $\mathcal{M}^*$.

Finally, we have

$$\max_{M\neq M_0, M\in\mathcal{M}} \frac{r(M)}{r(M_0)} = \max\Big\{ \max_{M_0\not\subset M} \exp\left(-T_1-T_2-T_3\right), \max_{M_0\subset M} \exp(-T_1-T_2-T_3)\Big\}.$$

As $T_3 = -\frac{1}{2}\log\frac{[\det(A_{M_0}^H A_{M_0})]}{[\det(A_M^H A_M)]}$ and under the identifiability condition (2.16), where we only consider $|M| \leq b|M_0|$, we have $T_3 > -\infty$. Together with the above analysis, we have

$$\max_{M_0\not\subset M} \exp(-T_1-T_2-T_3) \xrightarrow{P} 0, \tag{2.22}$$

since $\min_{M_0\not\subset M}\{T_1+T_2+T_3\} \to \infty$. Also,

$$\max_{M_0\subset M} \exp(-T_1-T_2-T_3) \to 0, \tag{2.23}$$

as $\min_{M_0\subset M} T_1+T_2+T_3 \to \infty$ if $\gamma > \frac{1+\delta}{1-\delta} - \frac{5\eta}{2(1-\delta)}$, which is guaranteed by the assumption.

So (2.22) and (2.23) together show that

$$\max_{M\neq M_0, M\in\mathcal{M}^*} \frac{r(M)}{r(M_0)} \xrightarrow{P} 0.$$

□

Moreover, if condition (2.18) holds, we have

$$\sum_{M\neq M_0, M\in\mathcal{M}^*} \frac{r(M)}{r(M_0)} \leq \sum_{j=1}^{km_0}\sum_{\mathcal{M}^*} \frac{r(M)}{r(M_0)} \leq km_0 \max_{M\neq M_0, M\in\mathcal{M}^*} |M_j^*| \frac{r(M)}{r(M_0)} \xrightarrow{P} 0$$

which shows that $r(M_0) \xrightarrow{P} 0$ over the class $\mathcal{M}^*$.

CHAPTER 3

Structural Break Detection in Non-stationary Network Vector Autoregression Models

Imagine a network, like a social network or a system of connected devices, is being observed over time. Each node in this network has certain measurements attached to it that can change, like the temperature of a device. Although the overall structure of the network remains constant, these measurements can vary, leading to a complex multivariate time series dataset that exhibits non-stationary characteristics over time. This chapter applies a piecewise stationary network vector autoregressive (NAR) model to analyze these network data. The main idea is to partition the entire dataset into segments where the NAR model for each segment remains stationary. The identification of these segments, along with the determination of the NAR processes' autoregressive lag orders, are treated as unknowns. The minimum description length (MDL) principle is employed to develop a criterion for model selection that estimates these unknown parameters. A two-stage genetic algorithm is then formulated to tackle this optimization challenge. The MDL criterion is proven to be consistent in identifying the number and positions of the breakpoints - the junctures where adjacent NAR segments intersect. The effectiveness of the proposed method is demonstrated through simulation studies and real data analysis.

3.1. Introduction

Consider a network $A = (a_{i_1 i_2}) \in \mathbb{R}^{K \times K}$ with K nodes that may represent different relationships in different situations, such as people's social networks [47], companies' economic networks, and physical site networks. Let $a_{i_1 i_2} = 1$ if there exists some kind of relationship from node i_1 to node i_2; for example, followers and followees on social media. On the other hand, $a_{i_1 i_2} = 0$

if such a relationship does not exist. Also, further assume A cannot be self-related: $a_{i_l i_l} = 0$ for $i_l = 1, \ldots, K$. Such relationships can be either directed or undirected.

From the network A we can collect continuous measurements $X_{it} \in \mathbb{R}$ from node $i = 1, \ldots, K$ at time $t = 1, \ldots, T$. Denote

$$\boldsymbol{X}_t = (X_{1t}, \ldots, X_{Kt})^T \in \mathbb{R}^K, \quad t = 1, \ldots, T,$$

as the measurements from all K nodes in the whole network at time point t.

One of the earlier and widely used models for the $\boldsymbol{X}_t$'s is the vector autoregressive (VAR) model; e.g., see [48] and [49]. The VAR model introduces $O(K^2)$ parameters to handle the interactions amongst the nodes, but the estimation problem is tremendously large if K is large. Besides, there might be other exogenous covariates related to the nodes that also influence the $\boldsymbol{X}_t$'s; e.g., personal information in social networks and regional development level in economic networks. The VAR model unusually fails to include such information.

The network vector autoregressive (NAR) model was thus proposed by [50] to model the $\boldsymbol{X}_t$'s. It contains much fewer parameters that also utilize the observed network structure A and also allows possible exogenous covariates. Other time series models designed for networks include [51, 52].

For each node i, assume there exists a q dimensional node-specific exogenous covariates $\boldsymbol{V}_i = (V_{i1}, \cdots, V_{iq})^T \in \mathbb{R}^q$. As stated in [50] and [53], a NAR(p_1, p_2) model assumes the measurements X_{it}'s are influenced by self lags (past values), network lags (past values of "related" nodes), and node specified covariates effects, and is given by

$$X_{it} = \beta_0 + \boldsymbol{V}_i^\top \boldsymbol{\gamma} + \sum_{m=1}^{p_1} \alpha_m \sum_{j=1}^{K} \frac{a_{ij}}{n_i} X_{j(t-m)} + \sum_{n=1}^{p_2} \beta_n X_{i(t-n)} + \varepsilon_{it}, \tag{3.1}$$

where $n_i = \sum_{l \neq i} a_{il}$ is the total number of nodes that i follows, $\beta_0, \alpha_m \in \mathbb{R}$, $\beta_n \in \mathbb{R}$, and $\boldsymbol{\gamma} = (\gamma_1, \cdots, \gamma_q) \in \mathbb{R}^q$ are, respectively, the coefficients for the network lags, the self lags and the node specified covariates. Also, p_1 and p_2 are the lag orders for the network lags and the self

lags, respectively. The noise ε_{it} is assumed to follow a normal distribution $N(0, \sigma_i^2)$. Lastly, write $\boldsymbol{W} = \text{diag}\{n_1^{-1}, \cdots, n_K^{-1}\}\boldsymbol{A} = (\boldsymbol{w}_1, \cdots, \boldsymbol{w}_k)^T$ as the row-normalized network.

The NAR model has been successfully applied to solve problems in different areas, including social media analysis [50], air quality studies [53], and economic growth evaluations [54]. However, the vast majority of these studies assume the underlying process is stationary over the whole time span, which can be an unrealistic assumption for multivariate time series observed in many modern applications [55, 56].

One possible approach to mitigate this issue is to partition the whole process into a number of shorter, stationary processes. That is, a sequence of piecewise stationary NAR models is used to model the non-stationary series $\{\boldsymbol{X}_t\}_{t=1}^T$.

A precise formulation is as follows. Suppose there are m_0 breakpoints; i.e., $\{\boldsymbol{X}_t\}_{t=1}^T$ is partitioned into $m_0 + 1$ piecewise stationary NAR models. The m_0 breakpoint locations $\{\tau_j\}_{j=1}^{m_0}$ satisfy $0 < \tau_1 < \tau_2 < \ldots < \tau_{m_0} < T + 1$, and for convenience, write $0 = \tau_0$ and $\tau_{m_0+1} = T + 1$. For all $j = 1, \ldots, m_0 + 1$, it is assumed that the j-th segment, $\{\boldsymbol{X}_t\}$ with $\tau_{j-1} \leq t < \tau_j$, follows a stationary NAR$(p_{1,j}, p_{2,j})$ model. It is also assumed that the network structure remains unchanged in all segments. Similar to (3.1), the j-th segment is modeled as

$$X_{it,j} = \beta_{0,j} + \boldsymbol{V}_i^\top \boldsymbol{\gamma}_j + \sum_{m=1}^{p_{1,j}} \alpha_{m,j} \sum_{l=1}^{K} \frac{a_{il}}{n_i} X_{l(t-m)} + \sum_{n=1}^{p_{2,j}} \beta_{n,j} X_{i(t-n)} + \varepsilon_{it,j}. \tag{3.2}$$

Throughout the chapter, we follow the same stationarity assumptions in [50] for each segment, for example, $\sum_{i=1}^{p_{1,j}}(|\alpha_i| + |\beta_i|) <= 1$ is satisfied which guarantees the piecewise stationarity. With the above piecewise NAR model, one needs to estimate the number m_0 and the locations $\{\tau_1, \cdots, \tau_{m_0}\}$ of the breakpoints. One also needs to estimate the model parameters in each segment, including the lag orders $p_{1,j}$ and $p_{2,j}$, and the regression coefficients $\beta_{0,j}$, $\alpha_{m,j}$, and $\beta_{n,j}$. It will be shown below that for each segment, once the lag orders are determined, the regression coefficient estimates can be obtained using maximum likelihood [50]. So the main challenge is to estimate the number and locations of the breakpoints, as well as the lag orders in each segment.

Notice that this can be seen as a statistical model selection problem, as different m_0 would lead to different piecewise stationary models with different numbers of model parameters.

One major contribution of this chapter is the development of a systematic method for selecting a best-fitting piecewise stationary NAR model (3.2). That is, to estimate the number and locations of the breakpoints, as well as the orders for each stationary NAR model between any two adjacent breakpoints. Once these quantities are estimated, the remaining model parameters can be estimated using maximum likelihood. The proposed method invokes the minimum description length (MDL) principle [29,57] to derive an objective criterion for model selection, and uses the genetic algorithm to solve the corresponding optimization problem.

Breakpoint detection in network problems has been widely investigated in recent years, for example, in the medical area [58]. Existing mainstream methods can be broadly divided into two categories. The first group of methods begins with summarizing a certain characteristic of each of the networks with a metric and then detects any possible breakpoints with respect to that metric. Examples of a network metric include various matrix norms [59,60] and centrality metrics [61,62]. Reducing a (complicated) network to a simple metric typically provides substantial speed gain, but at the same time, it may inevitably cause information loss, which in turn may adversely affect the final results. The second group of methods fits a dynamic network model to the data and uses model-based testing methods to detect breakpoints. Examples of such a network model include generalized hierarchical random graphs [63], Kronecker product graphs [64], and stochastic block models [65]. Some of these model assumptions could be restrictive, but if appropriate for the data at hand, these methods tend to provide excellent results.

One merit of the proposed method is that no strong restrictions are imposed on the network structures, which greatly increases the applicability of the method. To the best of the authors' knowledge, this is one of the first complete systematic studies that consider structural break estimation in non-stationary NAR models.

The rest of this chapter is organized as follows. Section 3.2 derives the MDL criterion for estimating the unknowns in the piecewise stationary NAR model. It also studies the theoretical

properties of the criterion. Section 3.3 develops a two-stage GA algorithm to minimize the MDL criterion. The empirical performance of the proposed method is illustrated in Section 3.4 via various numerical simulations and in Section 3.5 via an application to some real Manhattan yellow cab data. Lastly, concluding remarks are offered in Section 3.6, while technical details and additional simulation results are provided in the supplementary material.

3.2. Breakpoint Detection using MDL

The MDL principle is a popular method for deriving an effective model selection criterion. It defines the best-fitting model as the one that compresses the data into the shortest possible code length for storage, where the code length represents the bites needed to store the data. It was proposed by Rissanen [29, 57] and has been successfully applied to solve various model selection problems such as image segmentation [66], network constructions [67, 68, 69], and quantile and spline regression [70, 71]. This chapter focuses on the so-called "Two–Part MDL" [72], and this section derives the corresponding MDL criterion for fitting a piecewise stationary NAR model.

3.2.1. Derivation of the MDL Criterion. To store the observed data, one can split them into two parts: the first part is a fitted model and the second part is the corresponding residuals. If the fitted model is a good model, it will be more economical to store the data in this way. Denote $\mathrm{CL}(z)$ as the code length of any object z; thus, we want to minimize CL("data"). Also, denote the whole class of piecewise NAR models as $\mathcal{M}$, denote any model in $\mathcal{M}$ as $\mathcal{F} \in \mathcal{M}$ and its corresponding residuals as $\hat{\mathcal{E}}$. Then we have

$$\mathrm{CL}(\text{"data"}) = \mathrm{CL}(\text{"fitted model"}) + \mathrm{CL}(\text{"residuals"}) = \mathrm{CL}(\mathcal{F}) + \mathrm{CL}(\hat{\mathcal{E}}|\mathcal{F}).$$

We need a computable expression for CL("data") and we first calculate $\mathrm{CL}(\mathcal{F})$. Notice that to completely specify a model $\mathcal{F}$, we need to know the breakpoint number m and their locations $\mathcal{T} = \{\tau_1, \cdots, \tau_m\}$. In addition, for all $j = 1, \ldots, m+1$, we need to know the lag orders $\boldsymbol{p}_j = (p_{1,j}, p_{2,j})$ and regression parameters $\boldsymbol{\theta}_j = (\beta_{0,j}, \alpha_{1,j}, \cdots, \alpha_{p_{1,j},j}, \beta_{1,j}, \cdots, \beta_{p_{2,j},j}, \boldsymbol{\gamma}_j)$, for the j-th segment. Write $\mathcal{P} = (\boldsymbol{p}_1, \cdots, \boldsymbol{p}_{m+1})$ and $\hat{\Theta} = (\hat{\boldsymbol{\theta}}_1, \cdots, \hat{\boldsymbol{\theta}}_{m+1})$. Then we have $\mathcal{F} = (m, \mathcal{T}, \mathcal{P}, \hat{\Theta})$,

which leads to the code length decomposition:

$$\text{CL}(\mathcal{F}) = \text{CL}(m) + \text{CL}(\mathcal{T}) + \text{CL}(\mathcal{P}) + \text{CL}(\hat{\Theta}). \tag{3.3}$$

When applying MDL, the code length of an unknown positive integer I can be approximated by $\log(I)$ [57]. On the other hand, if I is known to be upper-bounded by I_u, then its code length is $\log(I_u)$. So the first three terms on the RHS of (3.3) are

$$\text{CL}(m) = \log(m+1), \tag{3.4}$$

$$\text{CL}(\mathcal{T}) = (m+1)\log(T), \tag{3.5}$$

$$\text{CL}(\mathcal{P}) = \sum_{j=1}^{m+1} \{\log(p_{1,j}) + \log(p_{2,j})\}, \tag{3.6}$$

where the additional 1 in $\text{CL}(m)$ is used to make the formula meaningful when $m = 0$.

For the last term in (3.3), we need to first estimate $\hat{\Theta}$ from model (3.2) and then encode the resulting estimated values. For estimation, we shall use the maximum likelihood method of [50], while for encoding, we shall use the result of [57] that any (scalar) maximum likelihood estimate calculated from N observations can be effectively encoded with $\frac{1}{2}\log(N)$ bits. We first describe the maximum likelihood method of [50].

Let $\boldsymbol{w}_i = (a_{il}/n_i : 1 \le l \le K)^T \in \mathbb{R}^K$ be the i-th row vector of the row normalized network matrix $\boldsymbol{W}$, and

$$Z^*_{l(t-1),j} := \{1, \boldsymbol{w}_l^T \boldsymbol{X}_{t-1,j}, \ldots, \boldsymbol{w}_l^T \boldsymbol{X}_{t-p_{1,j},j}, X_{l(t-1),j}, \ldots, X_{l(t-p_{2,j}),j}, \boldsymbol{V}_l^T\}^T \in \mathbb{R}^{p_{1,j}+p_{2,j}+q+1},$$

where $X_{l(t-1),j}$ represents the l-th element of $\boldsymbol{X}_{t-1,j}$. Let

$$\boldsymbol{Z}^*_{t-1,j} := (Z^*_{1(t-1),j}, \cdots, Z^*_{K(t-1),j})^T \in \mathbb{R}^{K\times(p_{1,j}+p_{2,j}+q+1)}.$$

Then the j-th segment, which is a NAR$(p_{1,j}, p_{2,j})$ model (see (3.2)), can be rewritten in vector form as

$$\boldsymbol{X}_{t,j} = \boldsymbol{Z}^*_{t-1,j}\boldsymbol{\theta}_j + \boldsymbol{\varepsilon}_j, \tag{3.7}$$

where $\boldsymbol{\varepsilon}_j \sim N_k(\boldsymbol{0}, \sigma_j^2 \boldsymbol{I}_k)$. Here the variances do not need to be the same for the proposed method to work, but for simplicity, below we will assume they are identical. With this, the maximum likelihood estimator of $\boldsymbol{\theta}_j$ is

$$\hat{\boldsymbol{\theta}}_j = \left(\sum_{t=\tau_{j-1}+p_{\max,j}+1}^{\tau_j} \boldsymbol{Z}^{*T}_{t-1,j} \boldsymbol{Z}^*_{t-1,j} \right)^{-1} \times \sum_{t=\tau_{j-1}+p_{\max,j}+1}^{\tau_j} \boldsymbol{Z}^*_{t-1,j} \boldsymbol{X}_{t,j} \in \mathbb{R}^{(p_{1,j}+p_{2,j}+q+1)}, \tag{3.8}$$

where $p_{\max,j} := \max(p_{1,j}, p_{2,j})$, $n_j := \tau_j - \tau_{j-1}$, and

$$\hat{\sigma}_j^2 = \frac{\sum_{t=\tau_{j-1}+p_{\max,j}+1}^{\tau_j} (\boldsymbol{X}_{t,j} - \boldsymbol{Z}^*_{t-1,j}\hat{\boldsymbol{\theta}}_j)^T (\boldsymbol{X}_{t,j} - \boldsymbol{Z}^*_{t-1,j}\hat{\boldsymbol{\theta}}_j)}{K(n_j - p_{\max,j})}. \tag{3.9}$$

As mentioned before, to encode a scalar maximum likelihood estimate, the code length is $\frac{1}{2}\log(N)$ if N observations were used for estimation. Therefore,

$$\mathrm{CL}(\hat{\Theta}) = \sum_{j=1}^{m+1} \frac{p_{1,j} + p_{2,j} + q + 1}{2} \log(n_j). \tag{3.10}$$

The last term in (3.3) that we need to calculate is $\mathrm{CL}(\hat{\mathcal{E}}|\mathcal{F})$, which equals the negative log (base 2) of the likelihood of the fitted model $\mathcal{F}$ [57]. From (3.7), (3.8) and (3.9), we have

$$\mathrm{CL}(\hat{\mathcal{E}}|\mathcal{F}) = \sum_{j=1}^{m+1} \left[\frac{K(n_j - p_{\max,j})}{2} \left\{ \log(2\pi\hat{\sigma}_j^2) + 1 \right\} \right] \log_2 e \tag{3.11}$$

Combining (3.4), (3.5), (3.6), (3.10) and (3.11) and using logarithm base e instead of base 2, (3.3) becomes

$$
\begin{aligned}
&\text{CL}(\text{“data”}) = \log(m+1) + (m+1)\log(T) \\
&+ \sum_{j=1}^{m+1} \left(\log(p_{1,j}) + \log(p_{2,j}) + \frac{p_{1,j} + p_{2,j} + q + 1}{2} \log(n_j) \right) \\
&+ \sum_{j=1}^{m+1} \left\{ \frac{K(n_j - p_{\max,j})}{2} \left(\log(2\pi\hat{\sigma}_j^2) + 1 \right) \right\} \log_2 e \\
&:= \text{MDL}(m, \tau_1, \cdots, \tau_m, p_{1,1}, p_{2,1}, \cdots, p_{1,m+1}, p_{2,m+1}).
\end{aligned} \tag{3.12}
$$

Thus, the MDL principle suggests that the best-fitting model for the observed data $\boldsymbol{X}_{t,j}, t = 1, \ldots, n_j, j = 1, \ldots, m$ is the one $\mathcal{F} \in \mathcal{M}$ that minimizes (3.12).

3.2.2. Theoretical Properties. Denote the true number of breakpoints as m_0 and the true locations of the breakpoints as $\mathcal{T}_0 = \{\tau_1^0, \cdots, \tau_{m_0}^0\}$. Define the true relative breakpoint locations as $\boldsymbol{\lambda}_0 = \{\lambda_1^0, \cdots, \lambda_{m_0}^0\}$ with $\tau_j^0 = \lfloor \lambda_j^0 T \rfloor$ for $j = 1, \ldots, m_0$, where $\lfloor x \rfloor$ represents the greatest integer that is less than or equal to x. Further, write $\boldsymbol{p} = (p_{1,1}, p_{2,1}, \cdots, p_{1,m+1}, p_{2,m+1})$ and $\boldsymbol{\lambda} = (\lambda_1, \cdots, \lambda_m)$. Note that the theoretical results in this subsection will be presented in terms of $\boldsymbol{\lambda}$ instead of $\mathcal{T}$.

As suggested by [73], for each segment, a sufficient number of data points are required to adequately estimate the corresponding NAR model parameters. For this reason, we impose the following constraint on the estimate of $\boldsymbol{\lambda}$. First, choose $\xi > 0$ sufficiently small enough that $\xi \ll \min_{i=1,\cdots,m_0+1}(\lambda_i^0 - \lambda_{i-1}^0)$. Then define

$$
A_m = \{ (\lambda_1, \ldots, \lambda_m \; 0 = \lambda_0 < \lambda_1 < \cdots < \lambda_m < \lambda_{m+1} = 1, \lambda_i - \lambda_{i-1} \geq \xi, i = 1, 2, \ldots, m+1 \}
$$

Lastly, we require the estimate of $\boldsymbol{\lambda}$ to be an element of A_m.

Using this constraint and (3.12), the unknown meta-parameters are given by

$$
\{\hat{m}, \hat{\boldsymbol{\lambda}}, \hat{\boldsymbol{p}}\} = \underset{m, \boldsymbol{p}, \boldsymbol{\lambda} \in A_m}{\arg\min} \frac{2}{T} \text{MDL}(m, \boldsymbol{\lambda}, \boldsymbol{p}). \tag{3.13}
$$

THEOREM 3.2.1. *For the piecewise stationary NAR model given by (3.2), when the true number of breakpoints m_0 is known, the estimate $\hat{\boldsymbol{\lambda}}$ defined by (3.13) satisfies*

$$\hat{\lambda}_j \xrightarrow{a.s.} \lambda_j^0, \quad j = 1, \ldots, m_0.$$

COROLLARY 3.2.1.1. *If the number of breakpoints m_0 is unknown and estimated with (3.13), then*

(1) The estimated number of breakpoints $\hat{m} \geq m_0$ for sufficient large T.

(2) When $\hat{m} > m_0$, for any $\lambda_j^0 \in \boldsymbol{\lambda}_0$, there exists a $\hat{\lambda}_k$ such that $|\lambda_j^0 - \hat{\lambda}_k| < \epsilon$, $\forall \epsilon > 0$ for large enough T.

(3) The lag order of the model in each segment cannot be underestimated; i.e., $\hat{p}_{1,j} \geq p_{1,j}^0$, $\hat{p}_{2,j} \geq p_{2,j}^0$, where $p_{1,j}^0$ and $p_{2,j}^0$ are the true lag orders.

If Assumption 1 below is satisfied, a consistency result of the MDL estimator (3.13) can be derived even when m_0 is not known.

ASSUMPTION 1. *For $j = 1, \cdots, m+1$, any fixed p_j and any sequence $\{g(T)\}_{T \leq 1}$ of integers that satisfies $g(T) \leq cT^{0.5}$ for some $c > 0$ when T is sufficiently large. Let $f_{\boldsymbol{p_j}}(\boldsymbol{X}_{i,j}|\boldsymbol{X}_{s,j}, s < i; \boldsymbol{\theta}_j)$ be the conditional density function of the i-th observation in the j-th segment. Also let $l_j(\boldsymbol{p}_j, \boldsymbol{\theta}_j, \boldsymbol{X}_{i,j}|\boldsymbol{X}_{s,j}, s < i) = \log f_{\boldsymbol{p}_j}(\boldsymbol{X}_{i,j}|\boldsymbol{X}_{s,j}, s < i; \boldsymbol{\theta_j})$ be the conditional log-likelihood function for $\boldsymbol{X}_{i,j}$. then*

$$\frac{1}{g(T)} \sum_{i=T-g(T)+1}^{T} l_j(\boldsymbol{p}_j, \boldsymbol{\theta}_j, \boldsymbol{X}_{i,j}|\boldsymbol{X}_{s,j}, s < i) \xrightarrow{a.s.} E(l_j(\boldsymbol{p}_j, \boldsymbol{\theta}_j, \boldsymbol{X}_{1,j}|\boldsymbol{X}_{s,j}, s < 1))$$

and

$$\frac{1}{g(T)} \sum_{i=T-g(T)+1}^{T} l_j'(\boldsymbol{p}_j, \boldsymbol{\theta}_j, \boldsymbol{X}_{i,j}|\boldsymbol{X}_{s,j}, s < i) \xrightarrow{a.s.} E(l_j'(\boldsymbol{p}_j, \boldsymbol{\theta}_j, \boldsymbol{X}_{1,j}|\boldsymbol{X}_{s,j}, s < 1))$$

where $l_j'(\boldsymbol{p}_j, \boldsymbol{\theta}_j, \boldsymbol{X}_{i,j}|\boldsymbol{X}_{s,j}, s < i)$ is the first derivative of $l_j(\boldsymbol{p}_j, \boldsymbol{\theta}_j, \boldsymbol{X}_{i,j}|\boldsymbol{X}_{s,j}, s < i)$.

This assumption is needed to control the effects at the two ends of the fitted segments so that the convergence rate of the location estimator can be established.

THEOREM 3.2.2. *For the piecewise stationary NAR model given by* (3.2)*, under the assumptions of Theorem 3.2.1(except for the known number of breakpoints) and Assumption 1, the estimator* $\{\hat{m}, \hat{\boldsymbol{\lambda}}\}$ *defined by (3.13) satisfies*

$$\hat{m} \xrightarrow{a.s.} m_0, \quad \hat{\boldsymbol{\lambda}} \xrightarrow{a.s.} \boldsymbol{\lambda}^{\mathbf{0}}.$$

The proofs of Theorem 3.2.1, Corollary 3.2.1.1, and Theorem 3.2.2 can be found in the supplementary material.

3.3. Practical Optimization of MDL Using Genetic Algorithms

The enormous searching space makes the minimization of (3.12) or (3.13) a non-trivial task. This section develops a genetic algorithm (GA) for solving this problem.

3.3.1. A Brief Introduction to Genetic Algorithms. GA is a search heuristic that can be dated back as early as [74], for which the main idea was inspired by Charles Darwin's theory of natural evolution. Typically, a GA begins with generating an initial set of possible solutions (*chromosomes*) to the optimization problem of interest, which is represented by vector form. Next, these chromosomes are weighted sampled as parents to generate their "offspring": parent chromosomes with better values for the optimization problem (i.e., larger values for maximization problems or smaller values for minimization problems) have higher chances of being chosen. An offspring chromosome is then produced by applying either a *crossover* or a *mutation* operation to the chosen parent chromosomes. Such a process repeats until some stopping criteria are met.

As suggested in [73], to preserve the evolution direction towards the optimal value, the best chromosome from the previous generation is preserved to replace the worst chromosome of the current generation. This process is known as the *elitist* step and guarantees the monotonously of the algorithm.

To speed up the algorithm, [75] introduced an island model version that is particularly suited for parallel computing. Rather than running with only one group of evolving chromosomes, the island model can simultaneously run NI (number of islands) subgroups of chromosomes. Periodically, chromosomes are allowed to migrate amongst the islands, a process known as *migration*. The migration policy that we use here is the same as in [73]. The purpose of migration is to avoid sub-optimal solutions for the subgroups. At every M_i-th generation, the worst M_N chromosomes in j-th island are replaced by the best M_N chromosomes in $(j-1)$-th island, for $j = 2, \ldots, NI$. The first island's worst M_N chromosomes are replaced by the best M_N chromosomes in the NI-th island.

3.3.2. Implementation Details. This subsection provides details of the tailored GA that we use to minimize (3.12) for the piecewise stationary NAR model (3.2).

3.3.2.1. *Chromosome representation.* In general, the representation of chromosomes plays an important role in the overall performance of GAs. A good representation should contain all the needed information of any potential solution for the calculation of (3.12). For the current problem, it suffices to include only the breakpoints $\mathcal{T}$ and the lag orders $\boldsymbol{p}$, as once these quantities are specified, the remaining unknown parameters can be uniquely calculated. Given this, we propose using the following constant-length representation for a chromosome $\boldsymbol{\delta} = (\delta_1, \cdots, \delta_T)$, where the gene values are

$$\begin{cases} \delta_t = -1 & \text{if time } t \text{ is not a breakpoint} \\ \delta_t = (p_{1,j}, p_{2,j}) & \text{if } t = \tau_{j-1} \text{ (i.e., time } t \text{ is the } (j-1)\text{-th} \\ & \text{breakpoint) and the } j\text{-th segment is a} \\ & \text{NAR } (p_{1,j}, p_{2,j}) \text{ model.} \end{cases} \tag{3.14}$$

If t is a breakpoint location, the t-th gene consists of two values, even though together they only use one gene index. This allows the length of the chromosomes to remain constant T irrespective of the number of breakpoints, which in turn facilitates the execution of the crossover and mutation operations.

3.3.2.2. *Maximum lag order and minimum span.* In practice we set the maximum possible lag order as $P_0 = 10$; i.e., $(p_{1,j}, p_{2,j}) \leq P_0$ for all j. Also, as mentioned before, we require each segment to have a minimum number of data points so that reasonable parameter estimates can be obtained. This requirement is called the minimum span constraint by [73]. For our problem, the minimum span $m_{p_{\max}}$ of a segment with a maximum lag order $p_{\max,j} = \max(p_{1,j}, p_{2,j})$ can be found in Table 3.1.

TABLE 3.1. Minimum number of data points required for different $p_{\max}$.

$p_{\max}$	0-1	2	3	4	5	6	7	8	9-10	11-20
$m_{p_{\max}}$	10	12	14	16	18	20	22	24	25	50

3.3.2.3. *Generating the first generation chromosomes.* The way we generate the first-generation chromosomes is summarized in Algorithm 1. We denote a pre-specified parameter r_G as the probability for any time point t to be a breakpoint. See section B of the supplementary material for other methods for generating the first-generation chromosomes.

Once the first generation is available, we select parent chromosomes from it and apply the crossover and mutation operations to produce offspring chromosomes. We denote the pre-specified probability for performing a crossover operation as r_C, and the probability for mutation is $1 - r_C$.

3.3.2.4. *Crossover.* In the crossover operation, one offspring chromosome is generated in a manner that is summarized by Algorithm 2.

3.3.2.5. *Mutation.* During mutation, one offspring chromosome is produced by Algorithm 3.

3.3.2.6. *Stopping Criterion.* As mentioned in the previous section, the island model will be used, which allows migration after every M_i generation. The algorithm will finish if the chromosome with the smallest MDL value does not change for M_C consecutive migrations or if the total number of migrations exceeds an upper bound M_U. The chromosome with the smallest MDL value will be taken as the final solution provided by the algorithm.

Algorithm 4 provides an overall summary that links all the above ingredients of the genetic algorithm.

Algorithm 1: Initialization

Input: *Minimum span for different lag order:* $m_{p_{\max}}$*;*
Probability of being a breakpoint: r_G*;*
The upper bound of lag orders: P_0*;*
Output: *Initialization chromosome* δ
initialize $\delta = (\delta_1, \cdots, \delta_T) = (-1, \cdot, -1)$*;* $i = 1$
while $i \leq T$ **do**
  Generate $r \sim Uniform(0, 1)$
  if $i == 1$ **then**
    SAMPLE $p_{1,i}, p_{2,i} \sim [P_0]$*;*
    SET $\delta_i = (p_{1,i}, p_{2,i})$*;* $p_{\max,i} = \max(p_{1,i}, p_{2,i})$*;*
    $i = i + m_{p_{\max,i}} + 1$
  else
    if $r < r_G$ **then**
      SAMPLE $p_{1,i}, p_{2,i} \sim [P_0]$*;*
      SET $\delta_i = (p_{1,i}, p_{2,i})$*;* $p_{\max,i} = \max(p_{1,i}, p_{2,i})$*;*
      $i = i + m_{p_{\max,i}} + 1$
    else
      $i = i + 1$
    end
  end
end

3.3.2.7. *Refined estimates for the lag orders.* Although the above GA provides good results in estimating the number and locations of the breakpoints, the estimated locations are not always equal to the true ones. This could have negative impacts on the estimation of the lag orders if the estimated segment contains data points from its adjacent segments.

As Corollary 3.2.1.1 states, although the estimated breakpoint locations are not necessarily to be exact, the true locations will likely be within some neighborhoods of the estimated ones. In light of this, when we estimate the lag orders $(p_{1,j}, p_{2,j})$ of the j-th estimated segment $[\hat{\tau}_{j-1}, \hat{\tau}_j)$, we only use data from $[\hat{\tau}_{j-1} + R_n, \hat{\tau}_j - R_n)$, where R_n is a calculated radius of the neighborhood, which can be calculated using the similar way as in [76]. The simulation results below show that this improves the estimation of the lag orders. We also only use those data from the shortened segment to estimate the regression parameters.

Figure 3.1 shows the flowchart of this two-stage genetic algorithm.

Algorithm 2: Crossover

Input: *Chromosomes of last generation;*
MDL values for last generation chromosomes;
Minimum span for different lag order: $m_{p_{\max}}$*;*
The upper bound of lag orders: P_0*;*

Output: *New generation chromosome* δ_{new}

initialize $\delta_{new} = (\delta_{new,1}, \cdots, \delta_{new,T}) = (-1, \cdot, -1)$*;* $i = 1$*;*
Weight Sample 2 parent chromosomes $\delta_{p_1}, \delta_{p_2}$ *based on their inverse MDL values;*
Set $i = 1$.
while $i \leq T$ **do**
  Generate $r \sim Uniform(0, 1)$
  if $r \leq 0.5$ **then**
    SET $\delta_{new,i} = \delta_{p_1,i}$*;*
    if $\delta_{new,i} \neq -1$ **then**
      SET $p_{\max,i} = \max(\delta_{new,i}[0], \delta_{new,i}[1])$*;*
      $i = i + m_{p_{\max,i}} + 1$
    else
      $i = i + 1$
    end
  else
  end
end

3.4. Simulation Results

3.4.1. General Parameter Settings. We first specify the parameter values that we used in the GA in all simulation settings: upper bound of lag order $P_0 = 10$; number of islands $NI = 40$; number of chromosomes in each island $n_m = 40$; migration frequency $M_i = 5$; migration numbers $M_N = 2$; stopping criterion $M_C = 20$, $M_U = 100$; initialization probability $r_G = 0.1$; crossover probability $r_C = 0.9$; mutation probabilities $r_P = r_N = 0.3$; neighbourhood radius $R_n = 0.5 \log(K) \log(T)$ as suggested in [76].

Three network structures from [50] are considered in each of the five simulation scenarios below:

I. Power-Law Distribution Structure: this structure mimics the phenomenon when a majority of nodes have very few edges while a few nodes have enormous numbers of edges. A discrete power-law distribution is used to generate in-degree $d_i = \sum_{j \neq i} a_{ji}$, where

Algorithm 3: Mutation

Input: *Chromosomes of last generation;*
MDL values for last generation chromosomes;
The probability of a gene mutating to non-breakpoint r_N*;*
The probability of a gene inherent from parent chromosome r_P*;*
Minimum span for different lag order: $m_{p_{\max}}$*;*
The upper bound of lag orders: P_0*;*

Output: *New generation chromosome* δ_{new}

initialize $\delta_{new} = (\delta_{new,1}, \cdots, \delta_{new,T}) = (-1, \cdot, -1)$*;* $i = 1$*;*

Weight Sample 1 parent chromosome δ_{p_1} *based on their inverse MDL values;*

Set $i = 1$.

while $i \leq T$ **do**
  Generate $r \sim Uniform(0, 1)$
  if *i == 1* **then**
    if $r < r_P$ **then**
      SET $\delta_{new,i} = \delta_{p_1,i}$*;*
      if $\delta_{new,i} \neq -1$ **then**
        SET $p_{\max,i} = \max(\delta_{new,i}[0], \delta_{new,i}[1])$*;*
        $i = i + m_{p_{\max,i}} + 1$
      else
        $i = i + 1$
      end
    else
      SAMPLE $p_{1,i}, p_{2,i} \sim [P_0]$*;*
      SET $\delta_i = (p_{1,i}, p_{2,i})$*;* $p_{\max,i} = \max(p_{1,i}, p_{2,i})$*;*
      $i = i + m_{p_{\max,i}} + 1$
    end
  else
    if $r < r_P$ **then**
      SET $\delta_{new,i} = \delta_{p_1,i}$*;*
      if $\delta_{new,i} \neq -1$ **then**
        SET $p_{\max,i} = \max(\delta_{new,i}[0], \delta_{new,i}[1])$*;*
        $i = i + m_{p_{\max,i}} + 1$
      else
        $i = i + 1$
      end
    else if $r > r_G + r_N$ **then**
      SAMPLE $p_{1,i}, p_{2,i} \sim [P_0]$*;*
      SET $\delta_i = (p_{1,i}, p_{2,i})$*;* $p_{\max,i} = \max(p_{1,i}, p_{2,i})$*;*
      $i = i + m_{p_{\max,i}} + 1$
    $\delta_{new,i} = -1$*;*
    $i = i + 1$
  end
end

Algorithm 4: Genetic algorithm for solving (3.13)

Input: *Minimum span for different lag order:* $m_{p_{\max}}$*;*
Probability of Crossover: r_C*;*
Probability of being a breakpoint: r_G*;*
The probability of a gene mutating to non-breakpoint r_N*;*
The probability of a gene inherent from parent chromosome r_P*;*
The upper bound of lag orders: P_0*;*

Output: *Final chromosome* δ_{final}

Initialize chromosomes using Algorithm 1

while *minimum MDL value changes* **do**
  Generate $r \sim$ *Uniform*$(0, 1)$
  if $r < r_C$ **then**
    Generate next generation using Crossover Algorithm 2
  else
    Generate next generation using Mutation Algorithm 3
  end
end

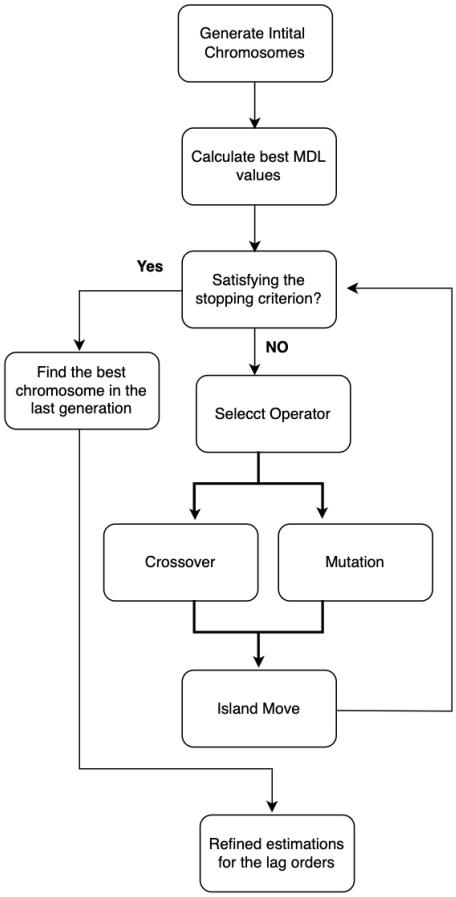


FIGURE 3.1. Flowchart for the two-stage genetic algorithm.

$P(d_i = x) = cx^{-1.2}$ with a constant c set to $c = 1.5$. Then for each node i, randomly select d_i nodes to follow it.

II. Dyad Independence Structure: A dyad is defined as a pair of nodes $D_{ij} = (a_{ij,a_{ji}})$, $1 \leq i < j \leq K$ and it is assumed that different dyads are independent. We set $P(D_{ij} = (1,1)) = 0.1$, $P(D_{ij} = (1,0)) = P(D_{ij} = (0,1)) = 0.05$, and $P(D_{ij} = (0,0)) = 0.85$.

III. Stochastic Block Structure: Randomly assign each node a block label uniformly from 5 groups; i.e., $\{1, \ldots, 5\}$. We set $P(a_{ij} = 1) = 0.15$ if $\{i, j\}$ belong to the same group, and $P(a_{ij} = 1) = 0.015$ if $\{i, j\}$ belong to different groups. This implies that nodes within the same group will have higher chances of being connected.

The node specific covariates are generated as $\boldsymbol{V}_i = 0.15\boldsymbol{Z}_i$, where $\boldsymbol{Z}_i = (Z_{i1}, \ldots, Z_{i4})^T \in \mathbb{R}^4$ is from a multivariate normal distribution with $\mathcal{N}_4(\mathbf{0}, \boldsymbol{\Sigma}_Z)$ with $\boldsymbol{\Sigma}_Z = (\sigma_{j_1 j_2}) = (0.5^{|j_1 - j_2|})$. For all different scenarios below, we set the number of time points $T = 300$, the number of nodes $K = 20$, and the number of simulation runs to 100.

3.4.2. Scenario 1: Impact of the refining step of Section 3.3.2.7. In this first scenario, breakpoints are set as $\mathcal{T} = (\frac{T}{3}, \frac{2T}{3}) = (100, 200)$, with two sets of error variances: $\sigma_1 = \sigma_2 = \sigma_3 = 0.1$ and $\sigma_1 = \sigma_2 = \sigma_3 = 0.3$. We have $p_{1,j} = p_{2,j}$ for all segments:

Segment 1: $p_{1,1} = p_{2,1} = 1$, $\boldsymbol{\theta}_1 = (\beta_{0,1}, \alpha_{1,1}, \beta_{1,1}, \boldsymbol{\gamma}_1) = (0, -0.1, 0.2, 0.1, 0.4, 0.1, 0.2) \in \mathbb{R}^7$.

Segment 2: $p_{1,2} = p_{2,2} = 2$, $\boldsymbol{\theta}_2 = (\beta_{0,2}, \alpha_{1,2}, \alpha_{2,2}, \beta_{1,2}, \beta_{2,2}, \boldsymbol{\gamma}_2) = (0, 0.2, -0.22, -0.12, 0.4, -0.1, 0.1, 0.2, -0.1) \in \mathbb{R}^9$.

Segment 3: $p_{1,3} = p_{2,3} = 2$, $\boldsymbol{\theta}_3 = (\beta_{0,3}, \alpha_{1,3}, \alpha_{2,3}, \beta_{1,3}, \beta_{2,3}, \boldsymbol{\gamma}_3) = (0, -0.12, 0.1, 0.25, -0.4, 0.2, -0.5, 0.1, 0.1) \in \mathbb{R}^9$.

The above regression coefficients guarantee that the NAR model in each segment is stationary [50, 53].

For each of the three network structures, 100 data sets were generated, and the proposed method was applied to estimate the breakpoints and other model parameters. One typical data set (with $\sigma_1 = \sigma_2 = \sigma_3 = 0.1$) from each of the network structures are shown in Figure 3.2, while the breakpoint estimation results for the 100 runs are summarized in Tables 3.2 and 3.3 respectively

for smaller and larger error variances. One can observe that the proposed method successfully detected the correct number of breakpoints in all cases when error variances were smaller. The method also produced excellent estimates for the breakpoint locations, as shown by their mean values and standard deviations.

We compare the estimation results of the lag orders without and with the refining step described in Section 3.3.2.7: the results for the smaller error variance cases are summarized, respectively, in Table 3.4 and 3.5. The results for the larger error variance cases are similar and hence omitted for brevity. One can observe that the refining step did indeed improve the estimation results of the lag orders. So if not specified, the refining step was applied in all the numerical work presented below.

The estimation results of the regression parameters $(\theta_1, \theta_2, \theta_3)$ are delayed to section C in the supplementary material.

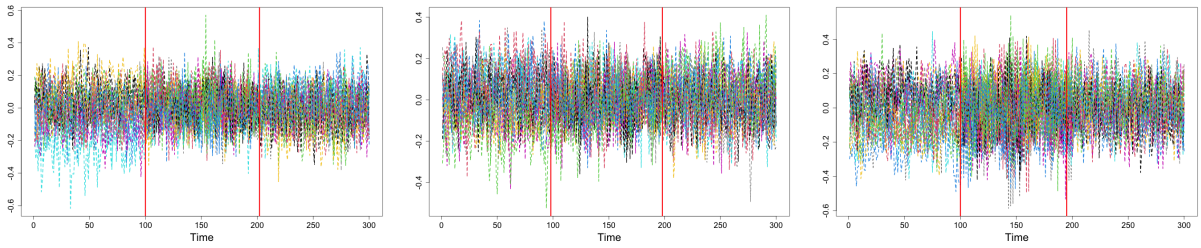


FIGURE 3.2. Typical simulated data sets with estimated breakpoints (vertical red lines). Left: power-law network structure; middle: dyad independence network structure; right: stochastic block network structure.

3.4.3. Scenario 2: Different segment lengths. In the second scenario, two sets of breakpoints were used: $\mathcal{T} = (\frac{T}{3}, \frac{2T}{3}) = (100, 200)$ and $\mathcal{T} = (\frac{T}{6}, \frac{5T}{6}) = (50, 250)$. The error variances are $\sigma_1 = \sigma_2 = \sigma_3 = 0.1$, and other model parameters are:

Segment 1: $p_{1,1} = p_{2,1} = 1$, $\boldsymbol{\theta}_1 = (\beta_{0,1}, \alpha_{1,1}, \beta_{1,1}, \boldsymbol{\gamma}_1) = (0, -0.1, 0.2, 0.1, 0.4, 0.1, 0.2) \in \mathbb{R}^7$.

Segment 2: $p_{1,2} = 2, p_{2,2} = 1$, $\boldsymbol{\theta}_2 = (\beta_{0,2}, \alpha_{1,2}, \alpha_{2,2}, \beta_{1,2}, \boldsymbol{\gamma}_2) = (0, 0.2, -0.22, -0.12, -0.1, 0.1, 0.2, -0.1)$ $\in \mathbb{R}^8$.

TABLE 3.2. Results of selected breakpoints of Scenario 1 when $\sigma_1 = \sigma_2 = \sigma_3 = 0.1$. Truth: locations of the true breakpoints; Mean/SD: means/standard deviations of the estimated breakpoint locations over the 100 simulation runs; Selection Rate: percentages of times that the correct number of breakpoints were selected.

	Breakpoint	Truth	Mean	SD	Selection Rate
Power-Law Structure	1	0.333	0.323	0.016	100%
	2	0.667	0.666	0.008	100%
Dyad Independence Structure	1	0.333	0.325	0.016	100%
	2	0.667	0.665	0.009	100%
Stochastic Block Structure	1	0.333	0.328	0.010	100%
	2	0.667	0.665	0.011	100%

TABLE 3.3. Similar to Table 3.2 but for $\sigma_1 = \sigma_2 = \sigma_3 = 0.3$.

	Breakpoint	Truth	Mean	SD	Selection Rate
Power-Law Structure	1	0.333	0.320	0.020	93%
	2	0.667	0.669	0.015	91%
Dyad Independence Structure	1	0.333	0.321	0.015	90%
	2	0.667	0.663	0.009	93%
Stochastic Block Structure	1	0.333	0.330	0.011	94%
	2	0.667	0.664	0.012	93%

Segment 3: $p_{1,3} = 1, p_{2,3} = 2, \boldsymbol{\theta}_3 = (\beta_{0,3}, \alpha_{1,3}, \beta_{1,3}, \beta_{2,3}, \boldsymbol{\gamma}_3) = (0, -0.12, 0.25, -0.4, 0.2, -0.5, 0.1, 0.1) \in \mathbb{R}^8$.

The estimated breakpoints from 100 simulation runs are summarized in Tables 3.6 and 3.7, in a similar fashion as in Scenario 1. The proposed method selected the correct number of breakpoints for all cases. It also gave excellent location estimates, as reflected by the mean and standard deviations.

TABLE 3.4. Estimated Lag orders in each segment of three network structures of Scenario 1 without the refining step of Section 3.3.2.7, where $\sigma_1 = \sigma_2 = \sigma_3 = 0.1$. The numbers are the proportions that a particular order was estimated. Bolded numbers correspond to the true orders.

	Orders	0	1	2	3	4	5
Power-Law Structure	$p_{1,1}$	0	**1**	0	0	0	0
	$p_{2,1}$	0	**1**	0	0	0	0
	$p_{1,2}$	0	0.05	**0.88**	0.06	0.01	0
	$p_{2,2}$	0	0.05	**0.85**	0.09	0.01	0
	$p_{1,3}$	0	0.12	**0.78**	0.08	0.02	0
	$p_{2,3}$	0	0.10	**0.78**	0.10	0.02	0
Dyad Independence Structure	$p_{1,1}$	0	**1**	0	0	0	0
	$p_{2,1}$	0	**1**	0	0	0	0
	$p_{1,2}$	0	0.11	**0.83**	0.05	0.01	0
	$p_{2,2}$	0	0.10	**0.80**	0.06	0.04	0
	$p_{1,3}$	0	0.23	**0.73**	0.04	0	0
	$p_{2,3}$	0	0.20	**0.75**	0.01	0.03	0.01
Stochastic Block Structure	$p_{1,1}$	0	**1**	0	0	0	0
	$p_{2,1}$	0	**1**	0	0	0	0
	$p_{1,2}$	0	0	**0.80**	0.20	0	0
	$p_{2,2}$	0	0	**0.83**	0.17	0	0
	$p_{1,3}$	0	0.12	**0.65**	0.19	0.04	0
	$p_{2,3}$	0	0.10	**0.70**	0.20	0	0

Estimation results of the lag orders are summarized in Tables 3.8 and 3.9. The results seem to be slightly worse for breakpoints $\mathcal{T} = (50, 250)$ when compared to $\mathcal{T} = (100, 200)$. One possible explanation is that the first and third segments are shorter, and hence there were less number of data points available for the estimation of the lag orders.

TABLE 3.5. Similar to Table 3.4 but with the refining step of Section 3.3.2.7

	Orders	0	1	2	3	4	5
Power-Law Structure	$p_{1,1}$	0	**1**	0	0	0	0
	$p_{2,1}$	0	**1**	0	0	0	0
	$p_{1,2}$	0	0	**1**	0	0	0
	$p_{2,2}$	0	0.01	**0.99**	0	0	0
	$p_{1,3}$	0	0.02	**0.98**	0	0	0
	$p_{2,3}$	0	0.02	**0.98**	0	0	0
Dyad Independence Structure	$p_{1,1}$	0	**1**	0	0	0	0
	$p_{2,1}$	0	**1**	0	0	0	0
	$p_{1,2}$	0	0	**1**	0	0	0
	$p_{2,2}$	0	0	**1**	0	0	0
	$p_{1,3}$	0	0	**0.97**	0.03	0	0
	$p_{2,3}$	0	0.02	**0.96**	0.02	0	0
Stochastic Block Structure	$p_{1,1}$	0	**1**	0	0	0	0
	$p_{2,1}$	0	**1**	0	0	0	0
	$p_{1,2}$	0	0	**1**	0	0	0
	$p_{2,2}$	0	0.01	**0.99**	0	0	0
	$p_{1,3}$	0	0	**1**	0	0	0
	$p_{2,3}$	0	0.01	**0.98**	0.01	0	0

3.4.4. Scenario 3: Correlated variance matrices. Here we set the breakpoints as $\mathcal{T} = (\frac{T}{3}, \frac{2T}{3}) = (100, 200)$ and adopt a more complicated error structure: the errors are correlated and the covariance matrix of the error terms is dense. To be more specific, we assume the covariance

TABLE 3.6. Similar to Table 3.2 but for Scenario 2 with true breakpoints $\mathcal{T} = (100, 200)$.

	Breakpoint	Truth	Mean	SD	Selection Rate
Power-Law	1	0.333	0.330	0.004	100%
Structure	2	0.667	0.662	0.010	100%
Dyad Independence	1	0.333	0.330	0.009	100%
Structure	2	0.667	0.660	0.011	100%
Stochastic Block	1	0.333	0.332	0.007	100%
Structure	2	0.667	0.662	0.010	100%

TABLE 3.7. Similar to Table 3.2 but for Scenario 2 with true breakpoints $\mathcal{T} = (50, 250)$.

	Breakpoint	Truth	Mean	SD	Selection Rate
Power-Law	1	0.167	0.167	0.012	100%
Structure	2	0.833	0.830	0.007	100%
Dyad Independence	1	0.167	0.166	0.011	100%
Structure	2	0.833	0.829	0.010	100%
Stochastic Block	1	0.167	0.165	0.007	100%
Structure	2	0.833	0.831	0.006	100%

matrix of error $\boldsymbol{\varepsilon}_j$ is $\boldsymbol{\Sigma}_{\boldsymbol{\varepsilon}_j} = 0.01((\sigma_{ij}))_{T\times T}$ with $\sigma_{ij} = 0.5^{|i-j|}$. All the remaining model parameters are the same as those in Scenario 2.

The estimation results for breakpoints and lag orders are summarized, respectively, in Tables 3.10 and 3.11. One can observe that the proposed method performed very well in this scenario, which confirms the applicability of the method in the case of correlated error terms. These results verify our previous claim that the variances of the error terms do not have to be the same for the proposed method to work.

TABLE 3.8. Similar to Table 3.5 but for Scenario 2 with true breakpoints $\mathcal{T} = (100, 200)$.

	Orders	0	1	2	3	4	5
Power-Law Structure	$p_{1,1}$	0	**1**	0	0	0	0
	$p_{2,1}$	0	**1**	0	0	0	0
	$p_{1,2}$	0	0.01	**0.99**	0	0	0
	$p_{2,2}$	0	**0.99**	0.01	0	0	0
	$p_{1,3}$	0	**0.96**	0.04	0	0	0
	$p_{2,3}$	0	0.01	**0.99**	0	0	0
Dyad Independence Structure	$p_{1,1}$	0	**1**	0	0	0	0
	$p_{2,1}$	0	**1**	0	0	0	0
	$p_{1,2}$	0	0.05	**0.95**	0	0	0
	$p_{2,2}$	0	**1**	0	0	0	0
	$p_{1,3}$	0	**0.98**	0.02	0	0	0
	$p_{2,3}$	0	0.02	**0.98**	0	0	0
Stochastic Block Structure	$p_{1,1}$	0	**1**	0	0	0	0
	$p_{2,1}$	0	**1**	0	0	0	0
	$p_{1,2}$	0	0.02	**0.98**	0	0	0
	$p_{2,2}$	0	**0.96**	0.04	0	0	0
	$p_{1,3}$	0	**0.98**	0.02	0	0	0
	$p_{2,3}$	0	0	**1**	0	0	0

3.4.5. Scenario 4: Slowly varying coefficient. In this scenario we consider the case that there is no breakpoint and one of the coefficients is slowly varying. The exact specification is: $p_1 = p_2 = 1$ and $\boldsymbol{\theta} = (\beta_0, \alpha_1, \beta_1, \boldsymbol{\gamma}_1) = (0, a_t, -0.1, 0.1, 0.4, 0.1, 0.2)$, where $a_t = 0.55 - 0.25\cos(\pi t/T)$

TABLE 3.9. Similar to Table 3.5 but for Scenario 2 with true breakpoints $\mathcal{T} = (50, 250)$.

	Orders	0	1	2	3	4	5
Power-Law Structure	$p_{1,1}$	0	**0.98**	0.02	0	0	0
	$p_{2,1}$	0	**1**	0	0	0	0
	$p_{1,2}$	0	0.04	**0.96**	0	0	0
	$p_{2,2}$	0	**0.99**	0.01	0	0	0
	$p_{1,3}$	0	**0.89**	0.11	0	0	0
	$p_{2,3}$	0	0.05	**0.95**	0	0	0
Dyad Independence Structure	$p_{1,1}$	0	**0.99**	0.01	0	0	0
	$p_{2,1}$	0	**0.99**	0.01	0	0	0
	$p_{1,2}$	0	0.11	**0.89**	0	0	0
	$p_{2,2}$	0	**0.96**	0.04	0	0	0
	$p_{1,3}$	0	**0.92**	0.08	0	0	0
	$p_{2,3}$	0	0.01	**0.97**	0.02	0	0
Stochastic Block Structure	$p_{1,1}$	0	**1**	0	0	0	0
	$p_{2,1}$	0	**1**	0	0	0	0
	$p_{1,2}$	0	0.04	**0.96**	0	0	0
	$p_{2,2}$	0	**0.94**	0.06	0	0	0
	$p_{1,3}$	0	**0.90**	0.10	0	0	0
	$p_{2,3}$	0	0.05	**0.95**	0	0	0

changes over time. Typical realizations of this process are shown in Figure 3.3 with breakpoints estimated by the proposed method.

For both the Power-Law and Dyad Independence Structures, no breakpoint was detected in any of the simulation runs, while for the Stochastic Block Structure, one breakpoint was always

TABLE 3.10. Similar to Table 3.2 but for Scenario 3.

	Breakpoint	Truth	Mean	SD	Selection Rate
Power-Law	1	0.333	0.324	0.011	100%
Structure	2	0.667	0.663	0.007	100%
Dyad Independence	1	0.333	0.325	0.010	100%
Structure	2	0.667	0.664	0.01	100%
Stochastic Block	1	0.333	0.323	0.012	100%
Structure	2	0.667	0.664	0.007	100%

detected near $0.6T$. The reason may be that the Stochastic Block structure has the lowest structure sparsity, so it contains less information in this difficult scenario.

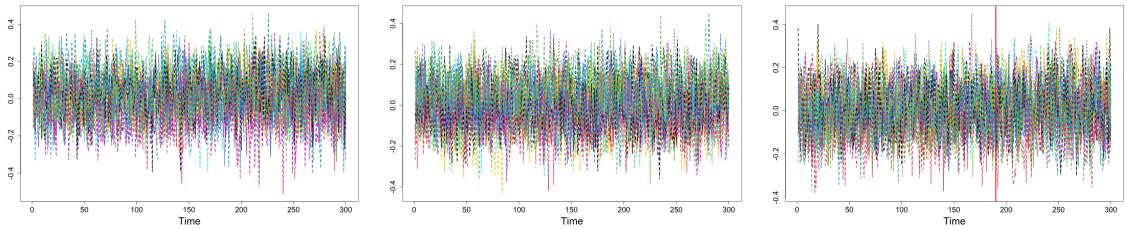


FIGURE 3.3. Typical simulated data sets with a slowly varying coefficient. Left: power-law network structure with no detected breakpoint; middle: dyad independence network structure with no detected breakpoint; right: stochastic block network structure with one detected breakpoint (red vertical line).

3.4.6. Scenario 5: Mis-specified $\boldsymbol{W}$. In this last scenario, the row normalized network matrix $\boldsymbol{W}$ is mis-specified. Although $\boldsymbol{W}$ is always assumed known and can be derived directly from the network structure matrix $\boldsymbol{A}$ in NAR models, in many applications, it is reasonable to assume that it is empirically defined and hence it may not be exactly accurate [53]. Here we use $\boldsymbol{W}_{obs} = \boldsymbol{W} + \pi_T$ with $\pi_T \sim N(0, 0.1^2)$ and $\pi_T \sim N(0, 0.5^2)$. Other model parameters are the same as those in Scenario 2 with true breakpoints $\mathcal{T} = (100, 200)$.

TABLE 3.11. Similar to Table 3.5 but for Scenario 3.

	Orders	0	1	2	3	4	5
Power-Law Structure	$p_{1,1}$	0	**1**	0	0	0	0
	$p_{2,1}$	0	**1**	0	0	0	0
	$p_{1,2}$	0	0	**0.99**	0.01	0	0
	$p_{2,2}$	0	**1**	0	0	0	0
	$p_{1,3}$	0	**0.97**	0.03	0	0	0
	$p_{2,3}$	0	0.02	**0.98**	0	0	0
Dyad Independence Structure	$p_{1,1}$	0	**1**	0	0	0	0
	$p_{2,1}$	0	**1**	0	0	0	0
	$p_{1,2}$	0	0.03	**0.97**	0	0	0
	$p_{2,2}$	0	**0.99**	0.01	0	0	0
	$p_{1,3}$	0	**0.98**	0.02	0	0	0
	$p_{2,3}$	0	0.02	**0.98**	0	0	0
Stochastic Block Structure	$p_{1,1}$	0	**1**	0	0	0	0
	$p_{2,1}$	0	**1**	0	0	0	0
	$p_{1,2}$	0	0.02	**0.98**	0	0	0
	$p_{2,2}$	0	**0.95**	0.05	0	0	0
	$p_{1,3}$	0	**0.99**	0.01	0	0	0
	$p_{2,3}$	0	0	**1**	0	0	0

The estimation results for the breakpoints and lag orders are summarized in Tables 3.12 to 3.15. These results are comparable to those from Scenario 2, which suggests that the introduction of the error term π_T did not affect the estimation results too much. In other words, the

results suggest that the proposed method is, to a certain extent, robust against changes in the network structure matrix.

TABLE 3.12. Similar to Table 3.2 but for Scenario 5 with $\pi_T \sim N(0, 0.1^2)$.

	Breakpoint	Truth	Mean	SD	Selection Rate
Power-Law	1	0.333	0.330	0.009	100%
Structure	2	0.667	0.660	0.008	100%
Dyad Independence	1	0.333	0.320	0.007	100%
Structure	2	0.667	0.664	0.009	100%
Stochastic Block	1	0.333	0.330	0.013	100%
Structure	2	0.667	0.663	0.010	100%

TABLE 3.13. Similar to Table 3.2 but for Scenario 5 with $\pi_T \sim N(0, 0.5^2)$.

	Breakpoint	Truth	Mean	SD	Selection Rate
Power-Law	1	0.333	0.331	0.010	94%
Structure	2	0.667	0.662	0.012	92%
Dyad Independence	1	0.333	0.323	0.010	95%
Structure	2	0.667	0.662	0.008	92%
Stochastic Block	1	0.333	0.331	0.015	91%
Structure	2	0.667	0.661	0.012	93%

3.5. Real Data Analysis

This section applies the proposed method to a Manhattan yellow cab demand data set, which was obtained from the NYC Taxi and Limousine Commission's website[1]. This dataset depicts the number of yellow cab pick-ups in different taxi zones and is aggregated spatially over the zipcodes. Here, Manhattan was divided into 64 taxi zones, as shown in Figure 3.4. Note that zones 103, 104,

[1]https://www1.nyc.gov/site/tlc/about/tlc-trip-record-data.page

TABLE 3.14. Similar to Table 3.5 but for Scenario 5 with $\pi_T \sim N(0, 0.1^2)$.

	Orders	0	1	2	3	4	5
Power-Law Structure	$p_{1,1}$	0	**1**	0	0	0	0
	$p_{2,1}$	0	**1**	0	0	0	0
	$p_{1,2}$	0	0.02	**0.98**	0	0	0
	$p_{2,2}$	0	**0.97**	0.03	0	0	0
	$p_{1,3}$	0	**0.95**	0.05	0	0	0
	$p_{2,3}$	0	0.02	**0.98**	0	0	0
Dyad Independence Structure	$p_{1,1}$	0	**1**	0	0	0	0
	$p_{2,1}$	0	**1**	0	0	0	0
	$p_{1,2}$	0	0.04	**0.96**	0	0	0
	$p_{2,2}$	0	**0.99**	0.01	0	0	0
	$p_{1,3}$	0	**0.98**	0.02	0	0	0
	$p_{2,3}$	0	0.02	**0.98**	0	0	0
Stochastic Block Structure	$p_{1,1}$	0	**1**	0	0	0	0
	$p_{2,1}$	0	**1**	0	0	0	0
	$p_{1,2}$	0	0.02	**0.98**	0	0	0
	$p_{2,2}$	0	**0.97**	0.03	0	0	0
	$p_{1,3}$	0	**1**	0	0	0	0
	$p_{2,3}$	0	0.01	**0.99**	0	0	0

105, 202, and 194 are isolated with no common boundaries with any other zones, thus these five zones were not included in the analysis.

TABLE 3.15. Similar to Table 3.5 but for Scenario 5 with $\pi_T \sim N(0, 0.5^2)$.

	Orders	0	1	2	3	4	5
Power-Law Structure	$p_{1,1}$	0	**0.98**	0	0	0	0.02
	$p_{2,1}$	0.01	**0.89**	0	0	0	0.1
	$p_{1,2}$	0	0.05	**0.90**	0	0	0.05
	$p_{2,2}$	0	**0.91**	0.06	0	0	0.03
	$p_{1,3}$	0	**0.90**	0.06	0	0	0.04
	$p_{2,3}$	0	0.04	**0.92**	0	0	0.04
Dyad Independence Structure	$p_{1,1}$	0	**0.95**	0.02	0.03	0	0
	$p_{2,1}$	0	**0.96**	0	0	0	0.04
	$p_{1,2}$	0	0.06	**0.89**	0	0	0.05
	$p_{2,2}$	0.08	**0.87**	0.01	0	0	0.05
	$p_{1,3}$	0	**0.88**	0.07	0	0	0.05
	$p_{2,3}$	0	0.02	**0.89**	0	0	0.09
Stochastic Block Structure	$p_{1,1}$	0	**0.97**	0.03	0	0	0
	$p_{2,1}$	0	**0.99**	0.01	0	0	0
	$p_{1,2}$	0	0.07	**0.90**	0	0	0.03
	$p_{2,2}$	0	**0.88**	0.05	0	0	0.07
	$p_{1,3}$	0	**0.92**	0	0	0	0.08
	$p_{2,3}$	0	0.02	**0.89**	0	0	0.09

For the remaining 59 zones, we aggregated the numbers of their yellow cab pick-ups temporally over 15-minutes intervals for the date April 16th, 2014. Thus, there are $T = 96$ time points with $K = 59$ nodes at each time point. The network structure $\boldsymbol{A} = (a_{i_1,i_2}) \in \mathbb{R}^{59\times 59}$ was constructed by using the physical relationships of these taxi zones: $a_{i_1,i_2} = 1$ if zones i_1 and i_2 share a common boundary, otherwise $a_{i_1,i_2} = 0$. We considered the average tip amount of the trips in different zones in April 2014 as the node-specified exogenous covariates.

We performed a first-order difference to the data to remove the first-order non-stationarities. Altogether, three breakpoints were detected by the proposed method; they are shown in Figure 3.5. These three breakpoints correspond to 6:15 AM, 11:30 AM, and 5:45 PM, which seem to coincide with the daily major changes in traffic patterns in Manhattan: people commute to work, go out for lunch, and return home after work. These results broadly agree with those in [76]. Lastly, the fitting time for this data set is about 46.7 seconds per generation on a 2020 Macbook Pro with an M1 chip.

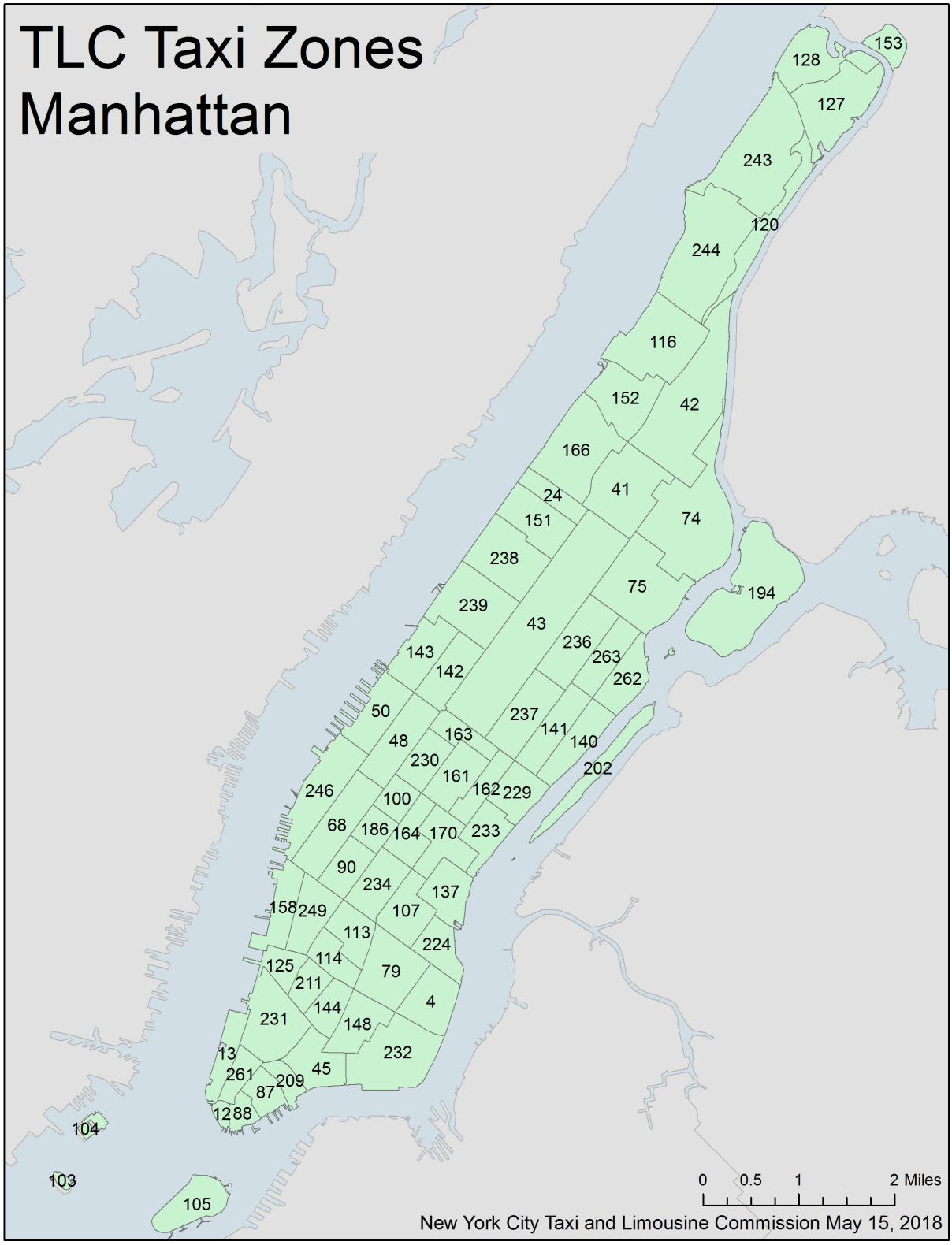


FIGURE 3.4. Taxi Zones of Manhattan.[2]

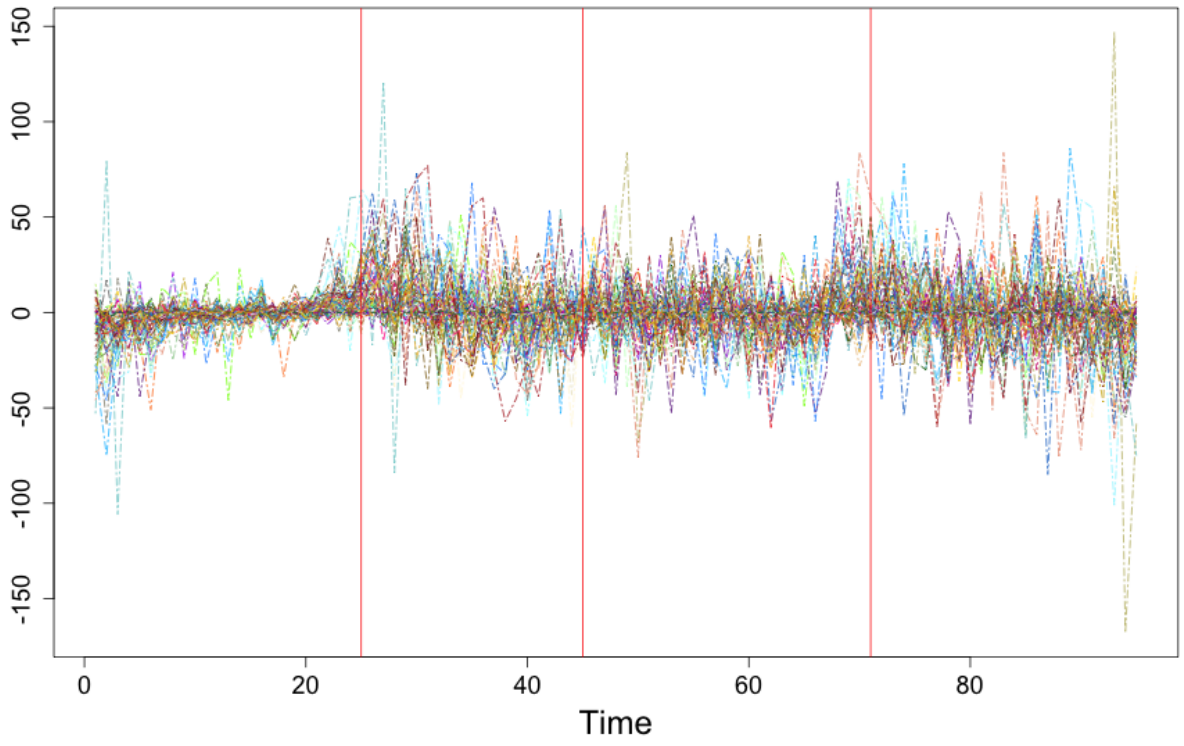


FIGURE 3.5. The one-lag difference time series of the Manhattan yellow cab data set in 59 taxi zones over 96 time points taken on April 16th, 2014. Altogether 3 breakpoints were detected (red vertical lines).

3.6. Concluding Remarks

This chapter developed a method for simultaneous multiple breakpoint detection and parameter estimation for piecewise stationary NAR models. The proposed method utilizes the MDL principle to derive an objective criterion for estimating the breakpoints as well as other model parameters. It has been shown that the MDL estimates enjoy desirable asymptotic properties. To optimize the MDL objective criterion, the proposed method uses a tailor-made GA. Through a sequence of simulation experiments, the proposed method is shown to possess excellent empirical properties. Lastly, the proposed method was applied to analyze a Manhattan yellow cab data set and yielded similar results as those reported in the literature.

[2]Source: https://www1.nyc.gov/site/tlc/about/tlc-trip-record-data.page

The proposed methodology offers several notable advantages, such as its statistical consistency and the straightforward interpretation offered by piecewise stationary NAR models. However, it is not without its limitations. One key constraint is its reliance on the piecewise stationary assumption; deviation from this assumption could lead to the identification of breakpoints that do not truly exist, such as in cases where the data undergo gradual changes without clear breakpoints. Additionally, the method presupposes an observed network structure that stays constant throughout the observation period, a condition that may not always hold. While network imputation techniques [77, 78, 79] offer a remedy by enabling the reconstruction of the missing network structure, they introduce a layer of uncertainty to the analysis.

Future work includes two ambitious goals. First, we aim to enhance the method by incorporating uncertainty quantification for the fitted piecewise stationary NAR model. Uncertainty quantification is of great important in a lot of area [80]. This development would provide a deeper understanding of the model's predictive confidence across different segments. Second, we plan to explore strategies for condensing the network without losing critical information. This endeavor will investigate approaches similar to factor modeling in high-dimensional time series analysis, aiming to maintain the important information hidden in the data while simplifying the model's complexity.

3.7. Supplementary materials

3.7.1. Proof and Technical Details.

3.7.1.1. *General Notations.* We follow the ideas of [81] and establish our theoretical results using proof by contradiction. We observe that the MDL criterion (3.12) can be decomposed into two parts: the minus log-likelihood part and the penalty part, where the latter can be controlled asymptotically.

Denote $\boldsymbol{p}_j = (p_{1,j}, p_{2,j})$, and w.l.o.g., assume $p_{1,j} > p_{2,j}$. Let $f_{\boldsymbol{p_j}}(\boldsymbol{X}_{i,j}|\boldsymbol{X}_{s,j}, s < i; \boldsymbol{\theta}_j)$ be the conditional density function of the i-th observation in the j-th segment. Also let $l_j(\boldsymbol{p}_j, \boldsymbol{\theta}_j, \boldsymbol{X}_{i,j}|\boldsymbol{X}_{s,j}, s < i) = \log f_{\boldsymbol{p_j}}(\boldsymbol{X}_{i,j}|\boldsymbol{X}_{s,j}, s < i; \boldsymbol{\theta_j})$ be the conditional log-likelihood function for $\boldsymbol{X}_{i,j}$.

From model (3.7), it can be seen that the distribution of $\boldsymbol{X}_{i,j}$ depends on $\boldsymbol{X}_{i-1,j}, \cdots, \boldsymbol{X}_{i-p_{1,j},j}$. However, for the first $p_{1,j}$ data, it is not possible to observe all the required preceding data. So, we need to handle the gap between them. To proceed, we define the real observed past data to be $\boldsymbol{y}_{i,j} = (\boldsymbol{X}_{1,1}, \cdots, \boldsymbol{X}_{n_1,1}, \cdots, \boldsymbol{X}_{1,j}, \cdots, \boldsymbol{X}_{i-1,j})$.

The theoretical conditional log-likelihood of the j-th segment given all the **unobserved** past is

$$L_j(\boldsymbol{p}_j, \boldsymbol{\theta}_j; \boldsymbol{X}_j) = \sum_{i=1}^{n_j} l_j(\boldsymbol{p}_j, \boldsymbol{\theta}_j, \boldsymbol{X}_{i,j} | \boldsymbol{X}_{s,j}, s < i) = \sum_{i=1}^{n_j} l_j(\boldsymbol{p}_j, \boldsymbol{\theta}_j, \boldsymbol{X}_{i,j} | \boldsymbol{X}_{s,j}, i - p_{1,j} \leq s < i).$$

The real observed conditional log-likelihood of the j-th segment given all the **real observed** past is

$$\begin{aligned}\tilde{L}_j(\boldsymbol{p}_j \boldsymbol{\theta}_j; \boldsymbol{X}_j) &= \sum_{i=1}^{n_j} l_j(\boldsymbol{p}_j, \boldsymbol{\theta}_j, \boldsymbol{X}_{i,j} | \boldsymbol{X}_{s,j}, s < i) \\ &= \sum_{i=1}^{p_{1,j}} l_j(\boldsymbol{p}_j, \boldsymbol{\theta}_j, \boldsymbol{X}_{i,j} | \boldsymbol{y}_{i,j}) + \sum_{i=p_{1,j}+1}^{n_j} l_j(\boldsymbol{p}_j, \boldsymbol{\theta}_j, \boldsymbol{X}_{i,j} | \boldsymbol{X}_{s,j}, i - p_{1,j} \leq s < i).\end{aligned}$$

Due to imperfect estimation and the use of the refining step of Section 3.3.2.7, for most practical situations we can only estimate parameters with a portion of the data in any one specific segmentation. To handle this issue, let λ_u, $\lambda_d \in [0, 1]$ and $\lambda_u - \lambda_d > \epsilon_\lambda > 0$, and define

$$\begin{aligned}L_j(\boldsymbol{p}_j, \boldsymbol{\theta}_j, \lambda_d, \lambda_u; \boldsymbol{X}_j) &= \sum_{i=[n_j\lambda_d]+1}^{[n_j\lambda_u]} l_j(\boldsymbol{p}_j, \boldsymbol{\theta}_j, \boldsymbol{X}_{i,j} | \boldsymbol{X}_{s,j}, s < i) \\ &= \sum_{i=[n_j\lambda_d]+1}^{[n_j\lambda_u]} l_j(\boldsymbol{p}_j, \boldsymbol{\theta}_j, \boldsymbol{X}_{i,j} | \boldsymbol{X}_{s,j}, i - p_{1,j} \leq s < i)\end{aligned}$$

and

$$\begin{aligned}\tilde{L}_j(\boldsymbol{p}_j, \boldsymbol{\theta}_j, \lambda_d, \lambda_u; \boldsymbol{X}_j) &= \sum_{i=[n_j\lambda_d]+1}^{[n_j\lambda_u]} l_j(\boldsymbol{p}_j, \boldsymbol{\theta}_j, \boldsymbol{X}_{i,j} | \boldsymbol{X}_{s,j}, s < i) \\ &= \sum_{i=[n_j\lambda_d]+1}^{[n_j\lambda_d]+p_{1,j}} l_j(\boldsymbol{p}_j, \boldsymbol{\theta}_j, \boldsymbol{X}_{i,j} | \boldsymbol{y}_{i,j}) + \sum_{i=[n_j\lambda_d]+p_{1,j}+1}^{[n_j\lambda_u]} l_j(\boldsymbol{p}_j, \boldsymbol{\theta}_j, \boldsymbol{X}_{i,j} | \boldsymbol{X}_{s,j}, i - p_{1,j} \leq s < i),\end{aligned}$$

which represent, respectively, the theoretical conditional log-likelihood and real observed log-likelihood functions based on partial segmentation. We begin by presenting two lemmas.

3.7.1.2. *Lemmas.*

LEMMA 3.7.1. *For the piecewise stationary NAR(p_1, p_2) model (3.2), for any fixed $\boldsymbol{p}_j$, we have*

$$\sup_{\lambda_d,\lambda_u}\sup_{\boldsymbol{\theta}_j\in\Theta_j(\boldsymbol{p}_j)}|\frac{1}{T}\tilde{L}_j(\boldsymbol{p}_j,\boldsymbol{\theta}_j,\lambda_d,\lambda_u;\boldsymbol{X}_j)-(\lambda_u-\lambda_d)L^{(j)}(\boldsymbol{p}_j,\boldsymbol{\theta}_j)|\xrightarrow{a.s}0, \tag{3.15}$$

where λ_d, λ_u are defined as before, $\Theta(\boldsymbol{p}_j)$ is a compact parameter space of $\boldsymbol{\theta}_j$, and $L^{(j)}(\boldsymbol{p}_j,\boldsymbol{\theta}_j) := E(l_j(\boldsymbol{p}_j,\boldsymbol{\theta}_j,\boldsymbol{X}_{1,j}|\boldsymbol{X}_{l,j}, l<1))$.

PROOF. First, we prove that for any segment j, $j=1,\ldots,m+1$ and fixed $\boldsymbol{p}_j$, we have

$$\sup_{\lambda_d,\lambda_u}\sup_{\boldsymbol{\theta}_j\in\Theta_j(p_j)}|\frac{1}{T}L_j(\boldsymbol{p}_j,\boldsymbol{\theta}_j,\lambda_d,\lambda_u;\boldsymbol{X}_j)-\frac{1}{T}\tilde{L}_j(\boldsymbol{p}_j,\boldsymbol{\theta}_j,\lambda_d,\lambda_u;\boldsymbol{X}_j)|=o(T^{-\frac{1}{2}}), \tag{3.16}$$

which means that we only need to consider the $L_j(\boldsymbol{p}_j,\boldsymbol{\theta}_j,\lambda_d,\lambda_u;\boldsymbol{X}_j)$ instead of $\tilde{L}_j(\boldsymbol{p}_j,\boldsymbol{\theta}_j,\lambda_d,\lambda_u;\boldsymbol{X}_j)$.

Now calculate

$$\begin{aligned}
&|\frac{1}{T}L_j(\boldsymbol{p}_j,\boldsymbol{\theta}_j,\lambda_d,\lambda_u;\boldsymbol{X}_j)-\frac{1}{T}\tilde{L}_j(\boldsymbol{p}_j,\boldsymbol{\theta}_j,\lambda_d,\lambda_u;\boldsymbol{X}_j)|\\
&\quad=|\frac{1}{T}\sum_{i=[n_j\lambda_d]+1}^{[n_j\lambda_d]+p_{1,j}}l_j(\boldsymbol{p}_j,\boldsymbol{\theta}_j,\boldsymbol{X}_{i,j}|\boldsymbol{X}_{s,j},s<i)-\sum_{i=[n_j\lambda_d]+1}^{[n_j\lambda_d]+p_{1,j}}l_j(\boldsymbol{p}_j,\boldsymbol{\theta}_j,\boldsymbol{X}_{i,j}|\boldsymbol{y}_{i,j})|\\
&\quad=|\frac{1}{2T\hat{\sigma}_j^2}\sum_{i=[n_j\lambda_d]+1}^{[n_j\lambda_d]+p_{1,j}}(2[(\boldsymbol{X}_{i,j}-\hat{\mathcal{B}}_{0,j}-\sum_{l=1}^{i-1}\hat{G}_{l,j}\boldsymbol{X}_{i-l,j})])^T\times(\boldsymbol{X}_{i,j}-\hat{\mathcal{B}}_{0,j}-\sum_{l=1}^{i-1}\hat{G}_{l,j}\boldsymbol{X}_{i-l,j})])\\
&\quad+(\sum_{r=i}^{p_{1,j}}\hat{G}_{l,j}\boldsymbol{X}_{i-l,j})^T(\sum_{r=i}^{p_{1,j}}\hat{G}_{l,j}\boldsymbol{X}_{i-l,j})-(\sum_{r=i}^{p_{1,j}}\hat{G}_{l,j}\boldsymbol{y}_{i-l,j})^T(\sum_{r=i}^{p_{1,j}}\hat{G}_{l,j}\boldsymbol{y}_{i-l,j})|.
\end{aligned}$$

Since $\|\boldsymbol{W}\|_{\max}=1$, $\sum_{i=1}^{p_{1,j}}(|\alpha_i|+|\beta_i|)<=1$ as well as $p_{1,j}$ is bounded, so as $T\to+\infty$, (3.16) is satisfied.

Due to the compactness of the parameter space $\Theta_j(\boldsymbol{p}_j)$ and the ergodic theorem for NAR$(p_{1,j},p_{2,j})$ model, we have

$$\sup_{\theta_j\in\Theta_j(\boldsymbol{p}_j)}|\frac{1}{T}L_j(\boldsymbol{p}_j,\boldsymbol{\theta}_j;\boldsymbol{X}_j)-L^{(j)}(\boldsymbol{p}_j,\boldsymbol{\theta}_j)|\xrightarrow{a.s}0,$$

which can be viewed as a strong law of large numbers for the time series.

Let $Q_{[0,1]}$ be the set of rational numbers in $[0,1]$, for $r_1,r_2\in Q_{[0,1]}$ and $r_1<r_2$, then we further have

$$\begin{aligned}
&\sup_{\boldsymbol{\theta}_j\in\Theta_j(\boldsymbol{p}_j)}|\frac{1}{T}L_j(\boldsymbol{p}_j,\boldsymbol{\theta}_j,r_1,r_2;\boldsymbol{X}_j)-(r_2-r_1)L^{(j)}(\boldsymbol{p}_j,\boldsymbol{\theta}_j)|\\
&=\sup_{\boldsymbol{\theta}_j\in\Theta_j(p_j)}|r_2(\frac{1}{Tr_2}\sum_{i=1}^{[Tr_2]}l_j(\boldsymbol{p}_j,\boldsymbol{\theta}_j,\boldsymbol{X}_{i,j}|\boldsymbol{X}_{s,j},s<i)-L^{(j)}(\boldsymbol{p}_j,\boldsymbol{\theta}_j))\\
&-r_1(\frac{1}{Tr_1}\sum_{i=1}^{[Tr_1]}l_j(\boldsymbol{p}_j,\boldsymbol{\theta}_j,\boldsymbol{X}_{i,j}|\boldsymbol{X}_{s,j},s<i)-L^{(j)}(\boldsymbol{p}_j,\boldsymbol{\theta}_j))|\xrightarrow{a.s}0
\end{aligned} \tag{3.17}$$

Let B_{r_1,r_2} be a probability measure 1 set that (3.17) holds, and $\omega \in B_{r_1,r_2}$ be any events in it. Define

$$B = \bigcap_{r_1,r_2 \in Q_{[0,1]}} B_{r_1,r_2}. \tag{3.18}$$

Using countable sub-additivity, it can be shown that $P(B) = 1$. Then for any $\omega \in B$ and for any $\lambda_u \in [0,1]$, we can choose r_1, $r_2 \in Q_{[0,1]}$ in such a way that $r_1 \leq \lambda_u \leq r_2$, and hence we have

$$\begin{aligned}
&\sup_{\boldsymbol{\theta}_j \in \Theta_j(\boldsymbol{p}_j)} |\frac{1}{T}\sum_{i=1}^{[T\lambda_u]} l_j(\boldsymbol{p}_j, \boldsymbol{\theta}_j, \boldsymbol{X}_{i,j}|\boldsymbol{X}_{s,j}, s<i) - \frac{1}{T}\sum_{i=1}^{[Tr_1]} l_j(\boldsymbol{p}_j, \boldsymbol{\theta}_j, \boldsymbol{X}_{i,j}|\boldsymbol{X}_{s,j}, s<i)| \\
&= \sup_{\boldsymbol{\theta}_j \in \Theta_j(\boldsymbol{p}_j)} |\frac{1}{T}\sum_{i=[Tr_1]+1}^{[T\lambda_u]} l_j(\boldsymbol{p}_j, \boldsymbol{\theta}_j, \boldsymbol{X}_{i,j}|\boldsymbol{X}_{s,j}, s<i)| \\
&\leq \sup_{\boldsymbol{\theta}_j \in \Theta_j(\boldsymbol{p}_j)} \frac{1}{T}\sum_{i=[Tr_1]+1}^{[T\lambda_u]} |l_j(\boldsymbol{p}_j, \boldsymbol{\theta}_j, \boldsymbol{X}_{i,j}|\boldsymbol{X}_{s,j}, s<i)| \\
&\leq \sup_{\boldsymbol{\theta}_j \in \Theta_j(\boldsymbol{p}_j)} \frac{1}{T}\sum_{i=[Tr_1]+1}^{[Tr_2]} |l_j(\boldsymbol{p}_j, \boldsymbol{\theta}_j, \boldsymbol{X}_{i,j}|\boldsymbol{X}_{s,j}, s<i)| \\
&\longrightarrow (r_2 - r_1) \sup_{\boldsymbol{\theta}_j \in \Theta_j(\boldsymbol{p}_j)} E|L^{(j)}(\boldsymbol{p}_j, \boldsymbol{\theta}_j)|.
\end{aligned}$$

Next, it can be verified due to the compactness of the parameter space as well as the moment boundless of the normal distribution that

$$\sup_{\boldsymbol{\theta}_j \in \Theta_j(\boldsymbol{p}_j)} E|L^{(j)}(\boldsymbol{p}_j, \boldsymbol{\theta}_j)|^{1+\epsilon} < +\infty.$$

So if $|r_2 - r_1| < \psi$, and by making ψ arbitrarily small, we can derive

$$\frac{1}{T} L_j(\boldsymbol{p}_j, \boldsymbol{\theta}_j, 0, \lambda_u; \boldsymbol{X}_j) \xrightarrow{a.s} \lambda_u L^{(j)}(\boldsymbol{p}_j, \boldsymbol{\theta}_j)$$

uniformly in $\boldsymbol{\theta}_j \in \Theta_j(\boldsymbol{p}_j)$. Similarly, we have

$$\sup_{\boldsymbol{\theta}_j \in \Theta_j(\boldsymbol{p}_j)} |\frac{1}{n} L_j(\boldsymbol{p}_j, \boldsymbol{\theta}_j, \lambda_d, \lambda_u; \boldsymbol{X}_j) - (\lambda_u - \lambda_d) L^{(j)}(\boldsymbol{p}_j, \boldsymbol{\theta}_j)| \xrightarrow{a.s} 0. \tag{3.19}$$

Now we prove (3.19) holds uniformly on $\lambda_d, \lambda_u \in [0,1]$. Let $\lambda_u - \lambda_d > \epsilon_\lambda$. For any fixed $\epsilon < \epsilon_\lambda$, we can choose one specific N such that $= 0 = r_0 < r_1 < \cdots < r_{N-1} < r_N = 1$ and $\max_{i \in [0,N]}(r_{i+1} - r_i) \leq \epsilon$. Then for any $\lambda_d, \lambda_u \in [0,1]$, we have specific $g < h$ such that

$r_g < \lambda_d < r_{g+1}$ and $r_h < \lambda_u < r_{h+1}$. Then we have for T large enough

(3.20)

$$
\begin{aligned}
&|\frac{1}{T}L_j(\boldsymbol{p}_j, \boldsymbol{\theta}_j, \lambda_d, \lambda_u; \boldsymbol{X}_j) - (\lambda_u - \lambda_d)L^{(j)}(\boldsymbol{p}_j, \boldsymbol{\theta}_j)| \\
&\qquad \leq |\frac{1}{T}L_j(\boldsymbol{p}_j, \boldsymbol{\theta}_j, \lambda_d, \lambda_u; \boldsymbol{X}_j) - \frac{1}{T}L_j(\boldsymbol{p}_j, \boldsymbol{\theta}_j, r_g, r_{h+1}; \boldsymbol{X}_j)| \\
&\qquad + |\frac{1}{T}L_j(\boldsymbol{p}_j, \boldsymbol{\theta}_j, r_g, r_{h+1}; \boldsymbol{X}_j) - (r_g - r_{h+1})L^{(j)}(\boldsymbol{p}_j, \boldsymbol{\theta}_j)| \\
&\qquad + |(r_g - r_{h+1})L^{(j)}(\boldsymbol{p}_j, \boldsymbol{\theta}_j) - (\lambda_u - \lambda_d)L^{(j)}(\boldsymbol{p}_j, \boldsymbol{\theta}_j)| \\
&\qquad \leq 2\epsilon \sup_{\boldsymbol{\theta}_j \in \Theta_j(\boldsymbol{p}_j)} E|L^{(j)}(\boldsymbol{p}_j, \boldsymbol{\theta}_j)| + \epsilon + 2\epsilon \sup_{\boldsymbol{\theta}_j \in \Theta_j(\boldsymbol{p}_j)} E|L^{(j)}(\boldsymbol{p}_j, \boldsymbol{\theta}_j)| \xrightarrow{a.s} 0,
\end{aligned}
$$

which completes the proof. □

LEMMA 3.7.2. *This lemma consists of three results.*

(R1) For the j-th stationary piece of a NAR model, there exists a model $\boldsymbol{p}_j^0 \in \mathcal{M}$ with parameter $\boldsymbol{\theta}_j^0 \in \mathbb{R}^{d_j}$ that satisfies $(\boldsymbol{p}_j^0, \boldsymbol{\theta}_j^0) = \arg\max_{\boldsymbol{p},\boldsymbol{\theta}} E(l_j(\boldsymbol{p}, \boldsymbol{\theta}, \boldsymbol{X}_{i,j}|\boldsymbol{X}_{s,j}, s < j))$. Also, the model $\boldsymbol{p}_j^0$ is uniquely identifiable.

(R2) Suppose that $\boldsymbol{p}_b$ is a bigger model than $\boldsymbol{p}_s$ with $\boldsymbol{p}_b$, $\boldsymbol{p}_s$ associated with parameter vectors $\boldsymbol{\theta}_b \in \Theta(\boldsymbol{p}_b) \subset \mathbb{R}^{d_b}$ and $\boldsymbol{\theta}_s \in \Theta(\boldsymbol{p}_s) \subset \mathbb{R}^{d_s}$, respectively. Here a bigger model means for every $\boldsymbol{\theta}_s \in \Theta(\boldsymbol{p}_s)$, there exists a $\boldsymbol{\theta}_b^* \in \Theta(\boldsymbol{p}_b)$ such that for every $\boldsymbol{X}_i$, the conditional densities are equal almost everywhere:

(3.21)
$$f_{\boldsymbol{p}_b}(\boldsymbol{X}_{i,j}|\boldsymbol{X}_{s,j}, s < i; \boldsymbol{\theta}_b^*) = f_{\boldsymbol{p}_s}(\boldsymbol{X}_{i,j}|\boldsymbol{X}_{s,j}, s < i; \boldsymbol{\theta}_s).$$

Then $\boldsymbol{\theta}^b$ can be partitioned into three parts $\boldsymbol{\theta}_b = (\boldsymbol{\beta}, \boldsymbol{\xi}, \boldsymbol{\pi})$, where $\boldsymbol{\beta} \in \Theta_{\boldsymbol{\beta}} \subset \mathbb{R}^{d_\beta}$, $\boldsymbol{\xi} \in \Theta_{\boldsymbol{\xi}} \subset \mathbb{R}^{d_s}$ and $\boldsymbol{\pi} \in \Theta_{\boldsymbol{\pi}} \subset \mathbb{R}^{d_\pi}$. Also, $d_\beta + d_s + d_\pi = d_b$ and $\Theta_{\boldsymbol{\beta}}$, $\Theta_{\boldsymbol{\xi}}$ and $\Theta_{\boldsymbol{\pi}}$ are compact. Now, for any given $\boldsymbol{\pi} \in \Theta_{\boldsymbol{\pi}}$, the vector $\boldsymbol{\theta}_b^* = (\mathbf{0}, \boldsymbol{\theta}_s, \boldsymbol{\pi})$ is the unique vector in the neighbourhood

$\{\boldsymbol{\theta}_b = (\boldsymbol{\beta}, \boldsymbol{\xi}, \boldsymbol{\pi}) : |\boldsymbol{\beta}| < \delta, |\boldsymbol{\xi} - \boldsymbol{\theta}_s| < \delta\}$ satisfying (3.21) for some $\delta > 0$.

(R3) If the true model order for the j-th piece is $\boldsymbol{p}_j^0$, $j = 1, \ldots, m+1$, and is specified, then

(3.22)
$$\hat{\boldsymbol{\theta}}_n^j(\hat{\lambda}_{j-1}, \hat{\lambda}_j) - \boldsymbol{\theta}_j^0 = O(\sqrt{\frac{\log\log(T)}{T}}), \quad a.s.,$$

where $\boldsymbol{\theta}_j^0$ is the true parameter vector and $\hat{\boldsymbol{\theta}}_n^j(\hat{\lambda}_{j-1}, \hat{\lambda}_j) = \arg\max_{\boldsymbol{\theta}_j} \tilde{L}_j(\boldsymbol{p}_j^0, \boldsymbol{\theta}_j, \hat{\lambda}_{j-1}, \hat{\lambda}_j; \boldsymbol{X}_j)$.

Suppose the specified model $\boldsymbol{p}_j$ is larger than the true model $\boldsymbol{p}_j^0$, then we have the partition

$$\hat{\boldsymbol{\theta}}_n = (\hat{\boldsymbol{\beta}}_n^j(\hat{\lambda}_{j-1}, \hat{\lambda}_j), \hat{\boldsymbol{\xi}}_n^j(\hat{\lambda}_{j-1}, \hat{\lambda}_j), \hat{\boldsymbol{\pi}}_n^j(\hat{\lambda}_{j-1}, \hat{\lambda}_j))$$

with

(3.23)
$$
\begin{aligned}
\hat{\boldsymbol{\beta}}_n^j(\hat{\lambda}_{j-1}, \hat{\lambda}_j) &= O(\sqrt{\frac{\log\log(T)}{T}}), \quad a.s. \\
\hat{\boldsymbol{\xi}}_n^j(\hat{\lambda}_{j-1}, \hat{\lambda}_j) - \boldsymbol{\theta}_j^0 &= O(\sqrt{\frac{\log\log(T)}{T}}), \quad a.s.
\end{aligned}
$$

PROOF. Results (R1) and (R2) are satisfied by our piecewise stationary NAR model, as the errors are assumed to be normally distributed, and the normal distribution is determined uniquely

by its mean and variances. It remains to prove (R3), and we shall only prove (3.23) as (3.22) can be proved similarly.

Define $\boldsymbol{\theta}_n^0 = (\mathbf{0}, \boldsymbol{\theta}_j^0, \hat{\boldsymbol{\pi}}_n^j)$, $\boldsymbol{\gamma}_n^0 = (\mathbf{0}, \boldsymbol{\theta}_j^0)$ and let $\hat{\boldsymbol{\gamma}}_n^j = (\hat{\boldsymbol{\beta}}_n^j(\hat{\lambda}_{j-1}, \hat{\lambda}_j), \hat{\boldsymbol{\xi}}_n^j(\hat{\lambda}_{j-1}, \hat{\lambda}_j))$ be the first two segments of $\hat{\boldsymbol{\theta}}_n$. Now apply Taylor expansion

$$\tilde{L}_j^{'}(\boldsymbol{p}_j^0, \hat{\boldsymbol{\theta}}_n, \hat{\lambda}_{j-1}, \hat{\lambda}_j; \boldsymbol{X}_j) = \tilde{L}_j^{'}(\boldsymbol{p}_j^0, \boldsymbol{\theta}_n^0, \hat{\lambda}_{j-1}, \hat{\lambda}_j; \boldsymbol{X}_j) + \tilde{L}_j^{''}(\boldsymbol{p}_j^0, \boldsymbol{\theta}_n^+, \hat{\lambda}_{j-1}, \hat{\lambda}_j; \boldsymbol{X}_j)(\hat{\boldsymbol{\gamma}}_n - \boldsymbol{\gamma}_n^0),$$

where $\boldsymbol{\theta}^+ = (\boldsymbol{\gamma}_n^+, \hat{\boldsymbol{\pi}}_n), \boldsymbol{\gamma}_n^+ \in \mathbb{R}^{d_\beta + d_s}$ and $|\boldsymbol{\gamma}_n^+ - \boldsymbol{\gamma}_n^0| \leq |\hat{\boldsymbol{\gamma}}_n - \boldsymbol{\gamma}_n^0|$. And we have $\tilde{L}_j^{'}(\boldsymbol{p}_j^0, \hat{\boldsymbol{\theta}}_n, \hat{\lambda}_{j-1}, \hat{\lambda}_j; \boldsymbol{X}_j) = 0$ due to the definition. Thus we know

$$\tilde{L}_j^{'}(\boldsymbol{p}_j^0, \boldsymbol{\theta}_n^0, \hat{\lambda}_{j-1}, \hat{\lambda}_j; \boldsymbol{X}_j) = -\tilde{L}_j^{''}(\boldsymbol{p}_j^0, \boldsymbol{\theta}_n^+, \hat{\lambda}_{j-1}, \hat{\lambda}_j; \boldsymbol{X}_j)(\hat{\boldsymbol{\gamma}}_n - \boldsymbol{\gamma}_n^0). \tag{3.24}$$

Combining Lemma 3.7.1, we have

$$\begin{aligned}
&\tilde{L}_j^{'}(\boldsymbol{p}_j^0, \boldsymbol{\theta}_n^0, \hat{\lambda}_{j-1}, \hat{\lambda}_j; \boldsymbol{X}_j) = L_j^{'}(\boldsymbol{p}_j^0, \boldsymbol{\theta}_n^0, \hat{\lambda}_{j-1}, \hat{\lambda}_j; \boldsymbol{X}_j) + O(T^{1/2}) \\
&= \sum_{i=[T\hat{\lambda}_{j-1}]+1}^{[T\hat{\lambda}_j]} l_j^{'}(\boldsymbol{p}_j, \boldsymbol{\theta}_n^0, \boldsymbol{X}_{i,j} | \boldsymbol{X}_{s,j}, s < i) + O(T^{0.5}) \\
&= \sum_{i=1}^{[T\hat{\lambda}_j]} l_j^{'}(\boldsymbol{p}_j, \boldsymbol{\theta}_n^0, \boldsymbol{X}_{i,j} | \boldsymbol{X}_{s,j}, s < i) - \sum_{i=1}^{[T\hat{\lambda}_{j-1}]} l_j^{'}(\boldsymbol{p}_j, \boldsymbol{\theta}_n^0, \boldsymbol{X}_{i,j} | \boldsymbol{X}_{s,j}, s < i) + O(T^{0.5}).
\end{aligned} \tag{3.25}$$

And for any fixed $\boldsymbol{\pi}$ and $\boldsymbol{X}$, we have $f_{\boldsymbol{p}_j}(\boldsymbol{X}_{i,j} | \boldsymbol{X}_{s,j}, s < i; (0, \boldsymbol{\theta}_j^0, \boldsymbol{\pi})) = f_{\boldsymbol{p}_s}(\boldsymbol{X}_{i,j} | \boldsymbol{X}_{s,j}, s < i; \boldsymbol{\theta}_j^0)$ almost everywhere. So we have $E_{\boldsymbol{p}_j^0, \boldsymbol{\theta}_j^0}(l_j^{'}(\boldsymbol{p}_j, (0, \boldsymbol{\theta}_j^0, \boldsymbol{\pi}), \boldsymbol{X}_{i,j} | \boldsymbol{X}_{s,j}, s < i)) = 0$. Following [82], we know that both $\sum_{i=1}^{[T\hat{\lambda}_j]} l_j^{'}(\boldsymbol{p}_j, \boldsymbol{\theta}_n^0, \boldsymbol{X}_{i,j} | \boldsymbol{X}_{s,j}, s < i)$ and $\sum_{i=1}^{[T\hat{\lambda}_{j-1}]} l_j^{'}(\boldsymbol{p}_j, \boldsymbol{\theta}_n^0, \boldsymbol{X}_{i,j} | \boldsymbol{X}_{s,j}, s < i) + O(T^{0.5})$ are of order $O(\sqrt{T \log\log(T)})$.

Thus, we have $\tilde{L}_j^{'}(\boldsymbol{p}_j^0, \boldsymbol{\theta}_n^0, \hat{\lambda}_{j-1}, \hat{\lambda}_j; \boldsymbol{X}_j) = O(\sqrt{T \log\log(T)})$. Due to the definition of $\boldsymbol{\gamma}_b = (\mathbf{0}, \boldsymbol{\theta}_j^0)$, $L_j^{''}(\boldsymbol{p}_j, (\mathbf{0}, \boldsymbol{\theta}_j^0, \boldsymbol{\pi}))$ is positive definite. Combining all the above together, we have (3.23). □

Generally speaking, Lemma 3.7.1 guarantees a good asymptotic property for the log-likelihood based on partially observed data. Lemma 3.7.2 guarantees the good identifiability of model orders in each segment as well as controlling the differences between estimated model coefficients and true ones once the lag order identifiability is satisfied.

3.7.1.3. *Proof of Theorem 3.2.1.* First, when the true breakpoint number $m = m_0$ is known, the MDL criterion (3.12) for model (3.2) is

$$\begin{aligned}\frac{2}{T}\,\mathrm{MDL}(\boldsymbol{\lambda}, \mathbf{p}) =& \frac{2}{T}(\log(m+1) + (m+1)\log(T) + \sum_{j=1}^{m+1}(\log(p_{1,j}) + \log(p_{2,j})) \\ &+ \sum_{j=1}^{m+1}\frac{p_{1,j} + p_{2,j} + q + 2}{2}\log(n_j) + \sum_{j=1}^{m+1}\frac{K(n_j - p_{1,j})}{2}(\log(2\pi\hat{\sigma}_j^2) + 1) \\ =& O(\frac{\log(T)}{T}) + \frac{2}{T}\sum_{j=1}^{m+1}\frac{K(n_j - p_j)}{2}\log(\hat{\sigma}_j^2) \\ \longrightarrow& O(\frac{\log(T)}{T}) + \sum_{j=1}^{m+1}K(\lambda_j - \lambda_{j-1})\log(\hat{\sigma}_j^2(\lambda_{j-1}, \lambda_j, p_j)).\end{aligned} \tag{3.26}$$

Recall the definition of B in (3.18). We shall show by contradiction that for any $\omega \in B$, $\hat{\boldsymbol{\lambda}} \longrightarrow \boldsymbol{\lambda}^0$. Now suppose for any $\omega \in B$, $\hat{\boldsymbol{\lambda}} \nrightarrow \boldsymbol{\lambda}^0$. Then we can assume there exists a subsequence $\{t_k\}$ such that $\hat{\boldsymbol{\lambda}} \longrightarrow \boldsymbol{\lambda}^* \not\longrightarrow \boldsymbol{\lambda}^0$ along the subsequence. We can further assume $\hat{\boldsymbol{p}}_j \longrightarrow \boldsymbol{p}_j^*$, $\hat{\boldsymbol{\theta}}_j \longrightarrow \boldsymbol{\theta}_j^*$, and $\hat{\sigma}_j^2(\hat{\lambda}_{j-1}, \hat{\lambda}_j, \hat{\boldsymbol{p}}_j) \longrightarrow \sigma_j^{2*}$. For notation simplicity, we denote $\boldsymbol{\phi}_j = (\boldsymbol{\theta}_j, \sigma_j^2)$. Therefore, $\hat{\boldsymbol{\phi}}_j \longrightarrow \boldsymbol{\phi}_j^*$.

For any of the estimated intervals $I_j^* = (\lambda_{j-1}^*, \lambda_j^*)$, $j = 1, \ldots, m$, only one of the following two possible cases is true. The first case is that the estimated interval is a subset of a true interval: $I_j^* \subset I_k^0$, i.e. $\lambda_{h-1}^0 \le \lambda_{j-1}^* \le \lambda_j^* \le \lambda_h^0$. The second case is that the estimated interval contains $g \ge 0$ true intervals: $\lambda_{h-1}^0 \le \lambda_{j-1}^* < \lambda_h^0 <, \cdots, < \lambda_j^* \le \lambda_{h+g+1}^0$. We will consider these two cases separately.

Case 1. This case assumes $\lambda_{h-1} \le \lambda_{j-1}^* \le \lambda_j^* \le \lambda_h$, in particular, we consider the case that $\lambda_{h-1} < \lambda_{j-1}^* \le \lambda_j^* < \lambda_h$. So by Lemma 3.7.1, we have

$$\frac{1}{T}\tilde{L}_j(\hat{\boldsymbol{p}}_j, \hat{\boldsymbol{\theta}}_j, \hat{\lambda}_{j-1}, \hat{\lambda}_j; \boldsymbol{X}_j) \xrightarrow{a.s} (\lambda_{j-1}^* - \lambda_j^*)L^{(j)}(\boldsymbol{p}_j^*, \boldsymbol{\theta}_j^*). \tag{3.27}$$

If $\boldsymbol{p}_j^* = \boldsymbol{p}_j^0$, then $\phi_j^* = \phi_h^0$. And if $\boldsymbol{p}_j^*$ underestimates $\boldsymbol{p}_j^0$, we utilize the K-L distance

$$D(f_{\boldsymbol{p}_h^0}; \theta_h^0 | f_{\boldsymbol{p}_j^*}; \boldsymbol{\theta}_j^*) = E_{\phi_h^0}(\log\frac{f_{\boldsymbol{p}_h^0}(\boldsymbol{X}_{1,j}|\boldsymbol{X}_{l,j}, l<1; \boldsymbol{\theta}_j^0)}{f_{\boldsymbol{p}_j^*}(\boldsymbol{X}_{1,j}|\boldsymbol{X}_{l,j}, l<1; \boldsymbol{\theta}_j^*)}).$$

It can be shown by Jensen's inequality that $D(f_{\boldsymbol{p}_h^0};\boldsymbol{\theta}_h^0|f_{\boldsymbol{p}_j^*};\boldsymbol{\theta}_j^*) \geq 0$ with equality only happens when $\boldsymbol{p}_j^* > \boldsymbol{p}_h^0$, i.e. $p_{1,j} > p_{1,h}^0$ and $p_{2,j} > p_{2,h}^0$. So when $\boldsymbol{p}_j^*$ underestimates $\boldsymbol{p}_h^0$,

$$L^{(h)}(\boldsymbol{p}_h^0, \boldsymbol{\theta}_h^0) > L^{(j)}(\boldsymbol{p}_j^*, \boldsymbol{\theta}_j^*). \tag{3.28}$$

To be more specific, (3.28) is

$$\hat{\sigma}^{*2}(\lambda_{j-1}^*, \lambda_j^*, \boldsymbol{p}_j^*) = (\sigma_h^0(\boldsymbol{p}_j^*))^2 \geq (\sigma_h^0)^2 \tag{3.29}$$

with equality happens only when $\boldsymbol{p}_j* > \boldsymbol{p}_h^0$.

Case 2: This case considers $\lambda_{h-1}^0 \leq \lambda_{j-1}^* < \lambda_h^0 < \cdots < \lambda_j^* \leq \lambda_{h+g+1}^0$, where $g \geq 0$. So the model in segment I_j^* is non-stationary, and we have

$$\begin{aligned}\frac{1}{T}\tilde{L}_j(\hat{\boldsymbol{p}}_j, \hat{\boldsymbol{\theta}}_j, \hat{\lambda}_{j-1}, \hat{\lambda}_j; \boldsymbol{X}_j) = &\frac{1}{T}\tilde{L}_j(\hat{\boldsymbol{p}}_j, \hat{\boldsymbol{\theta}}_j, \hat{\lambda}_{j-1}, \lambda_h^0; \boldsymbol{X}_j) \\ &+ \frac{1}{T}\sum_{l=h}^{h+g-1} \tilde{L}_j(\hat{\boldsymbol{p}}_j, \hat{\boldsymbol{\theta}}_j, \lambda_l^0, \lambda_{l+1}^0; \boldsymbol{X}_j) + \frac{1}{T}\tilde{L}_j(\hat{\boldsymbol{p}}_j, \hat{\boldsymbol{\theta}}_j, \lambda_{h+g}^0, \hat{\lambda}_j; \boldsymbol{X}_j).\end{aligned} \tag{3.30}$$

From Lemma 3.7.1 and the fact that $L^{(l)}(\boldsymbol{p}_l^0, \boldsymbol{\theta}_l^0) \geq L^{(j)}(\boldsymbol{p}_j^*, \boldsymbol{\theta}_j^*), \forall l = h, h+1, \cdots, h+g+1$, we have

$$\lim_{T\to\infty} \frac{1}{T}\tilde{L}_j(\hat{\boldsymbol{p}}_j, \hat{\boldsymbol{\theta}}_j, \hat{\lambda}_{j-1}, \lambda_h^0; \boldsymbol{X}_j) \leq (\lambda_h^0 - \lambda_{j-1}^*)L^{(h)}(\boldsymbol{p}_h^0, \boldsymbol{\theta}_h^0), \tag{3.31}$$

$$\lim_{T\to\infty} \frac{1}{T}\tilde{L}_j(\boldsymbol{p}_j^*, \boldsymbol{\theta}_j^*, \lambda_l^0, \lambda_{l+1}^0; \boldsymbol{X}_j) \leq (\lambda_{l+1}^0 - \lambda_l^0)L^{(l+1)}(\boldsymbol{p}_{l+1}^0, \boldsymbol{\theta}_{l+1}^0), \tag{3.32}$$

$$\lim_{T\to\infty} \frac{1}{T}\tilde{L}_j(\boldsymbol{p}_j^*, \boldsymbol{\theta}_j^*, \lambda_{h+g}^0, \hat{\lambda}_j; \boldsymbol{X}_j) \leq (\lambda_j^* - \lambda_{h+g}^0)L^{(h+g+1)}(\boldsymbol{p}_{h+g+1}^0, \boldsymbol{\theta}_{h+g+1}^0). \tag{3.33}$$

It is not possible that $(\boldsymbol{p}_j^*, \boldsymbol{\theta}_j^*) = (\boldsymbol{p}_l^0, \boldsymbol{\theta}_l^0)$ for all $l = 1, \cdots, g+h+1$, therefore at least one of (3.31), (3.32) and (3.33) is a strict inequality, which implies

$$\begin{aligned}\lim_{T\to\infty} \frac{1}{T}\tilde{L}_j(\hat{\boldsymbol{p}}_j, \hat{\boldsymbol{\theta}}_j, \hat{\lambda}_{j-1}, \hat{\lambda}_j; \boldsymbol{X}_j) <&(\lambda_h^0 - \lambda_{j-1}^*)L^{(h)}(\boldsymbol{p}_h^0, \boldsymbol{\theta}_h^0) + \sum_{l=h}^{h+g-1} (\lambda_{l+1}^0 - \lambda_l^0)L^{(l+1)}(\boldsymbol{p}_{l+1}^0, \boldsymbol{\theta}_{l+1}^0) \\ &+ (\lambda_j^* - \lambda_{h+g}^0)L^{(h+g+1)}(\boldsymbol{p}_{h+g+1}^0, \boldsymbol{\theta}_{h+g+1}^0).\end{aligned} \tag{3.34}$$

To be more specific, (3.34) implies

$$\hat{\sigma}_j^{*2}\left(\hat{\lambda}_{j-1}, \hat{\lambda}_j, \hat{\boldsymbol{p}}_j\right) \geq \frac{\lambda_h^0 - \lambda_{j-1}^*}{\lambda_j^* - \lambda_{j-1}^*}(\sigma_h^0)^2 + \frac{\lambda_{h+1}^0 - \lambda_h^0}{\lambda_j^* - \lambda_{j-1}^*}(\sigma_{h+1}^0)^2 + \cdots + \frac{\lambda_j^* - \lambda_{h+g}^0}{\lambda_j^* - \lambda_{j-1}^*}(\sigma_{h+g+1}^0)^2. \tag{3.35}$$

Now since m_0 is known and we assume $\hat{\boldsymbol{\lambda}}^* \not\longrightarrow \boldsymbol{\lambda}^0$, Case 2 must be true for at least one estimated interval $I_j = (\lambda_{j-1}, \lambda_j)$. Thus, by the concavity of the log function we have

(3.36)

$$
\begin{aligned}
&K(\lambda_j^* - \lambda_{j-1}^*) \log \hat{\sigma}_j^{*2} \left(\lambda_{j-1}^*, \lambda_j^*, \boldsymbol{p}_j^*\right) \\
&\quad \geq K\left(\lambda_j^* - \lambda_{j-1}^*\right) \left[\frac{\lambda_h^0 - \lambda_{j-1}^*}{\lambda_j^* - \lambda_{j-1}^*} \log(\sigma_h^0)^2 + \frac{\lambda_{h+1}^0 - \lambda_h^0}{\lambda_j^* - \lambda_{j-1}^*} \log(\sigma_{h+1}^0)^2 + \cdots + \frac{\lambda_j^* - \lambda_{h+g}^0}{\lambda_j^* - \lambda_{j-1}^*} \log(\sigma_{h+g+1}^0)^2\right] \\
&\quad = K[\left(\lambda_h^0 - \lambda_{j-1}^*\right) \log(\sigma_h^0)^2 + \left(\lambda_{h+1}^0 - \lambda_h^0\right) \log(\sigma_{h+1}^0)^2 \quad + \cdots + \left(\lambda_j^* - \lambda_{h+g}^0\right) \log(\sigma_{h+g+1}^0)^2].
\end{aligned}
$$

Therefore, it follows that

(3.37)

$$
\lim_{T\to\infty} \frac{2}{T} \mathrm{MDL}(\hat{\boldsymbol{\lambda}}, \hat{\boldsymbol{p}}) > \sum_{l=1}^{m_0+1} K\left(\lambda_l^0 - \lambda_{l-1}^0\right) \log(\sigma_l^0)^2 = \lim_{T\to\infty} \frac{2}{T} \mathrm{MDL}\left(\boldsymbol{\lambda}^0, \boldsymbol{p}^0\right) \geq \lim_{T\to\infty} \frac{2}{T} \mathrm{MDL}(\hat{\boldsymbol{\lambda}}, \hat{\boldsymbol{p}}),
$$

where the first inequality is due to (3.36), the second equality is due to (3.26) and the last inequality is due to the definition of the estimators. So this is a contradiction, which indicates $\hat{\boldsymbol{\lambda}} \longrightarrow \boldsymbol{\lambda}_0$, $\forall \omega \in B$. Thus, Theorem 3.2.1 is proved.

3.7.1.4. *Proof of Corollary 3.2.1.1.* It can be observed that the condition of known m_0 is only used once in the proof of Theorem 3.2.1 to guarantee there is at least one estimated interval I_j belonging to Case 2. In other words, once Case 2 is applied, contradiction (3.37) arises. And it is easy to verify that when $\hat{m} < m_0$, Case 2 applies and also, when $\hat{m} > m_0$ but $\boldsymbol{\lambda}^0$ is not a subset of the limit points of $\hat{\boldsymbol{\lambda}}$, Case 2 is also applied to at least one estimated interval. So Results 1 and 2 in Corollary 3.2.1.1 can be proved similarly by contradiction.

For Result 3, if $\hat{\boldsymbol{p}}_j$ underestimates $\boldsymbol{p}_h^0$, then (3.29) can be applied, and contradiction (3.37) appears again, which completes the proof.

3.7.2. Proof of Theorem 3.2.2. The main idea of this proof is also by contradiction. First, fix $\omega \in B$, where B is a probability one set defined in (3.18). Suppose now that $\hat{m} \not\to m_0$. Since the number of breakpoints is bounded, we can assume there exists a subsequence n_k such that $\hat{m} \to m^* \neq m_0$ for a sufficiently large k. From Corollary 3.2.1.1, $m^* > m_0$. As the relative breakpoints $\in [0, 1]$, we can assume there exists a limiting partition $\boldsymbol{\lambda}^*$ such that $\hat{\boldsymbol{\lambda}} \to \boldsymbol{\lambda}^* := (\lambda_1^*, \cdots, \lambda_{m^*}^*)$. From Corollary 3.2.1.1 again, we know that $\boldsymbol{\lambda}^0$ is a subset of $\boldsymbol{\lambda}^*$. Thus, every segment of $\boldsymbol{\lambda}^*$ is

contained in exactly one of the true segments. As the number of models in family $\mathcal{M}$ is finite, by taking another subsequence, we can assume $\hat{\boldsymbol{p}}_j = \boldsymbol{p}_j^*$ for a sufficiently large n_k. And using Corollary 3.2.1.1, $\boldsymbol{p}_j^*$ is no less than the dimension of the true model. To simplify the notation, below we replace n_k with n.

For large enough n, the MDL criterion for the piecewise stationary NAR model is

$$C_1 - \sum_{j=1}^{m^*+1} \tilde{L}_j(\boldsymbol{p}_j, \boldsymbol{\theta}_j, \hat{\lambda}_{j-1}, \hat{\lambda}_j; \boldsymbol{X}_j), \tag{3.38}$$

where $C_1 = O(\frac{\log(T)}{T})$. We assume the k-th true segment contains $d > 1$ segments from $\boldsymbol{\lambda}^*$: we let them be the $(i+1)$-th to the $(i+d)$-th segments. Now, suppose we fit one model over the d segments and define

$$\begin{aligned}
\tilde{\boldsymbol{\theta}}^n &= \arg\max_{\boldsymbol{\theta}} \tilde{L}_{i+1}(\boldsymbol{p}_j^0, \boldsymbol{\theta}, \hat{\lambda}_{i-1}, \hat{\lambda}_{i+d}; \boldsymbol{X}_j) \\
\tilde{\boldsymbol{\lambda}}_n &= \{\hat{\lambda}_1, \cdots, \hat{\lambda}_i, \hat{\lambda}_{i+d}, \cdots, \hat{\lambda}_{m^*}\} \\
\tilde{\boldsymbol{p}}_n &= \{\boldsymbol{p}_1^*, \cdots, \boldsymbol{p}_i^*, \boldsymbol{p}_k^0, \boldsymbol{p}_{i+d+1}^*, \cdots, \boldsymbol{p}_{m^*+1}^*\} \\
\tilde{\boldsymbol{\theta}}_n &= \{\hat{\boldsymbol{\theta}}_1, \cdots, \tilde{\boldsymbol{\theta}}^n, \hat{\boldsymbol{\theta}}_{i+d+1}, \cdots, \hat{\boldsymbol{\theta}}_{m^*}, \hat{\boldsymbol{\theta}}_{m^*+1}\}.
\end{aligned}$$

Also, we have the MDL criterion for the model $\{m^* - d + 1, \tilde{\boldsymbol{\lambda}}_n, \tilde{\boldsymbol{p}}_n\}$ as

$$C_2 - \sum_{1 \le j \le m^*+1, j \ne i+1, \cdots, i+d} \tilde{L}_j(\boldsymbol{p}_j, \boldsymbol{\theta}_j, \hat{\lambda}_{j-1}, \hat{\lambda}_j; \boldsymbol{X}_j) - \tilde{L}_{i+1}(\boldsymbol{p}_k^0, \boldsymbol{\theta}^n, \hat{\lambda}_i, \hat{\lambda}_{i+d}; \boldsymbol{X}_i) \tag{3.39}$$

with $C_2 = O(\frac{\log(T)}{T})$. So a contradiction will occur if (3.38) minus (3.39) is positive for large enough n due to the definition.

Notice that the model that corresponds to C_1 contains more segment, therefore $C_1 - C_2 = O(\frac{\log(T)}{T}) > 0$, which means it is enough to prove

$$\tilde{L}_{i+1}(\boldsymbol{p}_k^0, \boldsymbol{\theta}^n, \hat{\lambda}_i, \hat{\lambda}_{i+d}; \boldsymbol{X}_i) - \sum_{j=i+1}^{i+d} \tilde{L}_j(\boldsymbol{p}_j, \boldsymbol{\theta}_j, \hat{\lambda}_{j-1}, \hat{\lambda}_j; \boldsymbol{X}_j) = o(\frac{\log(T)}{T}) \tag{3.40}$$

Here we consider two cases. The first case is that, for all segments, the model orders are correctly specified or $\boldsymbol{p}_j^* = \boldsymbol{p}_h^0$ if the j-th estimated segment is within the h-th true segment. From [81] and [50], it can be shown that $\hat{\boldsymbol{\theta}}_j - \boldsymbol{\theta}_h^0 \xrightarrow{a.s.} 0$ if $\boldsymbol{p}_j^* = \boldsymbol{p}_h^0$. By the definition of $\hat{\boldsymbol{\theta}}_j$, we have $\tilde{L}_j'(\boldsymbol{p}_j^*, \boldsymbol{\theta}_j^*, \hat{\lambda}_{j-1}, \hat{\lambda}_j; \boldsymbol{X}_j) = 0$ for $j = i, \cdots, i+d$. So we have the following Taylor's

expansion

(3.41)

$$\tilde{L}_j(\boldsymbol{p}_j^*, \boldsymbol{\theta}_j^*, \hat{\lambda}_{j-1}, \hat{\lambda}_j; \boldsymbol{X}_j) = \tilde{L}_j(\boldsymbol{p}_j^*, \hat{\boldsymbol{\theta}}_j, \hat{\lambda}_{j-1}, \hat{\lambda}_j; \boldsymbol{X}_j) + (\hat{\boldsymbol{\theta}}_j - \boldsymbol{\theta}_j^*)^T \tilde{L}_j''(\boldsymbol{p}_j^*, \boldsymbol{\theta}_j^+, \hat{\lambda}_{j-1}, \hat{\lambda}_j; \boldsymbol{X}_j)(\hat{\boldsymbol{\theta}}_j - \boldsymbol{\theta}_j^*),$$

where $|\boldsymbol{\theta}_j^+ - \boldsymbol{\theta}_j^*| < |\hat{\boldsymbol{\theta}}_j - \boldsymbol{\theta}_j^*|$. Similarly, we also have

(3.42)

$$\tilde{L}_{i+1}(\boldsymbol{p}_k^0, \boldsymbol{\theta}_k^0, \hat{\lambda}_i, \hat{\lambda}_{i+d}; \boldsymbol{X}_i) = \tilde{L}_{i+1}(\boldsymbol{p}_k^*, \tilde{\boldsymbol{\theta}}^n, \hat{\lambda}_i, \hat{\lambda}_{i+d}; \boldsymbol{X}_i) + (\tilde{\boldsymbol{\theta}}^n - \boldsymbol{\theta}_k^0)^T \tilde{L}_{i+1}''(\boldsymbol{p}_k^0, \tilde{\boldsymbol{\theta}}^+, \hat{\lambda}_i, \hat{\lambda}_{i+1}; \boldsymbol{X}_i)(\tilde{\boldsymbol{\theta}}_n - \boldsymbol{\theta}_k^0),$$

where $|\tilde{\boldsymbol{\theta}}^+ - \boldsymbol{\theta}_k^0| < |\tilde{\boldsymbol{\theta}}^n - \boldsymbol{\theta}_k^0|$.

From Proposition 1, we know that the model order cannot be underestimated for each segment. Thus

(3.43)

$$f_{\boldsymbol{p}i}(\boldsymbol{X}_{i,j}|\boldsymbol{X}_{s,j}, s < i; \boldsymbol{\theta}_i^*) = f_{\boldsymbol{p}_{i+d}}(\boldsymbol{X}_{i,j}|\boldsymbol{X}_{s,j}, s < i; \boldsymbol{\theta}_{i+d}^*) = \cdots = f_{\boldsymbol{p}_k^0}(\boldsymbol{X}_{i,j}|\boldsymbol{X}_{s,j}, s < i; \boldsymbol{\theta}_k^0).$$

So we have

$$\tilde{L}_{i+1}(\boldsymbol{p}_k^0, \boldsymbol{\theta}_k^0, \hat{\lambda}_i, \hat{\lambda}_{i+d}; \boldsymbol{X}_i) = \sum_{j=[T\hat{\lambda}_i]+1}^{[T\hat{\lambda}_{i+d}]} l_j(\boldsymbol{p}_k^0, \boldsymbol{\theta}_k^0, \boldsymbol{X}_{i,j}|\boldsymbol{X}_{s,j}, s < i). \tag{3.44}$$

Therefore from (3.41), (3.42) and (3.44), equation (3.40) becomes

(3.45)

$$(\tilde{\boldsymbol{\theta}}^n - \boldsymbol{\theta}_k^0)^T \tilde{L}_{i+1}''(\boldsymbol{p}_k^0, \tilde{\boldsymbol{\theta}}^+, \hat{\lambda}_i, \hat{\lambda}_{i+1}; \boldsymbol{X}_i)(\tilde{\boldsymbol{\theta}}_n - \boldsymbol{\theta}_k^0) - \sum_{j=i+1}^{i+d} (\tilde{\boldsymbol{\theta}}^n - \boldsymbol{\theta}_k^0)^T \frac{1}{T} \tilde{L}_{i+1}''(\boldsymbol{p}_k^0, \tilde{\boldsymbol{\theta}}^+, \hat{\lambda}_i, \hat{\lambda}_{i+1}; \boldsymbol{X}_i)(\tilde{\boldsymbol{\theta}}_n - \boldsymbol{\theta}_k^0).$$

From Lemma 3.7.2, (3.45) is of order $O(\log\log(T)/T) = o(\frac{\log(T)}{T})$. Thus (3.40) is proved.

The second case happens when the estimated order $\boldsymbol{p}_j^*$ is larger than $\boldsymbol{p}_h^0$. The proof is similar to the above by applying Taylor's expansion to the relevant part of $\hat{\boldsymbol{\theta}}_j$, and (3.45) is also of order $O(\frac{\log\log(T)}{T})$, which completes the proof.

3.7.3. Another Method for Generating First-Generation Chromsomes. Inspired by [76], this appendix introduces a different method for initializing the GA; i.e., a different method for generating the first-generation chromosomes. The idea is to first reformulate the breakpoint detection problem as a variable selection problem, then set up an appropriate lasso-type estimator that, upon

solving, can yield a solution path to the problem, and lastly use the models on the solution path as the chromosomes.

Here for model (3.2), we denote $\mathbb{V}_j := (V_{1,j}, \cdots, V_{K,j})^T \in \mathbb{R}^{K\times q}$, $\mathcal{B}_{0,j} = (\beta_{01,j}, \cdots, \beta_{0N,j})^T := \beta_{0,j}\mathbf{1} + \mathbb{V}_j\gamma_j \in \mathbb{R}^K$, and $\boldsymbol{W} = \text{diag}\{n_1^{-1}, \cdots, n_K^{-1}\}\boldsymbol{A} = (\boldsymbol{w}_1, \cdots, \boldsymbol{w}_k)^T$ as the row-normalized network. Let $G_{l,j} = \alpha_{l,j}W + \beta_{l,j}I_K$ for $l = 1, 2, \cdots, p_{\max,j}$ with the convention that zero values are included for the relationship to hold; i.e. if $p_{1,j} > p_{2,j}$, $\beta_{m,j} = 0$ for $m > p_{2,j}$, whereas if $p_{1,j} < p_{2,j}$, $\alpha_{m,j} = 0$ for $m > p_{1,j}$. So the piecewise stationary NAR$(p_{1,j}, p_{2,j})$ model in the j-th segments can be rewritten in vector form as

$$\boldsymbol{X}_{t,j} = \mathcal{B}_{0,j} + \sum_{m=1}^{p_{\max,j}} G_{m,j}\boldsymbol{X}_{t-m,j} + \boldsymbol{\varepsilon}_j, \tag{3.46}$$

where $\boldsymbol{\varepsilon}_j \sim N_K(\mathbf{0}, \sigma_j^2\boldsymbol{I}_K)$.

If all the segments have the **same** max lag order (i.e., $p_{\max,1} = p_{\max,2} = \cdots, p_{\max,m_0+1} =: p_M$), we can re-express (3.46) in matrix form

$$\begin{pmatrix} \boldsymbol{X}'_{p_M+1} \\ \boldsymbol{X}'_{p_M+2} \\ \vdots \\ \boldsymbol{X}'_T \end{pmatrix} = \begin{pmatrix} 1 & \boldsymbol{X}'_{p_M} & \cdots & \boldsymbol{X}'_1 & 0 & & \cdots \\ 1 & \boldsymbol{X}'_{p_M+1} & \cdots & \boldsymbol{X}'_2 & 1 & \boldsymbol{X}'_{p_M+1} & \cdots & \boldsymbol{X}'_2 & & 0 \\ & \vdots & & & & & \ddots & & \\ 1 & \boldsymbol{X}'_{T-1} & \cdots & \boldsymbol{X}'_{T-p_M} & 1 & \boldsymbol{X}'_{T-1} & \ldots & \boldsymbol{X}'_{T-p_M} & \cdots & 1\boldsymbol{X}'_{T-1} \cdots \boldsymbol{X}'_{T-p_M} \end{pmatrix} \begin{pmatrix} \zeta'_1 \\ \zeta'_2 \\ \vdots \\ \zeta'_n \end{pmatrix} + \begin{pmatrix} \varepsilon'_{p_M+1} \\ \varepsilon'_{p_M+2} \\ \vdots \\ \varepsilon'_T \end{pmatrix}, \tag{3.47}$$

where $n = T - p_M$ and $\boldsymbol{A}'$ is the matrix transpose of $\boldsymbol{A}$. Denote $G^{(j)} = (\mathcal{B}_{0,j}, G_{1,j}, \cdots, G_{p_M,j}) \in \mathbb{R}^{K\times K(p_M+1)}$. Now when $i = 1$, $\zeta_i = G^{(1)}$, while for $i = 2, \cdots, n$,

$$\zeta_i = \begin{cases} G^{(j+1)} - G^{(j)}, & \text{when } i = \tau_j \text{ for some } j \\ 0, & \text{otherwise.} \end{cases}$$

So the location at which $\zeta_i \neq 0$ for $i \geq 2$ is a breakpoint.

Denote the first, second, third, and fourth vector/matrix in (3.47) as, respectively, $\mathcal{X}$, $\mathcal{Z}$, $\boldsymbol{\zeta}$ and $\mathcal{E}$, so that (3.47) can be written as $\mathcal{X} = \mathcal{Z}\boldsymbol{\zeta} + \mathcal{E}$. We can further re-express it in a vector form as

$$\mathbb{X} = \mathbb{Z}\boldsymbol{\zeta} + \mathbb{E},$$

where $\mathbb{X} = vec(\mathcal{X})$, $\mathbb{Z} = I_K \otimes \mathcal{Z}$ and $\mathbb{E} = vec(\mathcal{E})$, with $\otimes$ being the Kronecker product.

Since $\boldsymbol{\zeta}$ is a sparse vector with only a few nonzero elements at the breakpoint locations, any suitable l_1-penalty methods like the lasso can be used to solve the regression problem

$$\widehat{\boldsymbol{\zeta}} = \arg\min_{\boldsymbol{\zeta}} \frac{1}{n}\|\mathbb{X} - \mathbb{Z}\boldsymbol{\zeta}\|_2^2 + \lambda_{1,n}\|\boldsymbol{\zeta}\|_1, \tag{3.48}$$

where $\lambda_{1,n}$ is a tuning parameter. By solving (3.48), we will obtain a solution path of all possible breakpoints.

Although it is not always the case that the lag order $p_{\max}$ remains the same in all different segments, we can still use this method by setting $p_M \in \{1, \cdots, P_0\}$ to obtain first-generation chromosomes. Our experience suggests that this will help with the convergence speed of the GA.

3.7.4. Additional Simulation Results. This appendix summarizes the results of regression parameter estimation for simulation Scenarios 1 and 2. For each experimental setting, we report the means and standard deviations of the estimated parameter values for those simulation runs with the correct estimated lag order. The results are shown in Tables 3.16 to 3.18. In addition, we also report the relative absolute estimation errors (RAEs) for each of the three network structures, where RAE is defined as $\sum_{j,l} |(\theta_{jl} - \hat{\theta}_{jl})/\theta_{jl}|$ with θ_{jl} and $\hat{\theta}_{jl}$ being, respectively, the l-th entry of $\boldsymbol{\theta}_j$ and $\hat{\boldsymbol{\theta}}_j$. From these tables, one can see that the estimated parameters are very close to the corresponding true values with small standard deviations.

TABLE 3.16. The means and standard deviations (SDs) of the estimated regression coefficients under different network structures for simulation Scenario 1. The relative absolute estimation errors (RAEs) are also listed.

	True Value	Mean	SD	RAE
Power-Law Structure	$\boldsymbol{\theta}_1$ = (0, -0.1, 0.2, 0.1, 0.4, 0.1, 0.2)	(-0.001, -0.119, 0.207, 0.100, 0.382, 0.111, 0.199)	(0.004, 0.029, 0.026, 0.016, 0.029, 0.018, 0.015)	0.075
	$\boldsymbol{\theta}_2$ = (0, 0.2, -0.22, -0.12, 0.4, -0.1, 0.1, 0.2, -0.1)	(0.001, 0.198, -0.212, -0.125, 0.389, -0.104, 0.091, 0.205, -0.088)	(0.003, 0.030, 0.031, 0.036, 0.020, 0.013, 0.034, 0.037, 0.031)	
	$\boldsymbol{\theta}_3$ = (0, -0.12, 0.1, 0.25, -0.4, 0.2, -0.5, 0.1, 0.1)	(0.000, -0.131, 0.110, 0.246, -0.396, 0.197, -0.502, 0.102, 0.105)	(0.003, 0.031, 0.033, 0.0229, 0.019, 0.011, 0.034, 0.026, 0.023)	
Dyad Independence Structure	$\boldsymbol{\theta}_1$ = (0, -0.1, 0.2, 0.1, 0.4, 0.1, 0.2)	(0.000, -0.091, 0.199, 0.105, 0.402, 0.098, 0.197)	(0.002, 0.027, 0.023, 0.029, 0.026, 0.017, 0.021)	0.057
	$\boldsymbol{\theta}_2$ = (0, 0.2, -0.22, -0.12, 0.4, -0.1, 0.1, 0.2, -0.1)	(0.000, 0.181, -0.251, -0.128, 0.387, -0.102, 0.104, 0.210, -0.104)	(0.001, 0.043, 0.062, 0.026, 0.024, 0.023, 0.011, 0.035, 0.035)	
	$\boldsymbol{\theta}_3$ = (0, -0.12, 0.1, 0.25, -0.4, 0.2, -0.5, 0.1, 0.1)	(-0.001, -0.122, 0.091, 0.237, -0.398, 0.184, -0.496, 0.110, 0.092)	(0.002, 0.063, 0.039, 0.015, 0.023, 0.015, 0.014, 0.014, 0.020)	
Stochastic Block Structure	$\boldsymbol{\theta}_1$ = (0, -0.1, 0.2, 0.1, 0.4, 0.1, 0.2)	(0.001, -0.105, 0.200, 0.087, 0.406, 0.105, 0.196)	(0.002, 0.014, 0.021, 0.021, 0.015, 0.033, 0.021)	0.056
	$\boldsymbol{\theta}_2$ = (0, 0.2, -0.22, -0.12, 0.4, -0.1, 0.1, 0.2, -0.1)	(0.000, 0.190, -0.228, -0.129, 0.388, -0.104, 0.108, 0.210, -0.107)	(0.002, 0.022, 0.029, 0.023, 0.016, 0.024, 0.024, 0.016, 0.023)	
	$\boldsymbol{\theta}_3$ = (0, -0.12, 0.1, 0.25, -0.4, 0.2, -0.5, 0.1, 0.1)	(0.000, -0.122, 0.106, 0.238, -0.398, 0.197, -0.494, 0.093, 0.106)	(0.003, 0.026, 0.027, 0.024, 0.024, 0.034, 0.027, 0.023, 0.018)	

TABLE 3.17. Similar to Table 3.16 but for simulation Scenario 2 with breakpoints $\mathcal{T} = (100, 200)$.

	True Value	Mean	SD	RAE
Power-Law Structure	$\boldsymbol{\theta}_1$ = (0, -0.1, 0.2, 0.1, 0.4, 0.1, 0.2)	(0.000, -0.102, 0.187, 0.096, 0.412, 0.104, 0.197)	(0.003, 0.029, 0.019, 0.027, 0.032, 0.036, 0.034)	0.059
	$\boldsymbol{\theta}_2$ = (0, 0.2, -0.22, -0.12, -0.1, 0.1, 0.2, -0.1)	(0.000, 0.199, -0.223, -0.107, -0.090, 0.098, 0.192, -0.092)	(0.003, 0.026, 0.035, 0.032, 0.023, 0.026, 0.043, 0.029)	
	$\boldsymbol{\theta}_3$ = (0, -0.12, 0.25, -0.4, 0.2, -0.5, 0.1, 0.1)	(-0.001, -0.123, 0.244, -0.406, 0.204, -0.503, 0.094, 0.112)	(0.004, 0.025, 0.026, 0.016, 0.025, 0.032, 0.053, 0.029)	
Dyad Independence Structure	$\boldsymbol{\theta}_1$ = (0, -0.1, 0.2, 0.1, 0.4, 0.1, 0.2)	(-0.001, -0.090 , 0.208, 0.098, 0.406, 0.097, 0.201)	(0.002, 0.046, 0.019, 0.020, 0.031, 0.031, 0.025)	0.049
	$\boldsymbol{\theta}_2$ = (0, 0.2, -0.22, -0.12, -0.1, 0.1, 0.2, -0.1)	(0.000 , 0.165, -0.262, -0.124, -0.098, 0.090, 0.197, -0.098)	(0.002, 0.056, 0.049, 0.021, 0.018, 0.032, 0.023, 0.022)	
	$\boldsymbol{\theta}_3$ = (0, -0.12, 0.25, -0.4, 0.2, -0.5, 0.1, 0.1)	(0.001, -0.113, 0.237, -0.401, 0.201 -0.500 0.094 0.100)	(0.002, 0.063, 0.030, 0.027, 0.018, 0.034, 0.026, 0.021)	
Stochastic Block Structure	$\boldsymbol{\theta}_1$ = (0, -0.1, 0.2, 0.1, 0.4, 0.1, 0.2)	(0.002, -0.099, 0.202, 0.087, 0.392, 0.094, 0.203)	(0.004, 0.018, 0.013, 0.035, 0.023, 0.029, 0.019)	0.060
	$\boldsymbol{\theta}_2$ = (0, 0.2, -0.22, -0.12, -0.1, 0.1, 0.2, -0.1)	(-0.001, 0.215, -0.205, -0.125, -0.079, 0.094, 0.214, -0.118)	(0.003, 0.029, 0.027, 0.025, 0.033, 0.019, 0.024, 0.020)	
	$\boldsymbol{\theta}_3$ = (0, -0.12, 0.25, -0.4, 0.2, -0.5, 0.1, 0.1)	(0.000, -0.117, 0.250, -0.414, 0.201, -0.512, 0.109, 0.094)	(0.003, 0.015, 0.023, 0.030, 0.034, 0.023, 0.018, 0.023)	

TABLE 3.18. Similar to Table 3.16 but for simulation Scenario 2 with breakpoints $\mathcal{T} = (50, 250)$.

	True Value	Mean	SD	RAE
Power-Law Structure	$\boldsymbol{\theta}_1$ = (0, -0.1, 0.2, 0.1, 0.4, 0.1, 0.2)	(-0.001, -0.101, 0.178, 0.103, 0.426, 0.082, 0.198)	(0.004, 0.038, 0.032, 0.030, 0.032, 0.040, 0.022)	0.088
	$\boldsymbol{\theta}_2$ = (0, 0.2, -0.22, -0.12, -0.1, 0.1, 0.2, -0.1)	(0.000, 0.204, -0.222, -0.126, -0.099, 0.104, 0.200, -0.104)	(0.001, 0.023, 0.027, 0.016, 0.013, 0.008, 0.019, 0.013)	
	$\boldsymbol{\theta}_3$ = (0, -0.12, 0.25, -0.4, 0.2, -0.5, 0.1, 0.1)	(-0.001, -0.117, 0.249, -0.403, 0.209, -0.503, 0.093, 0.109)	(0.004, 0.035, 0.029, 0.034, 0.037, 0.028, 0.039, 0.042)	
Dyad Independence Structure	$\boldsymbol{\theta}_1$ = (0, -0.1, 0.2, 0.1, 0.4, 0.1, 0.2)	(0.001, -0.128, 0.181, 0.103, 0.416, 0.103, 0.208)	(0.004, 0.066, 0.038, 0.043, 0.039, 0.034, 0.045)	0.118
	$\boldsymbol{\theta}_2$ = (0, 0.2, -0.22, -0.12, -0.1, 0.1, 0.2, -0.1)	(0.001, 0.194, -0.228, -0.119, -0.111, 0.105, 0.201, -0.104)	(0.002, 0.027, 0.038, 0.018, 0.025, 0.020, 0.011, 0.016)	
	$\boldsymbol{\theta}_3$ = (0, -0.12, 0.25, -0.4, 0.2, -0.5, 0.1, 0.1)	(-0.002, -0.114, 0.256, -0.417, 0.197, -0.525, 0.107, 0.113)	(0.004, 0.081, 0.030, 0.032, 0.047, 0.037, 0.019, 0.029)	
Stochastic Block Structure	$\boldsymbol{\theta}_1$ = (0, -0.1, 0.2, 0.1, 0.4, 0.1, 0.2)	(0.000, -0.095, 0.195, 0.107, 0.399, 0.113, 0.189)	(0.004, 0.033, 0.034, 0.026, 0.045, 0.057, 0.042)	0.063
	$\boldsymbol{\theta}_2$ = (0, 0.2, -0.22, -0.12, -0.1, 0.1, 0.2, -0.1)	(0.000, 0.193, -0.220, -0.124, -0.098, 0.099, 0.198, -0.100)	(0.002, 0.016, 0.018, 0.017, 0.014, 0.023, 0.029, 0.017)	
	$\boldsymbol{\theta}_3$ = (0, -0.12, 0.25, -0.4, 0.2, -0.5, 0.1, 0.1)	(-0.001, -0.121, 0.243, -0.398, 0.206, -0.495, 0.084, 0.097)	(0.004, 0.028, 0.037, 0.040, 0.023, 0.040, 0.067, 0.045)	

CHAPTER 4

Change Point Detection in Sequential Pairwise Comparison Data with Covariate Information

This paper introduces the novel piecewise stationary covariate-assisted ranking estimation (PS-CARE) model for analyzing time-evolving pairwise comparison data, enhancing item ranking accuracy through the integration of covariate information. By partitioning the data into distinct, stationary segments, the PS-CARE model adeptly detects temporal shifts in item rankings, known as change points, whose number and positions are initially unknown. Leveraging the minimum description length (MDL) principle, this paper establishes a statistically consistent model selection criterion to estimate these unknowns. The practical optimization of this MDL criterion is done with the pruned exact linear time (PELT) algorithm.

Empirical evaluations reveal the method's promising performance in accurately locating change points across various simulated scenarios. An application to an NBA dataset yielded meaningful insights that aligned with significant historical events, highlighting the method's practical utility and the MDL criterion's effectiveness in capturing temporal ranking changes. To the best of the authors' knowledge, this research pioneers change point detection in pairwise comparison data with covariate information, representing a significant leap forward in the field of dynamic ranking analysis.

4.1. Introdution

The ranking problem has long held a pivotal role in numerous real-life applications, spanning diverse areas such as recommendation systems [83], university admissions [84], sports analytics [85, 86] election candidate evaluations [87], and web search algorithms [88]. These rankings not only offer insights into the comparative quality of entities but also shape subsequent decisions, highlighting the problem's significance. Over the years, this has led to a collection of methods designed to tackle ranking problems, including [89], [90], [91], [92], [93], and [94].

Of all these, the Bradley-Terry-Luce (BTL) model, introduced by [95] and later expanded upon by [96], stands out due to its widespread adoption. This model postulates an intrinsic score for each item being compared. Given n items undergoing pairwise comparison, denoted by intrinsic scores $\{\theta_i\}_{i=1}^n$, the probability that item i ranks above item j is given by:

$$P(\text{item } i \text{ beats item } j) = \frac{e^{\theta_i}}{e^{\theta_i} + e^{\theta_j}}. \tag{4.1}$$

The comparison result is denoted as $y \in \{0, 1\}$. However, the classic BTL model assumes static intrinsic scores, overlooking item-related covariates. Such oversight can lead to inefficient use of important information present in the data, especially when these covariates are available and relevant. For instance, a movie recommendation system might benefit from considering variables like genre and duration, just as a National Basketball Association (NBA) match prediction could be improved by accounting for team-specific attributes such as offensive and defensive capabilities.

Recognizing this limitation, [97] introduced the covariate-assisted ranking estimation (CARE) model. This innovative approach incorporates covariate information by assuming the intrinsic score for item i is the sum of two components

$$\theta_i = \alpha_i^* + \boldsymbol{z}_i^\top \boldsymbol{\beta}^*,$$

where $\boldsymbol{z}_i \in \mathbb{R}^d$ and $\boldsymbol{\beta}^* \in \mathbb{R}^d$ are, respectively, the observed covariate vector and the coefficient vector for item i. Together, the term $\boldsymbol{z}_i^\top \boldsymbol{\beta}^*$ captures the effects of these covariates, while α_i^* represents the information that cannot be explained by the covariates. Below are more details about the CARE model. It has been shown that the CARE model performs better than the classic BTL model in predictions and inferences when useful covariate information is available.

Though CARE offers a significant advancement, many real-world applications involve sequential comparisons, which means scores might evolve over time. While several methods have been developed to address this by assuming smooth temporal transitions in the BTL model [86, 98, 99, 100], they often do not account for sudden shifts, such as an NBA team's star player's injury. A notable exception is the recent work of [101] that recognizes these abrupt changes. However, its focus was limited to the traditional BTL model, neglecting the rich covariate data.

This paper introduces a systematic approach to detect abrupt changes while simultaneously accounting for covariate information. The core concept involves partitioning the data temporally into distinct segments and fitting a separate CARE model to each. Consequently, the junctions where adjacent CARE models converge signify abrupt changes. These junctures are hereafter referred to as *change points*. The task of estimating both the number and precise locations of these change points is non-trivial. To address this, this paper employs the minimum description length (MDL) principle [29,102] as an estimator. Furthermore, the PELT algorithm [103] is harnessed for the practical estimation process. Given the past successes of both MDL and PELT in diverse change point detection challenges, it is not surprising that the proposed method exhibits both compelling theoretical and empirical strengths.

The rest of this paper is organized as follows. Section 4.2 introduces the model formulation for the piecewise stationary covariate-assisted ranking estimation (PS-CARE) model. Section 4.3 derives the MDL criterion for estimating the unknowns in the PS-CARE model. It also studies the theoretical properties of the criterion. Section 4.4 develops a PELT algorithm to minimize the MDL criterion. The empirical performance of the proposed method is illustrated in Section 4.5 via various numerical simulations and in Section 4.6 via an application to some real NBA match data. Lastly, concluding remarks are offered in Section 4.7, while technical details are provided in the appendix.

4.2. Model Formulation

This section provides the precise definition of the PS-CARE model for change point detection. We first describe the CARE model of [97].

4.2.1. The CARE Model. Recall that in the CARE model, the intrinsic core for item i is modeled as the sum of two components (4.1): one component $\boldsymbol{z}_i^\top \boldsymbol{\beta}^*$ captures the contributions of the covariates and the other component α_i^* captures the variations that cannot be explained by the covariates. With this modification, the BTL score (4.1) becomes:

$$P(\text{item } i \text{ beats item } j) = \frac{e^{\alpha_i^* + \boldsymbol{z}_i^\top \boldsymbol{\beta}^*}}{e^{\alpha_i^* + \boldsymbol{z}_i^\top \boldsymbol{\beta}^*} + e^{\alpha_j^* + \boldsymbol{z}_j^\top \boldsymbol{\beta}^*}}.$$

Additional constraints are required to make the CARE model identifiable. Write $\boldsymbol{\alpha}^* = (\alpha_1^*, \cdots, \alpha_n^*)$ and $\boldsymbol{Z} = [\boldsymbol{z}_1, \cdots, \boldsymbol{z}_n]^\top$. These additional constraints are $\sum_{i=1}^n \alpha_i^* = 0$ and $\boldsymbol{Z}^\top \boldsymbol{\alpha}^* = 0$.

Furthermore, we collect all the parameters in $\boldsymbol{\xi} = (\boldsymbol{\alpha}^*, \boldsymbol{\beta}^{*\top})^\top$, and denote $\boldsymbol{w_i} = [1, \boldsymbol{z}_i]^\top$ and $\boldsymbol{W} = [\boldsymbol{w_1}, \cdots, \boldsymbol{w_n}]^\top$. With these, the constrained parameter set is defined as $\Theta_n = \{(\boldsymbol{\alpha}, \boldsymbol{\beta}) : \boldsymbol{W}^\top \boldsymbol{\alpha} = \boldsymbol{0}\}$.

Efficient methods for parameter estimation and uncertainty quantification for the CARE model are provided by [97].

4.2.2. Piecewise Stationary CARE model. This subsection presents the PS-CARE model for change point detection for the ranking problem, where the pairwise comparisons are done sequentially. Briefly, a PS-CARE model is a concatenation of a sequence of different CARE models, where changes occur when one model switches to a different one.

We need some notation to proceed. We assume T pairwise comparisons are performed at T distinct time points $t = 1, \cdots, T$. For any positive integer N, we denote $[N]$ as the set $\{1, \cdots, M\}$ containing all positive integers less than or equal to N. Thus, the comparisons occurred at $t \in [T]$.

Let $\boldsymbol{\xi}(t)$ denote the value of $\boldsymbol{\xi}$ at time t. We assume there are $K \geq 1$ change points $\{\tau_1, \cdots, \tau_K\}$ such that the following conditions are satisfied:

- $1 = \tau_0 < \tau_1 < \tau_2 < \cdots < \tau_K < \tau_{K+1} = T$,
- $\tau_k \in \{1, \cdots, T\}$ for all $k = 1, \cdots, K$,
- $\boldsymbol{\xi}(t) \neq \boldsymbol{\xi}(t-1)$ if $t \in \{\tau_1, \cdots, \tau_K\}$, and
- $\boldsymbol{\xi}(t) = \boldsymbol{\xi}(t-1)$ if $t \notin \{\tau_1, \cdots, \tau_K\}$.

In other words, the K change points $\{\tau_1, \cdots, \tau_K\}$ partition the whole time span $\{1, \cdots, T\}$ into $K+1$ segments, for which the value of $\boldsymbol{\xi}(t)$ remaining within each segment. The values of K and τ_k's are unknown and need to be estimated.

We also denote the relative location of $\{\tau_1, \cdots, \tau_K\}$ to be $\boldsymbol{\lambda} = (\lambda_1, \cdots, \lambda_K)$, where $\lambda_k = \tau_k/T$ for $i = 1, \cdots, K$, and naturally we set $\lambda_0 = 0$ and $\lambda_{K+1} = 1$. Assume that at time point t

item i and item j are compared and

$$(4.2)\quad P(y_t = 1) := P(\text{item } i \text{ beats item } j \text{ at time } t) = P[y_t = 1|\boldsymbol{\xi}(t)] = \frac{e^{\alpha^*_{it}+\boldsymbol{z}_i^\top \boldsymbol{\beta}^*_t}}{e^{\alpha^*_{it}+\boldsymbol{z}_i^\top \boldsymbol{\beta}^*_t} + e^{\alpha^*_{jt}+\boldsymbol{z}_j^\top \boldsymbol{\beta}^*_t}},$$

where $(\alpha^*_{1t}, \alpha^*_{2t}, \cdots, \alpha^*_{nt}, \boldsymbol{\beta}^*_t) = \boldsymbol{\xi}(t)$. We further denote $\boldsymbol{\xi}^{(k)}_T := \boldsymbol{\xi}(t), \forall t \in \{\tau_{k-1}+1, \cdots, \tau_k\}$ (if time t belongs to the k-th segment).

Define $\tilde{\boldsymbol{z}}_i = (\boldsymbol{e}_i^\top, \boldsymbol{z}_i^\top)^\top$, where $\{\boldsymbol{e}_i\}_{i=1}^n$ represents the canonical basis vectors in $\mathbb{R}^n$. Let $\boldsymbol{z}^*_t = \tilde{\boldsymbol{z}}_i - \tilde{\boldsymbol{z}}_j$ if items $i, j \in \{1, 2, \cdots, n\}$ are compared at time t where t belongs to the k-th segment or $\tau_{k-1}+1 \le t < \tau_k$, then the log-likelihood function of observation y_t can be written as

$$(4.3)\qquad l_k(\boldsymbol{\xi}^{(k)}_T; y_t, \boldsymbol{z}^*_t) := y_t \boldsymbol{z}^*_t \boldsymbol{\xi}^{(k)}_T - \log(1 + \exp(\boldsymbol{z}^{*T}_t \boldsymbol{\xi}^{(k)}_T)).$$

The log-likelihood of all data is then given by

$$(4.4)\quad L_T(\boldsymbol{y}) = \sum_{k=1}^{K+1} \sum_{t=\tau_{k-1}+1}^{\tau_k} l_k(\boldsymbol{\xi}^{(k)}_T; y_t, \boldsymbol{z}^*_t) = \sum_{k=1}^{K+1} \sum_{t=\tau_{k-1}+1}^{\tau_k} y_t \boldsymbol{z}^*_t \boldsymbol{\xi}^{(k)}_T - \log(1 + \exp(\boldsymbol{z}^{*T}_t \boldsymbol{\xi}^{(k)}_T)).$$

The main goal is to estimate the number and locations of the change points as well as the parameters related to the intrinsic scores for the items. In other words, we want to estimate K, $\boldsymbol{\lambda}$ and $\boldsymbol{\xi}^{(k)}_T$ for $k = \{1, 2, \cdots, K+1\}$. Thus, we can obtain the ranking of n items in each segment and recover the PS-CARE model. In order to estimate all these parameters simultaneously, the minimum description length criterion is applied.

4.3. Change Point Detection using MDL

The minimum description length (MDL) principle is a popular and effective method for deriving effective selection criteria for model selection problems. Instead of assuming the data is generated from a given model, it views statistical modeling as a means of generating descriptions of the observed data. Thus, it defines the best-fitting model as the one that compresses the data into the shortest possible code length for storage, where the code length can be thought of as the number of bits needed to store the observed data. The MDL principle was proposed by Rissanen [29, 102] and has been successfully applied to solve various model selection problems such as image segmentation [66], network constructions [67], network vector autoregression models [104],

and quantile and spline regression [105, 106]. There are various versions of MDL criteria, and this paper focuses on the so-called "Two-Part MDL" [107, e.g.,]. This section derives the corresponding MDL criterion for fitting a PS-CARE model.

4.3.1. Derivation of the MDL Criterion. To store the observed comparison results $\boldsymbol{y} = \{y_1, y_2 \cdots, y_T\}$, the data can be divided into two parts: the first part is a fitted model and the second part is the corresponding residuals that cannot be explained by the fitted model. If the fitted model describes the data well, it will be more economical to store the data in this way. Denote $\text{CL}(z)$ as the code length required to store any object z; thus, we want to minimize CL("observed data"). Also, denote the whole class of PS-CARE models as $\mathcal{M}$. Lastly, denote any model in $\mathcal{M}$ as $\mathcal{F} \in \mathcal{M}$, the estimated version of $\mathcal{F}$ as $\hat{\mathcal{F}}$, and its corresponding residuals as $\hat{\mathcal{E}}$. Then we have

$$\begin{aligned} \text{CL}(\text{"observed data"}) &= \text{CL}(\text{"fitted model"}) + \text{CL}(\text{"residuals"}) \\ &= \text{CL}(\hat{\mathcal{F}}) + \text{CL}(\hat{\mathcal{E}}|\hat{\mathcal{F}}). \end{aligned} \tag{4.5}$$

Now, we need computable expressions for $\text{CL}(\hat{\mathcal{F}})$ and $\text{CL}(\hat{\mathcal{E}}|\hat{\mathcal{F}})$ and we first calculate $\text{CL}(\hat{\mathcal{F}})$. Notice that in order to completely determine a model $\mathcal{F}$ for a PS-CARE model, the parameters that we need to know including change points number K and their locations $\boldsymbol{\tau} = \{\tau_1, \cdots, \tau_K\}$. In addition, for each segment $k = 1, \ldots, K+1$, we need to know the intrinsic score related parameters $\boldsymbol{\xi}_T^{(k)} = (\alpha_{1,k}, \alpha_{2,k}, \cdots, \alpha_{n,k}, \boldsymbol{\beta}_k)$, for the k-th segment. Write $\hat{\boldsymbol{\xi}}_{\boldsymbol{T}} = (\hat{\boldsymbol{\xi}}_T^{(1)}, \cdots, \hat{\boldsymbol{\xi}}_T^{(K+1)})$. Then we have $\hat{\mathcal{F}} = (K, \boldsymbol{\tau}, \hat{\boldsymbol{\xi}}_{\boldsymbol{T}})$, so the first part code length for fitted model $\hat{\mathcal{F}}$ can be represented as

$$\text{CL}(\hat{\mathcal{F}}) = \text{CL}(K) + \text{CL}(\boldsymbol{\tau}) + \text{CL}(\hat{\boldsymbol{\xi}}_{\boldsymbol{T}}). \tag{4.6}$$

In general [102], it takes about $\log(I)$ bits to encode an unknown integer I, and it takes $\log(I_u)$ bits to encode it if we know that it has an upper bound I_u. So the first two terms on the RHS of (4.6) are

$$\text{CL}(K) = \log(K+1), \tag{4.7}$$

$$\text{CL}(\boldsymbol{\tau}) = (K+1)\log(T), \tag{4.8}$$

where the additional 1 in $\mathrm{CL}(K)$ is used to make the formula meaningful when there are no change points, i.e., $K = 0$.

It remains to calculate the last term in (4.6). To obtain $\mathrm{CL}(\hat{\boldsymbol{\xi}}_{\boldsymbol{T}})$, we need to first estimate $\hat{\boldsymbol{\xi}}_{\boldsymbol{T}}$ from model (4.2) and then encode the estimated values we calculated. For estimation, we shall use the maximum likelihood method of [97], which has been proven to possess excellent asymptotic properties. Meanwhile, for encoding the maximum likelihood estimate we obtained, we shall use the result of [102] that any (scalar) maximum likelihood estimate calculated from N observations can be effectively encoded with $\frac{1}{2}\log(N)$ bits. The maximum likelihood estimate can be obtained by running a projected gradient descent algorithm, and we shall denote the maximum likelihood estimator of $\boldsymbol{\xi}_T$ by $\hat{\boldsymbol{\xi}}_T$.

As mentioned before, to encode a scalar maximum likelihood estimate, the code length is $\frac{1}{2}\log(N)$ if N observations were used for estimation. Therefore, for the PS-CARE model, we have

$$\mathrm{CL}(\hat{\boldsymbol{\xi}}_T) = \sum_{k=1}^{K+1} \frac{n+d-1}{2} \log(n_k), \tag{4.9}$$

where $n_k = \tau_k - \tau_{k-1}$ represents the length of k-th segment.

The second and last part in (4.5) that we need to calculate is $\mathrm{CL}(\hat{\mathcal{E}}|\hat{\mathcal{F}})$, which is the residuals of the fitted model $\hat{\mathcal{F}}$. It equals the negative log (base 2) of the likelihood of the fitted model $\hat{\mathcal{F}}$ [102]. From (4.4) we have

$$\mathrm{CL}(\hat{\mathcal{E}}|\hat{\mathcal{F}}) = \sum_{k=1}^{K+1} \sum_{t=\tau_{k-1}+1}^{\tau_k} \left(-y_{t,k}\boldsymbol{z}_t^{*T}\boldsymbol{\xi}_T^{(k)} + \log\left(1+\exp(\boldsymbol{z}_t^{*T}\boldsymbol{\xi}_T^{(k)})\right)\right)\log_2 e. \tag{4.10}$$

Combining (4.7), (4.8), (4.9) and (4.10) and using logarithm base e instead of base 2, (4.5) becomes

$$\begin{aligned}
\mathrm{CL}(\text{“data”}) &= \log(K+1) + (K+1)\log(T) + \sum_{k=1}^{K+1}\left(\frac{n+d-1}{2}\log(n_k)\right) \\
&\quad + \sum_{k=1}^{K+1}\sum_{t=\tau_{k-1}+1}^{\tau_k}\left(-y_{t,k}\boldsymbol{z}_t^{*T}\boldsymbol{\xi}_T^{(k)} + \log\left(1+\exp(\boldsymbol{z}_t^{*T}\boldsymbol{\xi}_T^{(k)})\right)\right)\log_2 e \\
&:= \mathrm{MDL}(K, \boldsymbol{\tau}, \boldsymbol{\xi}_T).
\end{aligned} \tag{4.11}$$

Thus, the MDL principle suggests that the best-fitting PS-CARE model for the observed data $\boldsymbol{y} = \{y_1, y_2 \cdots, y_T\}$, is the one $\hat{\mathcal{F}} \in \mathcal{M}$ that minimizes (4.11).

4.3.2. Theoretical Properties. Denote the true number of change points as K_0 and the true locations of the change points as $\boldsymbol{\tau_0} = \{\tau_1^0, \cdots, \tau_{K_0}^0\}$. Define the true relative change points locations as $\boldsymbol{\lambda}_0 = \{\lambda_1^0, \cdots, \lambda_{K_0}^0\}$ with $\tau_k^0 = \lfloor \lambda_k^0 T \rfloor$ for $k = 1, \ldots, K_0$, where $\lfloor x \rfloor$ represents the greatest integer that is less than or equal to x. Note that the theoretical results in this subsection will be presented in terms of $\boldsymbol{\lambda}$ instead of $\boldsymbol{\tau}$ since they are equivalent.

As suggested by [73], for each segment, a sufficient number of comparisons are required to estimate the corresponding CARE model parameters adequately. For this reason, we impose the following constraint on the estimate of $\boldsymbol{\lambda}$. First, choose $\epsilon_\lambda > 0$ sufficiently small enough that $\epsilon_\lambda \ll \min_{k=1,\cdots,K_0+1}(\lambda_k^0 - \lambda_{k-1}^0)$. Then define

$$A_{\epsilon_\lambda}^K = \{(\lambda_1, \ldots, \lambda_K), 0 = \lambda_0 < \lambda_1 < \cdots < \lambda_K < \lambda_{K+1} = 1, \lambda_k - \lambda_{k-1} \geq \epsilon_\lambda, k = 1, 2, \ldots, K+1\}, \tag{4.12}$$

so we require the estimate of $\boldsymbol{\lambda}$ to be an element of $A_{\epsilon_\lambda}^K$. Under this constraint, the number of change points is also bounded by $M = \lfloor 1/\epsilon_\lambda \rfloor + 1$. As for coefficient $\boldsymbol{\xi}_T$, its constraint set is $\Theta_n = \{(\boldsymbol{\alpha}, \boldsymbol{\beta}) : \boldsymbol{W}\boldsymbol{\alpha} = \boldsymbol{0}\}$, which guarantees the model identifiability.

To obtain the maximum likelihood estimator with desirable properties for the CARE model, several other assumptions are required [97].

ASSUMPTION 2. *Denote the projected matrix as* $\mathcal{P}_{\boldsymbol{W}} := \boldsymbol{W}(\boldsymbol{W}^\top \boldsymbol{W})^{-1}\boldsymbol{W}^\top$. *Assume that there exists a positive constant* c_0 *such that*

$$\|\mathcal{P}_{\boldsymbol{W}}\|_{2,\infty} = \left\| \boldsymbol{W}\left(\boldsymbol{W}^\top \boldsymbol{W}\right)^{-1} \boldsymbol{W}^\top \right\|_{2,\infty} \leq c_0 \sqrt{(d+1)/n}. \tag{4.13}$$

Assumption 2 is called the incoherence condition that guarantees the rows of $\mathcal{P}_{\mathbf{W}}$ to be nearly balanced and the row sum of squares all of order $(d+1)/n$ or smaller.

ASSUMPTION 3. *Denote* $\tilde{\boldsymbol{z}}_i = (\boldsymbol{e}_i^\top, \boldsymbol{z}_i^\top)^\top$, *where* $\{\boldsymbol{e}_i\}_{i=1}^n$ *represents the canonical basis vectors in* $\mathbb{R}^n$ *and* $\boldsymbol{\Sigma} = \sum_{i>j} (\tilde{\boldsymbol{z}}_i - \tilde{\boldsymbol{z}}_j)(\tilde{\boldsymbol{z}}_i - \tilde{\boldsymbol{z}}_j)^\top$. *Assume there exists positive constants* c_1 *and* c_2 *such that*

$$c_2 n \leq \lambda_{\min,\perp}(\boldsymbol{\Sigma}) \leq \|\Sigma\|_{op} \leq c_1 n, \tag{4.14}$$

where $\|\Sigma\|_{op}$ *is the operator norm of* $\mathbf{\Sigma}$ *and*

$$\lambda_{\min,\perp}(\mathbf{\Sigma}) := \min\left\{\boldsymbol{u} \mid \boldsymbol{u}^\top \mathbf{\Sigma}\boldsymbol{u} \geq \mu\|\boldsymbol{u}\|_2^2 \text{ for all } \boldsymbol{u} \in \Theta_n\right\}. \tag{4.15}$$

Assumption 3 constraints the covariance matrix $\mathbf{\Sigma}$ to be well-behaved in all directions inside the parameter space Θ_n by restricting its largest and smallest eigenvalues are both of order n.

Now, we introduce a connected graph notion that describes the sampling scheme for collecting comparison data over time. Following [101], we consider a connected comparison graph $\mathcal{G} = \mathcal{G}([n], E)$ with edge set $E \subseteq E_{\text{full}} := \{(i,j) : 1 \leq i < j \leq n\}$. At each time point $t \in [T]$, an element $(i_t, j_t) \in [n] \times [n]$ is randomly selected from the edge set E to determine which two items are to be compared. This selection process is independent over time. In our work, we do not require the graph to be fully connected. That is, we do not require every item to be compared with every other item. For a specific time interval $\mathcal{I}$, we define similarly a random comparison graph $\mathcal{G}_{\mathcal{I}}(V_{\mathbf{I}}, E_{\mathbf{I}})$ with vertex set $V := [n]$ and edge set $E_{\mathcal{I}} := \{(i,j) : i \text{ and } j \text{ are compared in } \mathcal{I}\} \subset E$. This graph notation will be useful in studying the theoretical properties of the proposed method and will be first used by Assumption 4 below.

ASSUMPTION 4. *In each segment* I_k, $k = 1, \cdots, K+1$, *suppose* $L_{g,h,k}$ *represents the number of times items* g *and* h *compared in segment* I_k, $g, h \in \{1, 2, \cdots, n\}$, *thus* $\sum_{g,h\in\mathcal{G}_k} L_{g,h,k} = t_k$, *where* t_k *represents the time points of segment* I_k *and* $\mathcal{G}_k$ *represents the graph of segment* I_k. *Let* $L_{min,k} = \min_{g,h\in[n]^2}(L_{g,h,k})$. *It is required* $L_{min,k} \leq c_1 \cdot n^{c_2}$ *for any absolute constants* $c_1, c_2 > 0$. *Also, it is required that* $n \cdot \frac{L_{min,k}}{t_k} > c_p \log(n)$ *for some* $c_p > 0$ *and* $d + 1 < n$, $(d+1)\log(n) \lesssim n \cdot \frac{L_{min,k}}{t_k}$, *where the notation* $a_n \lesssim b_n$ *denotes* $a_n = O(b_n)$.

Assumption 4 guarantees in each segment, the number of comparisons for every two items should meet some lower bound related to the covariate dimensions and item numbers. This assumption also guarantees the connectivity of graph $\mathcal{G}_k$ as well as the consistency of the maximum likelihood estimator in each segment [97].

Using this assumption and (4.11), the unknown parameters are given by

$$\{\hat{K}, \hat{\boldsymbol{\lambda}}_{\boldsymbol{T}}, \hat{\boldsymbol{\xi}}_T\} = \underset{K\leq M, \boldsymbol{\xi}_T\in\Theta_n, \boldsymbol{\lambda}\in A^K_{\epsilon_\lambda}}{\arg\min} \frac{2}{T}\text{MDL}(K, \boldsymbol{\lambda}, \boldsymbol{\xi}_T). \tag{4.16}$$

THEOREM 4.3.1. *For the PS-CARE model given by* (4.2), *denote the true number of change points as* K_0 *and the true relative locations of change points as* $\boldsymbol{\lambda}^0$. *The estimate* $\hat{\boldsymbol{\lambda}}$ *defined by* (4.16)

satisfies

$$\hat{K} \xrightarrow{P} K_0, \quad \hat{\boldsymbol{\lambda}}_{\boldsymbol{T}} \xrightarrow{P} \boldsymbol{\lambda}^0. \tag{4.17}$$

Theorem 4.3.1 shows that the MDL criterion enjoys some desirable consistency properties. The detailed proof of Theorem 4.3.1 can be found in the appendix.

4.4. Practical Optimization of MDL

Due to the huge search space, it is very challenging to locate the global minimum of (4.16). Here, we propose solving this optimization problem using the PELT algorithm of [103]. It has been shown that, for a class of change point detection problems, the PELT algorithm is an exact search algorithm with linear computational cost, which makes it extremely appealing in practice.

The objective MDL criterion (4.11) can be rewritten as

$$\sum_{k=1}^{K+1} \left[\mathcal{C}(y_{(\tau_{k-1}+1):\tau_k})\right] + \gamma f(K), \tag{4.18}$$

where

$$\mathcal{C}(y_{(\tau_{k-1}+1):\tau_k}) = \left(\frac{n+d-1}{2}\log(n_k)\right) + \sum_{t=\tau_{k-1}+1}^{\tau_k} (-y_{t,k}\boldsymbol{z}_t^{*\top}\boldsymbol{\xi}_T^{(k)} + \log(1+\exp(\boldsymbol{z}_t^{*\top}\boldsymbol{\xi}_T^{(k)})))\log_2 e$$

represents the cost function for a segment and $\gamma f(K) = \log(T)(K+1)$ is the remaining penalty part.

In order to apply the PELT algorithm, one assumption needs to be satisfied: there is a constant R such that, for all $t < s < T$,

$$\mathcal{C}(y_{(t+1):s}) + \mathcal{C}(y_{(s+1):T}) + R \leq \mathcal{C}(y_{(t+1):T}). \tag{4.19}$$

In our case, we can choose $R = \frac{n+d-1}{2}\log(\frac{8*\pi}{T})$. We refer the readers to [103] for the full description of the general version of the PELT algorithm, while the version that is tailored for the current problem can be found in Algorithm 5. Following the notations of [103], in Algorithm 5 we use $F(s)$ to denote the minimization from (4.18) for data $y_{1:s}$, $cp(s)$ to denote the estimated change

point set for time point $\{1:s\}$, and R_s to denote the time points that are possible to be the last change points prior to s.

Algorithm 5: Optimize MDL criterion based on the PELT Algorithm

Input: A set of observed pairwise comparison data $(y_1, y_2, \cdots, y_T)$, $(\boldsymbol{z}_1^*, \boldsymbol{z}_2^*, \cdots, \boldsymbol{z}_T^*)$.
A prespecified constant R satisfied (4.19)
A minimum length L for each estimated segments.

Initialization: Set $F(0) = \cdots = F(L-1) = -\gamma$; $F(i) = \mathcal{C}(y_{1:i}), \forall i = L, L+1, \cdots, 2L-1$; $R_1 = R_2 = \cdots = R_{2L-1} = \{0\}, R_{2L} = \{0, L\}$.
for $\tau^* = 2L, \cdots, T$ **do**
    Calculate $F(\tau^*) = \min_{\tau \in R_{\tau^*}} \left[F(\tau) + \mathcal{C}(y_{(\tau+1):\tau^*}) + \gamma\right]$.
    Let $\tau^1 = \arg\{\min_{\tau \in R_{\tau^*}} \left[F(\tau) + \mathcal{C}(y_{(\tau+1):\tau^*}) + \gamma\right]\}$.
    Set $cp(\tau^*) = [cp(\tau^1), \tau^1]$.
    Set $R_{\tau^*+1} = \{\tau^* \cap \{\tau \in R_{\tau^*} : F(\tau) + \mathcal{C}(y_{(\tau+1):\tau^*}) + \gamma < F(\tau^*)\}\}$.
end for

Output: The change points recorded in $cp(T)$.

4.5. Simulation Results

In this section, we report the numerical results of our proposed method in different simulation settings. We follow a similar convention as in [101]. Suppose we have K change points $\{\tau_k\}_{k\in[K]}$ in the sequential comparison data, $\tau_0 = 1$. Suppose the number of objects is n and the dimension of covariates is d. For each setting, we set the comparison graph $\mathcal{G}_{\mathcal{I}}(V_{\mathbf{I}}, E_{\mathbf{I}})$ to be the complete graph and $T = (K+1)\Delta$ with the true change point located at $\tau_i = i\Delta$ for $i \in [K]$. To generate the covariates $\boldsymbol{Z}$ and the coefficients $\{\alpha^*\}$ and $\boldsymbol{\beta}^*$, we follow the same idea as in [97]. The covariates are generated independently with $(\boldsymbol{z}_i)_j \sim \text{Uniform}[-0.5, 0.5]$ for all $i \in [n], j \in [d]$. For matrix $\boldsymbol{Z} = [\boldsymbol{z}_1, \boldsymbol{z}_2 \cdots, \boldsymbol{z}_n]^\top \in \mathbb{R}^{n\times d}$, column-wise normalization is applied and then scale $\boldsymbol{x}_i$ by $\boldsymbol{x}_i/h$ so that $\max_{i\in[n]} \|\boldsymbol{x}_i\|_2/h = \sqrt{(d+1)/n}$. Such initialization guarantees the Assumptions 2 and 3 are satisfied. For $\boldsymbol{\alpha}^*$, its elements are generated uniformly from Uniform$[-0.3, 0.3]$. For $\boldsymbol{\beta}^*$, it is generated uniformly from a hypersphere $\{\beta, \|\beta\|_2 = 0.5\sqrt{n/(d+1)}\}$. Then we projected $\boldsymbol{\xi}$ onto the linear space Θ_n.

4.5.1. Setting 1: Without Covariates. In the first simulation setting, we compare our methods to the DPLR method [101] when $d = 0$, which means no covariates exist. We set $K = 3$, $n = \{10, 15, 20, 30, 40, 50\}$ and $\Delta = \{400, 500, 650\}$. Using the procedure described above, we simulate the pairwise comparison data 100 times and apply both our proposed MDL method and DPLR method to detect the number and locations of change points.

Methods	n	Δ	mean of $\hat{\boldsymbol{\tau}}$	s.e. of $\hat{\boldsymbol{\tau}}$	% of $\hat{K} = K$
MDL	10	400	(400.0, 798.2, 1200.0)	(1.2, 1.0, 0.9)	98%
DPLR			(398.9, 799.2, 1199.8)	(2.0, 3.2, 1.9)	85%
MDL	15	400	(400.0, 798.0, 1200.0)	(1.0, 1.0, 0.9)	99%
DPLR			(402.3, 800.4, 1196.2)	(2.3, 2.4, 2.4)	87%
MDL	20	400	(401.5, 798.2, 1200.3)	(0.9, 1.3, 1.7)	100%
DPLR			(399.2, 797.5, 1199.3)	(2.6, 3.3, 2.7)	80%
MDL	30	500	(398.5, 804.1, 1199.6)	(0.8, 1.6, 0.9)	96%
DPLR			(392.7, 801.9, 1201.8)	(2.6, 2.4, 3.1)	69%
MDL	40	500	(499.1, 1000.1, 1500.1)	(0.9, 1.0, 1.4)	99%
DPLR			(501.1, 999.8, 1501.1)	(2.5, 2.6, 2.6)	90%
MDL	50	650	(649.0, 1298.6, 1950.5)	(1.1, 1.5, 1.4)	99%
DPLR			(646.1, 1294.3, 1951.1)	(2.5, 2.6, 1.7)	93%

TABLE 4.1. Comparison of MDL and DPLR under different simulation settings without covariates.

The results are summarized in Table 4.1. When calculating the means and standard errors of the estimated change points, we only consider the cases where the true number of change points was estimated. From Table 4.1, we can observe that for the non-covariate settings, our proposed MDL method outperforms the DPLR method. The reason might be that the DPLR method requires precise choices of some tuning parameters that might be hard to obtain, especially when the sample size (Δ) is not large enough.

4.5.2. Setting 2: With Covariates. In this simulation setting, the parameters are $n = \{10, 15, 20, 30, 40, 50\}$, $d = 5$, and $\Delta = \{700, 800, 1100, 1300, 1300, 2000\}$. As before, we simulate the pairwise comparison data 100 times and apply both the MDL and DPLR methods to detect the number and locations of change points. Notice that the DPLR method was not designed to incorporate covariate information, and hence no such information was fed to it.

The results are summarized in Table 4.2. As expected, given that the proposed MDL method takes the covariate information into account, it produces superior performance when compared to the DPLR method.

Methods	n	d	Δ	mean of $\hat{\boldsymbol{\tau}}$	s.e. of $\hat{\boldsymbol{\tau}}$	% of $\hat{K} = K$
MDL	10	5	700	(699.2, 1400.6, 2099.2)	(0.6, 0.7, 0.4)	96%
DPLR				(697.9, 1398.6, 2096.2)	(1.3, 1.5, 1.6)	60%
MDL	15	5	800	(799.0, 1600.4, 2399.4)	(0.9, 0.5, 0.5)	97%
DPLR				(798.3, 1602.4, 2402.2)	(1.9, 1.3, 1.4)	66%
MDL	20	5	1100	(1100.0, 2200.2, 3300.3)	(0.8, 0.7, 0.6)	97%
DPLR				(1099.2, 2198.5, 3295.3)	(1.1, 1.2, 2.2)	84%
MDL	30	5	1100	(1299.8, 2599.7, 3901.0)	(0.5, 0.5, 0.6)	96%
DPLR				(1299.0, 2599.0, 3898.3)	(1.1, 1.2, 1.4)	89%
MDL	40	5	1300	(1299.5, 2599.8, 3899.8)	(0.6, 0.5, 0.6)	98%
DPLR				(1301.2, 2601.6, 3903.9)	(1.5, 1.4, 1.4)	94%
MDL	50	5	2000	(2000.8, 3999.5, 5998.1)	(0.8, 0.8, 0.7)	97%
DPLR				(2001.1, 4000.3, 6000.1)	(1.7, 1.8, 1.8)	95%

TABLE 4.2. Comparison of MDL and DPLR under different simulation settings with covariates.

4.6. Real Data Analysis

In this section, we study the game records of the NBA data as in [101]. The NBA season starts in October and ends in April the next year. One season is usually referred to as a two-year span. We focus on the same time range as in [101], which contains records from season 1980-1981 to season 2015-2016 with 24 teams that were founded before 1990. It has been shown in [101] that

the data is non-stationary and contains multiple change points. We utilize the overall mean salary, the mean 3-point shoot percentage, and the mean rebound number of each team over the selected period as exogenous covariates. These 3 covariates respectively represent, to a certain extent, the investment, attack ability, and defense ability of each team.

We then apply the MDL method to detect the number and locations of change points. We utilize the even-time point matches as the training dataset and the odd-time point matches as the test dataset. The results are summarized in Table 4.3.

Our methods detect 9 different change points, which divide the whole history into 10 time periods. The NBA facts can explain these 9 change points. To be more specific, the first and second periods, S1980-S1985 and S1986-S1989, represent the times that Larry Bird in the *Celtics* and Michael Johnson in the *Lakers* ruled this era together. In the third period, S1990-S1994, Michael Jordan in the *Bulls* won the triple crown. However, in S1994, Michael Jordan retired for the first time, and the *Rockets* won 2 champions. In S1995, Michael Jordan came back, and the *Bulls* achieved another triple crown. In S1998, Michael Jordan retired again, and Shaq and Kobe helped the *Lakers* dominate the period S1998-S2003. In S2004-S2006, the "Big 3" in *Spurs* emerged. In S2007-S2009, Kobe helped the *Lakers* win another 2 champions. In S2010-S2011, Derk in the *Mavericks* defeated the "Big 3" in the *Heat*, but in S2011-S2013, the *Heat* dominated the scene. Lastly, in S2014-S2015, Stephen Curry helped the *Warriors* take the lead.

When comparing our results with those in [101], our method detected some important events like Michael Jordan retiring for the first time in S1994 and the *Mavericks* defeated *Heat* in S2011, while the DPLR method [101] failed to detect them. Also, the MDL method has, in terms of minus log-likelihood value, a smaller loss: the loss for DPLR is approximately 8880 while the MDL's loss is approximately 8000, which is a 10% decrease.

4.7. Concluding Remarks

This paper addresses the challenge of detecting change points within sequential pairwise comparison data, incorporating covariate information for enhanced accuracy. At its core, this paper

S1980-S1985		S1986-S1989		S1990-S1994		S1995-S1997		S1998-S2003	
Celtics	1.0921	**Lakers**	1.2076	**Bulls**	0.8809	**Jazz**	1.1172	**Spurs**	0.8749
76ers	0.9512	**Pistons**	0.9508	**Spurs**	0.6787	**Bulls**	0.9546	**Lakers**	0.8284
Lakers	0.7574	**Celtics**	0.8223	**Suns**	0.6489	**Heat**	0.8756	**Kings**	0.6981
Bucks	0.7534	**Trail Blazers**	0.6566	**Jazz**	0.6255	**Lakers**	0.8113	**Mavericks**	0.5416
Nuggets	0.0798	**Bulls**	0.5303	**Knicks**	0.5510	**Trail Blazers**	0.5792	**Timberwolves**	0.4022
Trail Blazers	0.0626	**Jazz**	0.4991	**Rockets**	0.5184	**Hornets**	0.4965	**Trail Blazers**	0.3938
Suns	0.0551	**Bucks**	0.3841	**Trail Blazers**	0.4513	**Pacers**	0.4329	**Jazz**	0.3569
Spurs	0.0536	**Mavericks**	0.3534	**Cavaliers**	0.3044	**Knicks**	0.4063	**Pacers**	0.2981
Nets	0.0311	**76ers**	0.3311	**Pacers**	0.1609	**Rockets**	0.3865	**Suns**	0.1213
Pistons	-0.033	**Suns**	0.2723	**Lakers**	0.1603	**Cavaliers**	0.3706	**76ers**	0.0730
Knicks	-0.1204	**Rockets**	0.2544	**Magic**	0.1531	**Magic**	0.2291	**Hornets**	0.0625
Rockets	-0.1816	**Cavaliers**	0.1816	**Warriors**	0.0203	**Pistons**	0.2100	**Pistons**	-0.0058
Jazz	-0.2818	**Nuggets**	0.1209	**Celtics**	-0.0237	**Suns**	0.2063	**Bucks**	-0.0401
Bulls	-0.2942	**Knicks**	0.0379	**Hornets**	-0.1150	**Timberwolves**	0.0199	**Rockets**	-0.0841
Mavericks	-0.2974	**Pacers**	-0.0352	**Nets**	-0.2435	**Spurs**	-0.0685	**Heat**	-0.1403
Kings	-0.2989	**Spurs**	-0.0702	**Heat**	-0.25397	**Bucks**	-0.1756	**Knicks**	-0.1477
Warriors	-0.4193	**Warriors**	-0.0789	**Pistons**	-0.28443	**Nets**	-0.4655	**Nets**	-0.1539
Pacers	-0.5259	**Kings**	-0.7207	**Nuggets**	-0.32888	**76ers**	-0.6068	**Magic**	-0.2635
Clippers	-0.6087	**Nets**	-0.8198	**Clippers**	-0.39468	**Kings**	-0.6594	**Celtics**	-0.2675
Cavaliers	-0.7531	**Timberwolves**	-0.9295	**Kings**	-0.41518	**Warriors**	-0.7487	**Nuggets**	-0.4180
Heat	NA	**Clippers**	-0.9662	**Bucks**	-0.58409	**Celtics**	-0.8135	**Clippers**	-0.5736
Hornets	NA	**Magic**	-0.9875	**76ers**	-0.80124	**Clippers**	-0.8698	**Cavaliers**	-0.6686
Magic	NA	**Hornets**	-1.0816	**Mavericks**	-0.96357	**Mavericks**	-1.1372	**Warriors**	-0.7081
Timberwolves	NA	**Heat**	-1.1920	**Timberwolves**	-0.98895	**Nuggets**	-1.5718	**Bulls**	-1.1913

S2004-S2006		S2007-S2009		S2010-S2011m		S2011m-S2013		S2014-S2015	
Spurs	0.9037	**Lakers**	0.9619	**Bulls**	1.0465	**Spurs**	0.9457	**Warriors**	1.6399
Mavericks	0.8443	**Celtics**	0.8124	**Spurs**	0.8086	**Clippers**	0.8324	**Spurs**	1.2530
Suns	0.8190	**Cavaliers**	0.7386	**Heat**	0.6899	**Heat**	0.7754	**Clippers**	0.8556
Pistons	0.7526	**Magic**	0.7017	**Lakers**	0.6330	**Pacers**	0.5711	**Cavaliers**	0.7377
Rockets	0.2473	**Spurs**	0.5872	**Mavericks**	0.5353	**Rockets**	0.5179	**Rockets**	0.4701
Heat	0.1795	**Mavericks**	0.5297	**Celtics**	0.4928	**Warriors**	0.4497	**Mavericks**	0.4178
Cavaliers	0.1540	**Jazz**	0.4169	**Magic**	0.4406	**Nuggets**	0.4091	**Trail Blazers**	0.3920
Nuggets	0.1502	**Suns**	0.4024	**Nuggets**	0.3650	**Bulls**	0.3619	**Heat**	0.1986
Nets	0.0460	**Nuggets**	0.3997	**Trail Blazers**	0.0628	**Knicks**	0.2510	**Bulls**	0.1531
Bulls	0.0219	**Trail Blazers**	0.3177	**Rockets**	0.0303	**Mavericks**	0.2371	**Pacers**	0.1095
Kings	-0.0057	**Rockets**	0.2942	**76ers**	0.0296	**Trail Blazers**	0.1987	**Jazz**	0.0933
Lakers	-0.0220	**Hornets**	0.2866	**Pacers**	0.0008	**Nets**	0.1871	**Celtics**	0.0752
Clippers	-0.0649	**Bulls**	-0.0684	**Knicks**	-0.0209	**Lakers**	-0.1328	**Pistons**	0.0251
Jazz	-0.0926	**Pistons**	-0.2101	**Suns**	-0.0213	**Timberwolves**	-0.1340	**Hornets**	0.0007
Pacers	-0.1213	**Heat**	-0.2717	**Jazz**	-0.0385	**Suns**	-0.2172	**Bucks**	-0.0649
Timberwolves	-0.1568	**Warriors**	-0.3373	**Clippers**	-0.0771	**Jazz**	-0.2758	**Kings**	-0.3145
Warriors	-0.1936	**76ers**	-0.3645	**Hornets**	-0.3269	**Celtics**	-0.4300	**Suns**	-0.3763
Magic	-0.2521	**Pacers**	-0.3706	**Warriors**	-0.3615	**Kings**	-0.5275	**Magic**	-0.3995
76ers	-0.2864	**Bucks**	-0.5188	**Bucks**	-0.3695	**Hornets**	-0.5792	**Nuggets**	-0.4144
Hornets	-0.4496	**Kings**	-0.7237	**Pistons**	-0.6596	**Pistons**	-0.6031	**Nets**	-0.5219
Celtics	-0.4546	**Knicks**	-0.8034	**Kings**	-0.6985	**Bucks**	-0.6512	**Knicks**	-0.9580
Bucks	-0.5658	**Nets**	-0.9115	**Timberwolves**	-0.7597	**Cavaliers**	-0.7121	**Timberwolves**	-1.0322
Knicks	-0.7506	**Timberwolves**	-1.0381	**Nets**	-0.9547	**76ers**	-0.7279	**Lakers**	-1.0736
Trail Blazers	-0.8684	**Clippers**	-1.1249	**Cavaliers**	-1.0610	**Magic**	-0.9004	**76ers**	-1.4612

TABLE 4.3. Fitted scores for 24 selected teams in the seasons S1980-S2016 of NBA. It is divided into 10 time periods based on the 9 estimated change points. Within each period, the teams are ranked based on their fitted scores.

introduced the piecewise stationary covariate-assisted ranking estimation (PS-CARE) model, an innovative extension of the CARE model designed to handle these data complexities.

We developed a comprehensive methodology for accurately estimating the PS-CARE model's unknown parameters, including both the number and precise locations of change points. Central to our approach is the application of the minimum description length (MDL) principle, which facilitated the derivation of an objective criterion for parameter estimation. It has been shown the MDL estimates are statistically consistent.

The practical optimization of the MDL criterion was achieved using the PELT algorithm. Our extensive simulation experiments underscored the excellent empirical performance of our proposed methodology. When applied to an NBA dataset, our methodology not only identified meaningful results but also correlated these findings with significant historical events within the dataset's timeline, showcasing the practical relevance of our approach.

In conclusion, this paper contributes significantly to the field of dynamic ranking systems by presenting the PS-CARE model as a powerful tool for change point detection in sequential pairwise comparison data, especially when covariate information is available. The demonstrated success of the PS-CARE model, with its proven methodological rigor and empirical validation, paves the way for future research and offers valuable insights for practitioners and researchers alike.

4.8. Supplementary materials

4.8.1. Proof and Technical Details. This appendix presents technical details and the proof for Theorem 4.3.1. We will first define some notations and introduce several lemmas.

4.8.1.1. *Lemmas.* The k-th segment of $\{Y_t\}$ is modeled by a stationary time series $\boldsymbol{x_k} = \{x_{t,k}\}_{t\in\mathbb{Z}}$ such that

$$y_t = x_{t-\tau_{k-1},k}, \quad \forall \tau_{k-1}+1 \le t \le \tau_k$$

and $T_k = \tau_k - \tau_{k-1}$ for $k = 1, \cdots, K+1$.

Define $l_i(\boldsymbol{\xi}_T^{(j)}; x_{i,k}, \boldsymbol{z}_t^* | x_{s,k}, s < i)$ as the conditional log-likelihood function at time i for observations in the k-th segment, given all the past observations. And the conditional log-likelihood

of k-th segment, $\boldsymbol{x_k} = \{x_{t,k}, t = 1, 2, \cdots, T_j\}$ given all the past observations is

$$L_T^{(k)}(\boldsymbol{\xi}_T^{(k)}; \boldsymbol{x_k}) = \sum_{i=1}^{T_k} l_i(\boldsymbol{\xi}_T^{(k)}; x_{i,k}, \boldsymbol{z}_t^* | x_{s,k}, s < i).$$

However, it is impossible to observe all the past observations $\{x_{t,k}\}_{t<0}$ in practice. Denote $\boldsymbol{y_{i,k}} = (y_1, \cdots, y_{\tau_{k-1}+i-1})$ as the observed past in practice. Thus, the observed likelihood for the k-th segment is given by

$$\tilde{L}_T^{(k)}(\boldsymbol{\xi}_T^{(k)}; \boldsymbol{x_k}) = \sum_{i=1}^{T_k} l_i(\boldsymbol{\xi}_T^{(k)}; x_{i,k}, \boldsymbol{z}_t^* | \boldsymbol{y_{i,k}}).$$

Now, for estimating the parameters of (4.16), consider the situation where only a portion of the data in the k-th segment is chosen to perform the parameter estimation. Define $\lambda_l, \lambda_u \in [0, 1]$ and $\lambda_l < \lambda_u, \lambda_u - \lambda_l > \epsilon_\lambda$, where ϵ_λ is defined in (4.12). To simplify the notation, we write

$$\sup_{\lambda_l, \lambda_u} = \sup_{\lambda_l, \lambda_u \in [0,1], \lambda_u - \lambda_l > \epsilon_\lambda}.$$

Next, define the true and observed log-likelihood function based on a portion of the data in the k-th segment as follows:

$$L_T^{(k)}(\boldsymbol{\xi}_T^{(k)}, \lambda_l, \lambda_u; \boldsymbol{x_k}) = \sum_{i=[T_k\lambda_l]+1}^{T_k\lambda_u} l_i(\boldsymbol{\xi}_T^{(k)}; x_{i,k}, \boldsymbol{z}_t^* | x_{s,k}, s < i), \tag{4.8.20}$$

$$\tilde{L}_T^{(k)}(\boldsymbol{\xi}_T^{(k)}, \lambda_l, \lambda_u; \boldsymbol{x_k}) = \sum_{i=[T_k\lambda_l]+1}^{T_k\lambda_u} l_i(\boldsymbol{\xi}_T^{(k)}; x_{i,k}, \boldsymbol{z}_t^* | \boldsymbol{y_{i,k}}). \tag{4.8.21}$$

In practice, we can only use (4.8.21) instead of (4.8.20); thus, our first lemma controls the quality of the approximation.

LEMMA 4.8.0.1. *For any $k = 1, \cdots, K+1$, the first and second derivatives $L_n^{'(k)}$, $\tilde{L}_n^{'(k)}$ and $L_n^{''(k)}$, $\tilde{L}_n^{''(k)}$ with respect to $\boldsymbol{\xi}_T^{(k)}$ of the function defined in (4.8.21) and (4.8.20) satisfy*

$$\sup_{\lambda_l, \lambda_u} \sup_{\boldsymbol{\xi}_T^{(k)} \in \Theta_n} \left| \frac{1}{T} L_T^{(k)} \left(\boldsymbol{\xi}_T^{(k)}, \lambda_l, \lambda_u, \mathbf{x_k} \right) - \frac{1}{T} \tilde{L}_T^{(k)} \left(\boldsymbol{\xi}_T^{(k)}, \lambda_l, \lambda_u, \mathbf{x_k} \right) \right| = o(1), \tag{4.8.22}$$

$$\sup_{\lambda_l, \lambda_u} \sup_{\boldsymbol{\xi}_T^{(k)} \in \Theta_n} \left| \frac{1}{T} L_T^{'(k)} \left(\boldsymbol{\xi}_T^{(k)}, \lambda_l, \lambda_u, \mathbf{x_k} \right) - \frac{1}{T} \tilde{L}_T^{'(k)} \left(\boldsymbol{\xi}_T^{(k)}, \lambda_l, \lambda_u, \mathbf{x_k} \right) \right| = o(1), \tag{4.8.23}$$

$$\sup_{\lambda_l, \lambda_u} \sup_{\boldsymbol{\xi}_T^{(k)} \in \Theta_n} \left| \frac{1}{T} L_T^{''(k)} \left(\boldsymbol{\xi}_T^{(k)}, \lambda_l, \lambda_u, \mathbf{x_k} \right) - \frac{1}{T} \tilde{L}_T^{''(k)} \left(\boldsymbol{\xi}_T^{(k)}, \lambda_l, \lambda_u, \mathbf{x_k} \right) \right| = o(1). \tag{4.8.24}$$

PROOF. We shall only prove (4.8.22), while (4.8.23) and (4.8.24) can be proved using similar arguments. For the PS-CARE model defined in (4.2), we have

$$L_T^{(k)}\left(\boldsymbol{\xi}_T^{(k)}, \lambda_l, \lambda_u, \mathbf{x_k}\right) = \sum_{i=[T_k\lambda_l]+1}^{T_k\lambda_u} l_i(\boldsymbol{\xi}_T^{(k)}; x_{i,k}, \boldsymbol{z}_t^* | x_{s,k}, s<i)$$

$$= \sum_{i=[T_k\lambda_l]+1}^{T_k\lambda_u} \left(x_{i,k}\boldsymbol{z}_i^{*T}\hat{\boldsymbol{\xi}}_T^{(k,1)} - \log(1+\exp(\boldsymbol{z}_i^{*T}\hat{\boldsymbol{\xi}}_T^{(k,1)}))\right)$$

and

$$\tilde{L}_T^{(k)}\left(\boldsymbol{\xi}_T^{(k)}, \lambda_l, \lambda_u, \mathbf{x_k}\right) = \sum_{i=[T_k\lambda_l]+1}^{T_k\lambda_u} l_i(\boldsymbol{\xi}_T^{(k)}; x_{i,k}, \boldsymbol{z}_t^* | \boldsymbol{y_{i,k}})$$

$$= \sum_{i=[T_k\lambda_l]+1}^{T_k\lambda_u} \left(x_{i,k}\boldsymbol{z}_i^{*T}\hat{\boldsymbol{\xi}}_T^{(k,2)} - \log(1+\exp(\boldsymbol{z}_i^{*T}\hat{\boldsymbol{\xi}}_T^{(k,2)}))\right),$$

where $\hat{\boldsymbol{\xi}}_T^{(k,1)}$ and $\hat{\boldsymbol{\xi}}_T^{(k,2)}$ are the maximum likelihood estimators based on true past and observed past in the $j-th$ segment respectively.

So, we have

(4.8.25)

$$\sup_{\lambda_l,\lambda_u}\sup_{\boldsymbol{\xi}_T^{(k)}\in\Theta_n}\left|\frac{1}{T}L_T^{(k)}\left(\boldsymbol{\xi}_T^{(k)}, \lambda_l, \lambda_u, \mathbf{x_k}\right) - \frac{1}{T}\tilde{L}_T^{(k)}\left(\boldsymbol{\xi}_T^{(k)}, \lambda_l, \lambda_u, \mathbf{x_k}\right)\right|$$

$$\leq \sup_{\lambda_l,\lambda_u}\frac{1}{T}\sum_{i=[T_k\lambda_l]+1}^{T_j\lambda_u}\left(x_{i,k}\left|\boldsymbol{z}_i^{*T}\hat{\boldsymbol{\xi}}_T^{(k,1)} - \boldsymbol{z}_i^{*T}\hat{\boldsymbol{\xi}}_T^{(k,2)}\right| + \left|\log\left(1+\frac{\exp(\boldsymbol{z}_i^{*T}\hat{\boldsymbol{\xi}}_T^{(k,2)})) - \exp(\boldsymbol{z}_i^{*T}\hat{\boldsymbol{\xi}}_T^{(k,1)}))}{1+\exp(\boldsymbol{z}_i^{*T}\hat{\boldsymbol{\xi}}_{k,1}))}\right)\right|\right)$$

$$\leq \sup_{\lambda_l,\lambda_u}\frac{1}{T}\sum_{i=[T_k\lambda_l]+1}^{T_k\lambda_u}\left(x_{i,k}\|\boldsymbol{z}_i^{*T}\hat{\boldsymbol{\xi}}_T^{(k,1)} - \boldsymbol{z}_i^{*T}\hat{\boldsymbol{\xi}}_T^{(k,2)}\|_\infty + \log\left(1+\|\frac{\exp(\boldsymbol{z}_i^{*T}\hat{\boldsymbol{\xi}}_T^{(k,2)})) - \exp(\boldsymbol{z}_i^{*T}\hat{\boldsymbol{\xi}}_T^{(k,1)}))}{1+\exp(\boldsymbol{z}_i^{*T}\hat{\boldsymbol{\xi}}_T^{(k,1)}))}\|_\infty\right)\right).$$

Assume $\tilde{\boldsymbol{\beta}}_T^*$ be the true parameter vector of j-th segment. According to Theorem 3.1 in [97], as long as Assumptions 2 to 4 are satisfied, we have

$$\|\tilde{\boldsymbol{Z}}\hat{\boldsymbol{\xi}}_T^{(k,i)} - \tilde{\boldsymbol{Z}}\tilde{\boldsymbol{\beta}}_T^*\|_\infty = O(L_{min,k}^{-1/2}), \text{ for } i=1,2,$$

$$\frac{\left\|e^{\tilde{\boldsymbol{Z}}\hat{\boldsymbol{\xi}}_T^{(k,i)}} - e^{\widetilde{\boldsymbol{X}}\tilde{\boldsymbol{\beta}}_T^*}\right\|_\infty}{\left\|e^{\tilde{\boldsymbol{Z}}\tilde{\boldsymbol{\beta}}_T^*}\right\|_\infty} = O(L_{min,k}^{-1/2}), \text{ for } i=1,2.$$

And we have

$$\|\boldsymbol{z}_i^{*T}\hat{\boldsymbol{\xi}}_T^{(k,1)} - \boldsymbol{z}_i^{*T}\hat{\boldsymbol{\xi}}_T^{(k,2)}\|_\infty \leq \|\boldsymbol{z}_i^{*T}\hat{\boldsymbol{\xi}}_T^{(k,1)} - \boldsymbol{z}_i^{*T}\hat{\boldsymbol{\xi}}_T^*\|_\infty + \|\boldsymbol{z}_i^{*T}\hat{\boldsymbol{\xi}}_T^{(k,2)} - \boldsymbol{z}_i^{*T}\hat{\boldsymbol{\xi}}_T^*\|_\infty,$$

$$\left\|\frac{\exp(\boldsymbol{z}_i^{*T}\hat{\boldsymbol{\xi}}_T^{(k,2)})) - \exp(\boldsymbol{z}_i^{*T}\hat{\boldsymbol{\xi}}_T^{(k,1)}))}{1+\exp(\boldsymbol{z}_i^{*T}\hat{\boldsymbol{\xi}}_T^{(k,1)}))}\right\|_\infty \leq \exp(\|\boldsymbol{z}_i^{*T}\hat{\boldsymbol{\xi}}_T^{(k,1)} - \boldsymbol{z}_i^{*T}\hat{\boldsymbol{\xi}}_T^*\|_\infty) + \exp(\|\boldsymbol{z}_i^{*T}\hat{\boldsymbol{\xi}}_T^{(k,2)} - \boldsymbol{z}_i^{*T}\hat{\boldsymbol{\xi}}_T^*\|_\infty).$$

Thus (4.8.25) $= o(1)$ is satisfied. □

LEMMA 4.8.0.2. *For $k = 1, \cdots, K+1$, there exists an $\epsilon > 0$ such that*

$$\sup_{\boldsymbol{\xi}_T^{(k)} \in \Theta_n} E \left| l_k \left(\boldsymbol{\xi}_T^{(k)}; x_{1,k} \mid x_{l,k}, l < 1 \right) \right|^{\epsilon} < \infty,$$

$$\sup_{\boldsymbol{\xi}_T^{(k)} \in \Theta_n} E \left| l_k^{'} \left(\boldsymbol{\xi}_T^{(k)}; x_{1,k} \mid x_{l,k}, l < 1 \right) \right|^{\epsilon} < \infty,$$

$$\sup_{\boldsymbol{\xi}_T^{(k)} \in \Theta_n} E \left| l_k^{''} \left(\boldsymbol{\xi}_T^{(k)}; x_{1,k} \mid x_{l,k}, l < 1 \right) \right| < \infty.$$

PROOF. Since $l_k \left(\boldsymbol{\xi}_T^{(k)}; x_{1,k} \mid x_{l,k}, l < 1 \right) = x_{1,k} \boldsymbol{z}_i^{*T} \hat{\boldsymbol{\xi}}_T^{(k)} - \log(1 + \exp(\boldsymbol{z}_i^{*T} \hat{\boldsymbol{\xi}}_T^{(k)}))$, which is the probability of one item beating another item, and hence within $(0, 1)$. Thus, Lemma 4.8.0.2 is proved. □

LEMMA 4.8.0.3. *For each $k = 1, \cdots, K+1$,*

$$\sup_{\boldsymbol{\xi}_T^{(k)} \in \Theta_n} \left| \frac{1}{T} L_T^{(k)} \left(\boldsymbol{\xi}_T^{(k)}; \mathbf{x}_k \right) - L_k \left(\boldsymbol{\xi}_T^{(k)} \right) \right| \xrightarrow{a.s} 0, \tag{4.8.26}$$

$$\sup_{\boldsymbol{\xi}_T^{(k)} \in \Theta_n} \left| \frac{1}{T} L_T^{'(k)} \left(\boldsymbol{\xi}_T^{(k)}; \mathbf{x}_k \right) - L_j^{'} \left(\boldsymbol{\xi}_T^{(k)} \right) \right| \xrightarrow{a.s} 0, \tag{4.8.27}$$

$$\sup_{\boldsymbol{\xi}_T^{(k)} \in \Theta_n} \left| \frac{1}{T} L_T^{''(k)} \left(\boldsymbol{\xi}_T^{(k)}; \mathbf{x}_k \right) - L_k^{''} \left(\boldsymbol{\xi}_T^{(k)} \right) \right| \xrightarrow{a.s} 0, \tag{4.8.28}$$

where

$$L_k \left(\boldsymbol{\xi}_T^{(k)} \right) = E \left(l_k \left(\boldsymbol{\xi}_T^{(k)}; x_{1,k} \mid x_{l,k}, l < 1 \right) \right),$$

$$L_k^{'} \left(\boldsymbol{\xi}_T^{(k)} \right) = E \left(l_k^{'} \left(\boldsymbol{\xi}_T^{(k)}; x_{1,k} \mid x_{l,k}, l < 1 \right) \right),$$

$$L_k^{''} \left(\boldsymbol{\xi}_T^{(k)} \right) = E \left(l_k^{''} \left(\boldsymbol{\xi}_T^{(k)}; x_{1,k} \mid x_{l,k}, l < 1 \right) \right).$$

PROOF. Here we only prove (4.8.28), as (4.8.26) and (4.8.27) can be proved using similar arguments. Since $\{\boldsymbol{x_k}\}$ is a stationary ergodic process, we only need to prove

$$\frac{1}{T} \sum_{i=1}^{[T\lambda_k]} l_i(\boldsymbol{\xi}_T^{(k)}; x_{i,k}, \boldsymbol{z}_t^* | x_{s,k}, s < i) \xrightarrow{a.s} \lambda_k E \left(l_k \left(\boldsymbol{\xi}_T^{(k)}; x_{1,k} \mid x_{l,k}, l < 1 \right) \right).$$

This can be proved by the ergodic theorem. Let $\mathbb{Q}_{[0,1]}$ be the set of rational numbers in $[0, 1]$. For $\lambda_j \in \mathbb{Q}_{[0,1]}$,

$$\frac{1}{T} \sum_{i=1}^{[T\lambda_k]} l_i(\boldsymbol{\xi}_T^{(k)}; x_{i,k}, \boldsymbol{z}_t^* | x_{s,k}, s < i) \xrightarrow{a.s} \lambda_k E \left(l_k \left(\boldsymbol{\xi}_T^{(k)}; x_{1,k} \mid x_{l,k}, l < 1 \right) \right). \tag{4.8.29}$$

If B_λ is the set of ω's for which (4.8.29) holds, then set $B = \cap_{\lambda_k \in \mathbb{Q}_{[0,1]}} B_\lambda$ and $P(B) = 1$. Moreover, for $\omega \in B$ and any $s \in [0,1]$, choose $\lambda_1, \lambda_2 \in \mathbb{Q}_{[0,1]}$ such that $\lambda_1 \leq \lambda_k \leq \lambda_2$, hence

$$\left| \frac{1}{T} \sum_{i=1}^{[T\lambda_k]} l_i(\boldsymbol{\xi}_T^{(k)}; x_{i,k}, \boldsymbol{z}_t^* | x_{s,k}, s < i) - \frac{1}{T} \sum_{i=1}^{[T\lambda_1]} l_i(\boldsymbol{\xi}_T^{(k)}; x_{i,k}, \boldsymbol{z}_t^* | x_{s,k}, s < i) \right|$$

$$\leq \frac{1}{T} \sum_{i=[T\lambda_1]}^{[T\lambda_2]} \left| l_i(\boldsymbol{\xi}_T^{(k)}; x_{i,k}, \boldsymbol{z}_t^* | x_{s,k}, s < i) \right| \longrightarrow (\lambda_2 - \lambda_1) E \left| l_i(\boldsymbol{\xi}_T^{(k)}; x_{1,k}, \boldsymbol{z}_t^* | x_{s,k}, s < i) \right|.$$

By making $\lambda_2 - \lambda_1$ arbitrarily small, it follows from the ergodic theorem that

$$\frac{1}{T} \sum_{i=1}^{[T\lambda_k]} l_i(\boldsymbol{\xi}_T^{(k)}; x_{i,k}, \boldsymbol{z}_t^* | x_{s,k}, s < i) \longrightarrow \lambda_k E(l_i(\boldsymbol{\xi}_T^{(k)}; x_{1,k}, \boldsymbol{z}_t^* | x_{s,k}, s < i)).$$

To establish convergence on $D[0,1]$, it is suffice to show that for any $\omega \in B$, we have

$\frac{1}{T} \sum_{i=1}^{[T\lambda_k]} l_i(\boldsymbol{\xi}_T^{(k)}; x_{i,k}, \boldsymbol{z}_t^* | x_{s,k}, s < i) \longrightarrow \lambda_k E(l_i(\boldsymbol{\xi}_T^{(k)}; x_{1,k}, \boldsymbol{z}_t^* | x_{s,k}, s < i))$ uniformly on [0,1].

Since $\epsilon > 0$, we can choose $\lambda_1, \lambda_2, \cdots, \lambda_K \in \mathbb{Q}_{[0,1]}$ such that $0 = \lambda_0 < \lambda_1 < \cdots < \lambda_{K+1} = 1$, with $\lambda_i - \lambda_{i-1} < \epsilon$. Then for any $\lambda_k \in [0,1]$, $\lambda_{i-1} < \lambda_k \leq \lambda_i$ and

$$\left| \frac{1}{T} \sum_{i=1}^{[T\lambda_k]} l_i(\boldsymbol{\xi}_T^{(k)}; x_{i,k}, \boldsymbol{z}_t^* | x_{s,k}, s < i) - \lambda_k E(l_i(\boldsymbol{\xi}_T^{(k)}; x_{1,k}, \boldsymbol{z}_t^* | x_{s,k}, s < i)) \right|$$

$$\leq \left| \frac{1}{T} \sum_{i=1}^{[T\lambda_k]} l_i(\boldsymbol{\xi}_T^{(k)}; x_{i,k}, \boldsymbol{z}_t^* | x_{s,k}, s < i) - \frac{1}{T} \sum_{i=1}^{[T\lambda_{i-1}]} l_i(\boldsymbol{\xi}_T^{(k)}; x_{i,k}, \boldsymbol{z}_t^* | x_{s,k}, s < i) \right|$$

$$+ \left| \frac{1}{T} \sum_{i=1}^{[T\lambda_{i-1}]} l_i(\boldsymbol{\xi}_T^{(k)}; x_{i,k}, \boldsymbol{z}_t^* | x_{s,k}, s < i) - \lambda_{i-1} E(l_i(\boldsymbol{\xi}_T^{(k)}; x_{1,k}, \boldsymbol{z}_t^* | x_{s,k}, s < i)) \right|$$

$$+ \left| \lambda_{i-1} E(l_i(\boldsymbol{\xi}_T^{(k)}; x_{1,k}, \boldsymbol{z}_t^* | x_{s,k}, s < i)) - \lambda_k E(l_i(\boldsymbol{\xi}_T^{(k)}; x_{1,k}, \boldsymbol{z}_t^* | x_{s,k}, s < i)) \right|,$$

where the first term is bounded by

$$\frac{1}{T} \sum_{i=[T\lambda_{i-1}]}^{[T\lambda_i]} |l_i(\boldsymbol{\xi}_T^{(k)}; x_{i,k}, \boldsymbol{z}_t^* | x_{s,k}, s < i)| \longrightarrow (\lambda_i - \lambda_{i-1}) E \left| l_i(\boldsymbol{\xi}_T^{(k)}; x_{1,k}, \boldsymbol{z}_t^* | x_{s,k}, s < i) \right|$$

$$< \epsilon E \left| l_i(\boldsymbol{\xi}_T^{(k)}; x_{1,k}, \boldsymbol{z}_t^* | x_{s,k}, s < i) \right|.$$

Let T be large enough so that this term is less than $\epsilon E\left|l_i(\boldsymbol{\xi}_T^{(k)}; x_{1,k}, \boldsymbol{z}_t^*|x_{s,k}, s<i)\right|$ for $i = 1, \cdots, K$. So it follows that

$$\left|\frac{1}{T}\sum_{i=1}^{[T\lambda_k]} l_i(\boldsymbol{\xi}_T^{(k)}; x_{i,k}, \boldsymbol{z}_t^*|x_{s,k}, s<i) - \lambda_k E(l_i(\boldsymbol{\xi}_T^{(k)}; x_{1,k}, \boldsymbol{z}_t^*|x_{s,k}, s<i))\right|$$

$$< \epsilon E|l_i(\boldsymbol{\xi}_T^{(k)}; x_{1,k}, \boldsymbol{z}_t^*|x_{s,k}, s<i)| + \epsilon + \epsilon E|l_i(\boldsymbol{\xi}_T^{(k)}; x_{1,k}, \boldsymbol{z}_t^*|x_{s,k}, s<i)|.$$

Since ϵ can be arbitrarily small, (4.8.26) is proved, and (4.8.27) and (4.8.28) can be proved in a similar manner. □

LEMMA 4.8.0.4. *For the PS-CARE model defined above, we have*

$$\sup_{\lambda_l,\lambda_u}\sup_{\boldsymbol{\xi}_T^{(k)}\in\Theta_n}\left|\frac{1}{T}\tilde{L}_T^{(k)}\left(\boldsymbol{\xi}_T^{(k)}, \lambda_l, \lambda_u; \mathbf{x}_k\right) - (\lambda_l-\lambda_u)\, L_k\left(\boldsymbol{\xi}_T^{(k)}\right)\right| \xrightarrow{a.s.} 0, \tag{4.8.30}$$

$$\sup_{\lambda_l,\lambda_u}\sup_{\boldsymbol{\xi}_T^{(k)}\in\Theta_n}\left|\frac{1}{T}\tilde{L}_T^{\prime(k)}\left(\boldsymbol{\xi}_T^{(k)}, \lambda_l, \lambda_u; \mathbf{x}_k\right) - (\lambda_l-\lambda_u)\, L_k^{\prime}\left(\boldsymbol{\xi}_T^{(k)}\right)\right| \xrightarrow{a.s.} 0, \tag{4.8.31}$$

$$\sup_{\lambda_l,\lambda_u}\sup_{\boldsymbol{\xi}_T^{(k)}\in\Theta_n}\left|\frac{1}{T}\tilde{L}_T^{\prime\prime(k)}\left(\boldsymbol{\xi}_T^{(k)}, \lambda_l, \lambda_u; \mathbf{x}_k\right) - (\lambda_l-\lambda_u)\, L_k^{\prime\prime}\left(\boldsymbol{\xi}_T^{(k)}\right)\right| \xrightarrow{a.s.} 0. \tag{4.8.32}$$

PROOF. Here only prove (4.8.30), as (4.8.31) and (4.8.32) can be proved in a similar manner. From Lemma (4.8.0.1), we only need to prove $L_T^{(k)}\left(\boldsymbol{\xi}_T^{(k)}, \lambda_l, \lambda_u; \mathbf{x}_k\right)$ instead of $\tilde{L}_T^{(k)}\left(\boldsymbol{\xi}_T^{(k)}, \lambda_l, \lambda_u; \mathbf{x}_k\right)$. Let $\mathbb{Q}_{[0,1]}$ be a set of rational numbers in $[0,1]$. Then $\forall r_1, r_2 \in \mathbb{Q}_{[0,1]}$ with $r_1 < r_2$, by (4.8.0.3), we have

$$\sup_{\boldsymbol{\xi}_T^{(k)}\in\Theta_n}\left|\frac{1}{T}L_T^{(k)}\left(\boldsymbol{\xi}_T^{(k)}, \lambda_l, \lambda_u; \mathbf{x}_k\right) - (r_2-r_1)\, L_k\left(\boldsymbol{\xi}_T^{(k)}\right)\right|$$

$$= \sup_{\boldsymbol{\xi}_T^{(k)}\in\Theta_n} |r_2\left(\frac{1}{Tr_2}\sum_{i=1}^{[Tr_2]} l_i(\boldsymbol{\xi}_T^{(k)}; x_{i,k}, \boldsymbol{z}_t^*|x_{s,k}, s<i) - L_k\left(\boldsymbol{\xi}_T^{(k)}\right)\right) \tag{4.8.33}$$

$$- r_1\left(\frac{1}{Tr_1}\sum_{i=1}^{[Tr_1]} l_i(\boldsymbol{\xi}_T^{(k)}; x_{i,k}, \boldsymbol{z}_t^*|x_{s,k}, s<i) - -L_k\left(\boldsymbol{\xi}_T^{(k)}\right)\right)| \xrightarrow{a.s.} 0.$$

Let B_{r_1,r_2} be the probability one set of $\omega's$ for which (4.8.33) holds. Set

$$B = \cap_{r_1,r_2\in\mathbb{Q}_{[0,1]}} B_{r_1,r_2}.$$

It is well-known that $P(B) = 1$. Moreover for any $\omega \in B$ and any $\lambda \in [0,1]$, we can choose $r_l, r_u \in \mathbb{Q}_{[0,1]}$ such that $r_l \leq \lambda \leq r_u$. So we have

$$\begin{aligned}
&\sup_{\boldsymbol{\xi}_T^{(k)} \in \Theta_n} \left| \frac{1}{T} \sum_{i=1}^{[T\lambda]} l_i(\boldsymbol{\xi}_T^{(k)}; x_{i,k}, \boldsymbol{z}_t^* | x_{s,k}, s < i) - \frac{1}{T} \sum_{i=1}^{[Tr_l]} l_i(\boldsymbol{\xi}_T^{(k)}; x_{i,k}, \boldsymbol{z}_t^* | x_{s,k}, s < i) \right| \\
&\leq \sup_{\boldsymbol{\xi}_T^{(k)} \in \Theta_n} \frac{1}{T} \sum_{i=[Tr_l]+1}^{[Tr_u]} \left| l_i(\boldsymbol{\xi}_T^{(k)}; x_{i,k}, \boldsymbol{z}_t^* | x_{s,k}, s < i) \right| \\
&\longrightarrow (r_u - r_l) \sup_{\boldsymbol{\xi}_T^{(k)} \in \Theta_n} E \left| l_i(\boldsymbol{\xi}_T^{(k)}; x_{1,k}, \boldsymbol{z}_t^* | x_{s,k}, s < i) \right|.
\end{aligned}$$

From Lemma 4.8.0.2, we have $\sup_{\boldsymbol{\xi}_T^{(k)} \in \Theta_n} E \left| l_i(\boldsymbol{\xi}_T^{(k)}; x_{i,k}, \boldsymbol{z}_t^* | x_{s,k}, s < i) \right| < \infty$. So let $r_u - r_l < \epsilon$ where ϵ can be arbitrarily small, we have $L_T^{(k)} \left(\boldsymbol{\xi}_T^{(k)}, 0, \lambda; \mathbf{x}_k \right) \xrightarrow{a.s.} \lambda L_k \left(\boldsymbol{\xi}_T^{(k)} \right)$ uniformly in $\boldsymbol{\xi}_T^{(k)} \in \Theta_n$. With the same idea, we have

$$\sup_{\boldsymbol{\xi}_T^{(k)} \in \Theta_n} \left| \frac{1}{T} L_T^{(k)} \left(\boldsymbol{\xi}_T^{(k)}, \lambda_l, \lambda_u; \mathbf{x}_k \right) - (\lambda_l - \lambda_u) L_k \left(\boldsymbol{\xi}_T^{(k)} \right) \right| \xrightarrow{a.s.} 0 \tag{4.8.34}$$

for any λ_l and λ_u in $[0,1]$ with $\lambda_l < \lambda_u$. The next step is to show the convergence in (4.8.34) is uniform in λ_l, λ_u with $\lambda_u - \lambda_l > \epsilon_\lambda$. For any fixed positive $\epsilon < \epsilon_\lambda$, choose a large K_1 such that with $r_0, \cdots, r_{K_1} \in \mathbb{Q}_{[0,1]}$ such that $0 = r_0 < r_1 < \cdots < r_{K_1} = 1$ and $\max_{i=1,\cdots,K_1} \leq \epsilon$. Then for any $\lambda_l, \lambda_u \in [0,1]$, we can find g and h such that $g < h$ and $r_{g-1} < \lambda_l < r_g$, $r_{h-1} < \lambda_u < r_h$. Thus we have

$$\begin{aligned}
&\left| \frac{1}{T} L_T^{(k)} \left(\boldsymbol{\xi}_T^{(k)}, \lambda_l, \lambda_u; \mathbf{x}_k \right) - (\lambda_l - \lambda_u) L_k \left(\boldsymbol{\xi}_T^{(k)} \right) \right| \\
\leq &\left| \frac{1}{T} L_T^{(k)} \left(\boldsymbol{\xi}_T^{(k)}, \lambda_l, \lambda_u; \mathbf{x}_k \right) - \frac{1}{T} L_T^{(k)} \left(\boldsymbol{\xi}_T^{(k)}, r_{g-1}, r_h; \mathbf{x}_k \right) \right| \\
&+ \left| \frac{1}{T} L_T^{(k)} \left(\boldsymbol{\xi}_T^{(k)}, r_{g-1}, r_h; \mathbf{x}_k \right) - (r_h - r_{g-1}) L_k \left(\boldsymbol{\xi}_T^{(k)} \right) \right| \\
&+ \left| (r_g - r_{h-1}) L_k \left(\boldsymbol{\xi}_T^{(k)} \right) - (\lambda_l - \lambda_u) L_k \left(\boldsymbol{\xi}_T^{(k)} \right) \right|.
\end{aligned}$$

Let T be large enough and the third term is almost surely bounded by

$$\sup_{\boldsymbol{\xi}_T^{(k)} \in \Theta_n} \left| (r_g - r_{h-1}) L_k \left(\boldsymbol{\xi}_T^{(k)} \right) + (\lambda_l - \lambda_u) L_k \left(\boldsymbol{\xi}_T^{(k)} \right) \right| < 2\epsilon \sup_{\boldsymbol{\xi}_T^{(k)} \in \Theta_n} E \left| l_i(\boldsymbol{\xi}_T^{(k)}; x_{1,k}, \boldsymbol{z}_t^* | x_{s,k}, s < i) \right|.$$

By (4.8.33), the second term is bounded by ϵ for sufficiently large T. It follows that

$$\begin{aligned}
&\sup_{\lambda_l, \lambda_u} \sup_{\boldsymbol{\xi}_T^{(k)} \in \Theta_n} \left| \frac{1}{T} \tilde{L}_T^{(k)} \left(\boldsymbol{\xi}_T^{(k)}, \lambda_l, \lambda_u; \mathbf{x}_k \right) - (\lambda_l - \lambda_u) L_k \left(\boldsymbol{\xi}_T^{(k)} \right) \right| \\
&< 2\epsilon \sup_{\boldsymbol{\xi}_T^{(k)} \in \Theta_n} E \left| l_i(\boldsymbol{\xi}_T^{(k)}; x_{1,k}, \boldsymbol{z}_t^* | x_{s,k}, s < i) \right| + \epsilon + 2\epsilon \sup_{\boldsymbol{\xi}_T^{(k)} \in \Theta_n} E \left| l_i(\boldsymbol{\xi}_T^{(k)}; x_{1,k}, \boldsymbol{z}_t^* | x_{s,k}, s < i) \right|,
\end{aligned}$$

for sufficiently large T, a.s. And since ϵ can be arbitrarily small, and independent of λ_l, λ_u, thus (4.8.30) is proved. □

LEMMA 4.8.0.5. *Lemma 4.8.0.4 also holds if we substitute* $\sup_{\lambda_l,\lambda_u}$ *by* $\sup_{\underline{\lambda_l},\overline{\lambda_u}}$. *Where* $\sup_{\underline{\lambda_l},\overline{\lambda_u}} = \sup_{\substack{-h_n<\lambda_l<\lambda_u<1+k_n \\ \lambda_u-\lambda_l>\epsilon_\lambda}}$ *for any pre-specified sequence* h_n *and* k_n *which are converging to* 0 *as* $n \to \infty$.

PROOF. First define $\grave{\lambda}_l = \max(0, \lambda_l)$, $\ddot{\lambda}_l = \min(0, \lambda_l)$, $\grave{\lambda}_u = \min(1, \lambda_u)$ and $\ddot{\lambda}_u = \max(1, \lambda_u)$. Then we have

$$
\begin{aligned}
&\frac{1}{T_k} L_T^{(k)}\left(\boldsymbol{\xi}_T^{(k)}, \lambda_l, \lambda_u; \mathbf{x}_k\right) - (\lambda_l - \lambda_u) L_k\left(\boldsymbol{\xi}_T^{(k)}\right) \\
=&\frac{1}{T_k} L_T^{(k)}\left(\boldsymbol{\xi}_T^{(k)}, \grave{\lambda}_l, \grave{\lambda}_u; \mathbf{x}_k\right) - \left(\grave{\lambda}_l - \grave{\lambda}_u\right) L_k\left(\boldsymbol{\xi}_T^{(k)}\right) - (\ddot{\lambda}_u - 1 - \ddot{\lambda}_l) L_k\left(\boldsymbol{\xi}_T^{(k)}\right) \\
&+ \frac{1}{T_k} \sum_{i=T_{k-1}+[T_k(\ddot{\lambda}_l)]+1}^{T_{k-1}} l_i(\boldsymbol{\xi}_T^{(k)}; x_{i,k-1}, \boldsymbol{z}_t^* | x_{s,k-1}, s<i) \qquad (4.8.35) \\
&+ \frac{1}{T_k} \sum_{i=1}^{[T_k(\ddot{(\lambda)}_u - 1)]} l_i(\boldsymbol{\xi}_T^{(k)}; x_{i,k+1}, \boldsymbol{z}_t^* | x_{s,k+1}, s<i).
\end{aligned}
$$

Since $0 \le \grave{\lambda}_l < \grave{\lambda}_u \le 1$, the sum of the first two terms in (4.8.35) converges to 0 a.s by Lemma 4.8.0.4. And for any $\delta > 0$, $\max(|\ddot{\lambda}_l|, |\ddot{\lambda}_u - 1|) < \delta$ for sufficiently large enough T. Thus the third term in (4.8.35) is bounded by $2\delta |L_k\left(\boldsymbol{\xi}_T^{(k)}\right)|$. The fourth term is bounded by

$$
\frac{1}{T_k} \sum_{i=T_{k-1}-[T_k\delta]}^{T_{k-1}} l_i(\boldsymbol{\xi}_T^{(k)}; x_{i,k-1}, \boldsymbol{z}_t^* | x_{s,k-1}, s<i) \xrightarrow{a.s} \delta E |l_i(\boldsymbol{\xi}_T^{(k)}; x_{i,k-1}, \boldsymbol{z}_t^* | x_{s,k-1}, s<i)|.
$$

The last term in (4.8.35) can be bounded similarly. And since δ can be arbitrarily small, so (4.8.35) converges to 0 uniformly in λ_l, λ_u. □

LEMMA 4.8.0.6. *Let* $\tilde{\boldsymbol{\beta}}_k^0$ *be the true model parameter. Define*

$$
\hat{\boldsymbol{\xi}}_T^{(k,\lambda_l,\lambda_u)} = \hat{\boldsymbol{\xi}}_T^{(k)}(\lambda_l, \lambda_u) = \underset{\boldsymbol{\xi}_T^{(k)} \in \Theta_n}{\arg\max}, \tilde{L}_T^{(k)}\left(\boldsymbol{\xi}_T^{(k)}, \lambda_l, \lambda_u; \mathbf{x}_k\right),
$$

$$
\boldsymbol{\xi}_k^* = \underset{\boldsymbol{\xi}_T^{(k)} \in \Theta_n}{\arg\max} L_k\left(\boldsymbol{\xi}_T^{(k)}\right).
$$

We have

$$
\sup_{\underline{\lambda_l},\overline{\lambda_u}} \left| \frac{1}{T} L_T^{(k)}\left(\hat{\boldsymbol{\xi}}_T^{(k,\lambda_l,\lambda_u)}, \lambda_l, \lambda_u; \mathbf{x}_k\right) - (\lambda_l - \lambda_u) L_k\left(\boldsymbol{\xi}_k^*\right) \right| \xrightarrow{a.s} 0 \qquad (4.8.36)
$$

and

$$
\sup_{\underline{\lambda_l},\overline{\lambda_u}} \left| \hat{\boldsymbol{\xi}}_T^{(k)}(\lambda_l, \lambda_u) - \boldsymbol{\xi}_k^0 \right| \xrightarrow{a.s} 0. \qquad (4.8.37)
$$

PROOF. By the definition of $\hat{\boldsymbol{\xi}}_T^{(k,\lambda_l,\lambda_u)}$, we have

$$\tilde{L}_T^{(k)}\left(\hat{\boldsymbol{\xi}}_T^{(k,\lambda_l,\lambda_u)},\lambda_l,\lambda_u;\mathbf{x}_k\right)\geq\tilde{L}_T^{(k)}\left(\boldsymbol{\xi}_k^*,\lambda_l,\lambda_u;\mathbf{x}_k\right)$$

for all λ_l,λ_u,T. Combined with Lemma 4.8.0.1 and Lemma 4.8.0.3, we have

$$\begin{aligned}
&(\lambda_l-\lambda_u)\left\{L_k\left(\boldsymbol{\xi}_k^*\right)-L_k\left(\hat{\boldsymbol{\xi}}_T^{(k,\lambda_l,\lambda_u)}\right)\right\}\\
&\leq\sup_{\underline{\lambda_d},\bar{\lambda}_u}\left\{(\lambda_u-\lambda_l)\,L_k\left(\boldsymbol{\xi}_k^*\right)-\frac{1}{T}\tilde{L}_T^{(k,\lambda_l,\lambda_u)}\left(\boldsymbol{\xi}_k^*,\lambda_l,\lambda_u;\mathbf{x_k}\right)\right.\\
&\quad\left.+\frac{1}{T}\tilde{L}_T^{(k,\lambda_l,\lambda_u)}\left(\hat{\boldsymbol{\xi}}_T^{(k,\lambda_l,\lambda_u)},\lambda_l,\lambda_u;\mathbf{x}_k\right)-(\lambda_u-\lambda_l)\,L_k\left(\hat{\boldsymbol{\xi}}_T^{(k,\lambda_l,\lambda_u)}\right)\right\}\\
&=\sup_{\underline{\lambda_d},\bar{\lambda}_u}\left\{\left(\lambda_u-\lambda_l\right)L_k\left(\boldsymbol{\xi}_k^*\right)-\frac{1}{T}L_T^{(k,\lambda_l,\lambda_u)}\left(\boldsymbol{\xi}_k^*,\lambda_l,\lambda_u;\mathbf{x_k}\right)\right.\\
&\quad\left.+\frac{1}{T}L_T^{(k,\lambda_l,\lambda_u)}\left(\hat{\boldsymbol{\xi}}_T^{(k,\lambda_l,\lambda_u)},\lambda_l,\lambda_u;\mathbf{x}_k\right)-\left(\lambda_u-\lambda_l\right)L_k\left(\hat{\boldsymbol{\xi}}_T^{(k,\lambda_l,\lambda_u)}\right)\right\}+o(1)\\
&\leq 2\sup_{\underline{\lambda_d},\bar{\lambda}_u}\sup_{\boldsymbol{\xi}_T^{(k)}\in\Theta_n}\left|\frac{1}{T}\tilde{L}_T^{(k)}\left(\boldsymbol{\xi}_T^{(k)},\lambda_l,\lambda_u;\mathbf{x}_k\right)-(\lambda_u-\lambda_l)\,L_k\left(\boldsymbol{\xi}_T^{(k)}\right)\right|+o(1)\overset{a.s.}{\to}0.
\end{aligned}$$

And since $L_k(\boldsymbol{\xi}_k^*)$ is the maximum value for all $\boldsymbol{\xi}$, it follows that

$$\left|L_k(\hat{\boldsymbol{\xi}}_T^{(k,\lambda_l,\lambda_u)})-L_k(\boldsymbol{\xi}_k^*)\right|\overset{a.s.}{\to}0.$$

Thus, using Lemma 4.8.0.5, (4.8.36) is proved. Due to the identifiability of MLE for the CARE model, (4.8.37) is also proved. □

LEMMA 4.8.0.7. *Let $\boldsymbol{y}=\{y_t;t=1,\cdots,T\}$ be the observations from a PS-CARE model specified by the vector $(K_0,\boldsymbol{\lambda^0},\boldsymbol{\xi^0})$. Assume the number of change points K_0 is known. The estimator $(\hat{\boldsymbol{\lambda}}_T,\hat{\boldsymbol{\xi}}_{\boldsymbol{T}})$is defined by*

$$\{\hat{\boldsymbol{\lambda}}_T,\hat{\boldsymbol{\xi}}_{\boldsymbol{T}}\}=\underset{\boldsymbol{\xi_T}\in\Theta_n,\boldsymbol{\lambda_T}\in A_\epsilon^{K_0}}{\arg\min}\frac{2}{T}MDL(K_0,\boldsymbol{\lambda_T},\boldsymbol{\xi_T}),$$

where $A_\epsilon^{K_0}$ is defined in (4.12). Under Assumptions 2 to 4, for sufficiently large T we have

$$\hat{\boldsymbol{\lambda}}_T\xrightarrow{P}\boldsymbol{\lambda}^o.$$

PROOF. Let B be the probability one set in which Lemma 4.8.0.5 and Lemma 4.8.0.6 hold. And we will show that $\forall\omega\in B$, we have $\hat{\boldsymbol{\lambda}}_T\to\boldsymbol{\lambda}^0$ and $\hat{\boldsymbol{\xi}}_T\to\boldsymbol{\xi}_0$. We will prove this by contradiction. Here we assume $\hat{\boldsymbol{\lambda}}_T\to\boldsymbol{\lambda}^*\neq\boldsymbol{\lambda}^0$ along a subsequence $\{T_k\}$. Also, we further assume $\hat{\boldsymbol{\xi}}_T\to\boldsymbol{\xi}^*\neq\boldsymbol{\xi}_0$. To simplify the future notation, we replace T_k with T. For sufficiently large T, we have

$$\frac{2}{T}MDL(K_0,\boldsymbol{\lambda_T},\boldsymbol{\xi_T})=c_T-\frac{1}{T}\sum_{k=1}^{K+1}L_T^{(k)}\left(\hat{\boldsymbol{\xi}}_T^{(k)},\hat{\lambda}_{k-1},\hat{\lambda}_k;\mathbf{y}\right),$$

where c_T is of order $O(\log(T)/T)$. Here we simplify the notation of $\hat{\boldsymbol{\xi}}_T^{(k)}(\hat{\lambda}_l, \hat{\lambda}_u)$ to $\hat{\boldsymbol{\xi}}_T^{(k)}$ when there are no misunderstandings.

For each estimated interval, its limiting $I_k^*(\lambda_{k-1}^*, \lambda_k^*), k = 1, \cdots, K+1$, there are two possible cases. The first case is when I_k^* is totally contained in one true interval $(\lambda_{i-1}^0, \lambda_i^0)$. The second case is when I_k^* covers $m+2 (m \geq 0)$ true intervals $(\lambda_{i-1}^0, \lambda_i^0), \cdots, (\lambda_{i+m}^0, \lambda_{i+m+1}^0)$. We consider the two cases individually.

Case 1: If $\lambda_{i-1}^0 \leq \lambda_{k-1}^* \leq \lambda_k^* \leq \lambda_i^0$, in particular, we only consider the inequality case. Then if $\lambda_k^* = \lambda_i^0$ or $\lambda_{k-1}^* = \lambda_{i-1}^0$, as $\hat{\lambda}_{k-1} \to \lambda_{i-1}^0$ and $\hat{\lambda}_k \to \lambda_i^0$, the estimated segment can only include a decreasing proportion of observations from the adkacent segments. Then $\max(\hat{\lambda}_k - \lambda_i^0, 0)$ and $\max(\lambda_{i-1}^0 - \hat{\lambda}_{k-1}, 0)$ play the role of h_n and k_n in Lemma 4.8.0.5. So we have from Lemma 4.8.0.5 that

$$\frac{1}{T} L_T^{(k)} \left(\hat{\boldsymbol{\xi}}_T^{(k)}, \hat{\lambda}_{k-1}, \hat{\lambda}_k; \mathbf{y} \right) \xrightarrow{a.s.} (\lambda_k^* - \lambda_{k-1}^*) L_i \left(\boldsymbol{\xi}_i^0 \right).$$

Case 2: If $\lambda_{i-1}^o \leq \lambda_{k-1}^* < \lambda_i^o < \ldots < \lambda_{i+k}^o < \lambda_k^* \leq \lambda_{i+m+1}^o$ for some $m \geq 0$. Then, for sufficiently large T, the estimated stationary process is thus non-stationary so that we can partition the likelihood by the true segment change point as below:

$$\begin{aligned} (4.8.38) \quad \frac{1}{T} L_T^{(k)} \left(\hat{\boldsymbol{\xi}}_T^{(k)}, \hat{\lambda}_{k-1}, \hat{\lambda}_k; \mathbf{y} \right) &= \frac{1}{T} L_T^{(k)} \left(\hat{\boldsymbol{\xi}}_T^{(k)}, \hat{\lambda}_{k-1}, \lambda_i^0; \mathbf{y} \right) + \\ &\frac{1}{T} \sum_{l=i}^{i+m-1} L_T^{(k)} \left(\hat{\boldsymbol{\xi}}_T^{(k)}, \lambda_l^0, \lambda_{l+1}^0; \mathbf{y} \right) + \frac{1}{T} L_T^{(k)} \left(\hat{\boldsymbol{\xi}}_T^{(k)}, \lambda_{i+k}^0, \hat{\lambda}_k; \mathbf{y} \right). \end{aligned}$$

Each of the likelihood functions in (4.8.38) involves observations from one of the stationary segments. From Lemma 4.8.0.4 and $L_i(\boldsymbol{\xi}_l^0) \geq L_i(\boldsymbol{\xi}_k^*)$ for all $l = i, i+1/cdots, i+m+1$, we have

$$\begin{aligned} \lim_{T\to\infty} \frac{1}{T} L_T^{(k)} \left(\hat{\boldsymbol{\xi}}_T^{(k)}, \hat{\lambda}_{k-1}, \lambda_i^0; \mathbf{y} \right) &\leq \left(\lambda_i^o - \lambda_{k-1}^* \right) L_i \left(\boldsymbol{\xi}_i^0 \right), \\ \lim_{T\to\infty} \frac{1}{T} L_T^{(k)} \left(\boldsymbol{\xi}_T^*, \hat{\lambda}_l^0, \lambda_{l+1}^0; \mathbf{y} \right) &\leq \left(\lambda_{l+1}^o - \lambda_l^0 \right) L_{l+1} \left(\boldsymbol{\xi}_{l+1}^0 \right), \\ \lim_{T\to\infty} \frac{1}{T} L_T^{(k)} \left(\hat{\boldsymbol{\xi}}_T^{(k)}, \hat{\lambda}_{i+m}^0, \hat{\lambda}_k; \mathbf{y} \right) &\leq \left(\lambda_k^* - \lambda_{i+m}^0 \right) L_{i+m+1} \left(\boldsymbol{\xi}_{i+m+1}^0 \right). \end{aligned}$$

Note that the strict inequalities hold for at least one of the above equations since $\boldsymbol{\xi}_k^*$ cannot be correctly specified for all the different segments. Thus we have

(4.8.39)

$$\begin{aligned} &\lim_{T\to\infty} \frac{1}{T} L_T^{(k)} \left(\hat{\boldsymbol{\xi}}_T^{(k)}, \hat{\lambda}_{k-1}, \hat{\lambda}_k; \mathbf{y} \right) \\ &< \left(\lambda_i^o - \lambda_{k-1}^* \right) L_i \left(\boldsymbol{\xi}_i^0 \right) + \sum_{l=i}^{i+m-1} \left(\lambda_{l+1}^o - \lambda_l^0 \right) L_l \left(\boldsymbol{\xi}_l^0 \right) + \left(\lambda_k^* - \lambda_{i+m}^0 \right) L_{i+m+1} \left(\boldsymbol{\xi}_{i+m+1}^0 \right). \end{aligned}$$

Now, as the number of estimated segments is the same as the true number of segments and $\lambda^* \neq \lambda^0$, there is at least one segment in which case 2 applies. Thus, for T large enough, we have

$$\begin{aligned}&\frac{2}{T}MDL(K_0, \boldsymbol{\lambda_T}, \boldsymbol{\xi_T})\\ &> \frac{c_T}{T} - \sum_{i=1}^{K+1}(\lambda_i^0 - \lambda_{i-1}^0)L_i\left(\boldsymbol{\xi}_i^0\right) = \frac{2}{T}MDL(K_0, \boldsymbol{\lambda}^0, \boldsymbol{\xi}_i^0) \geq \frac{2}{T}MDL(K_0, \boldsymbol{\lambda_T}, \boldsymbol{\xi_T}),\end{aligned}$$

which is a contradiction. Hence $\hat{\boldsymbol{\lambda}}_n \neq \boldsymbol{\lambda}^0$ for all $\omega \in B$. Thus, the lemma is proved. □

LEMMA 4.8.0.8. *Under the conditions of Lemma 4.8.0.7, if the number of change points is unknown and is estimated from the data using (4.16), then*

A. The number of change points cannot be underestimated, which means that $\hat{K} \leq K_0$ for T is large enough almost surely.
B. When $\hat{K} > K_0$, $\boldsymbol{\lambda}^0$ must be a subset of the limit points of $\hat{\boldsymbol{\lambda}}_T$, which means for any given $\omega \in B$, $\omega > 0$ and $\lambda_k^0 \in \boldsymbol{\lambda}^0$, there exists a $\hat{\lambda}_i \in \hat{\boldsymbol{\lambda}}_T$ such that $|\lambda_k^0 - \hat{\lambda}_i| < \epsilon$ for sufficiently large T.

PROOF. Notice that in the proof of Lemma 4.8.0.7, the assumption of known K_0 is only used to guarantee that case 2 is applied at least once. No matter how many segments $\boldsymbol{\lambda}^*$ contains, contradiction (4.8.39) arises whenever case 2 applies. So this lemma is proved. □

LEMMA 4.8.0.9. *Denote $\boldsymbol{\lambda}^0 = (\lambda_1^0, \lambda_2^0, \cdots, \lambda_{K_0}^0)$ as the true change points. Then with $(\hat{K}, \hat{\boldsymbol{\lambda}}_{\boldsymbol{T}}, \hat{\boldsymbol{\xi}}_{\boldsymbol{T}})$ defined in (4.16), for each $k = 1, 2, \cdots, K_0$, there exists a $\hat{\lambda}_{i_k} \in \hat{\boldsymbol{\lambda}}_T, 1 \leq i_k \leq \hat{K}$ such that for any $\delta > 0$*

$$\left|\lambda_k^0 - \hat{\lambda}_{i_k}\right| = O_p(T^{\delta-1}).$$

PROOF. From Lemma 4.8.0.8 we can assume that $\hat{K} \geq K_0$ and for each λ_k^0 there exists a $\hat{\lambda}_{i_k}$ such that $\left|\lambda_k^0 - \hat{\lambda}_{i_k}\right| = o(1)$ a.s., where $1 < i_1 < i_2 < \cdots < i_m < \hat{K}$. By construction, for every $m = 0, \cdots, \hat{K} - 1$, we have $\left|\hat{\lambda}_{K+1} - \hat{\lambda}_K\right| > \lambda_\epsilon$, so $\hat{\lambda}_{i_k}$ is the estimated location of change-point closets to λ_k^0 for sufficiently large T. We only need to prove that, for any $\delta > 0$, there exists a $c > 0$ such that

$$P\left(\exists l, \left|\lambda_l^o - \hat{\lambda}_{i_l}\right| > cT^{\delta-1}\right) \to 0.$$

Define $\tilde{\boldsymbol{\lambda}}_T = \{\hat{\lambda}_1, \hat{\lambda}_2, \cdots, \lambda_l^0, \hat{\lambda}_{i_l+1}, \cdots, \hat{\lambda}_{K_0}\}$, where $|\lambda_l^0 - \hat{\lambda}_{i_l}| > cT^{\delta-1}$. By the definition of $(\hat{K}, \hat{\boldsymbol{\lambda}}_{\boldsymbol{T}}, \hat{\boldsymbol{\xi}}_{\boldsymbol{T}})$, it is suffice to show that

$$P(MDL((\hat{K}, \hat{\boldsymbol{\lambda}}_{\boldsymbol{T}}, \hat{\boldsymbol{\xi}}_{\boldsymbol{T}})) < MDL((\hat{K}, \tilde{\boldsymbol{\lambda}}_{\boldsymbol{T}}, \hat{\boldsymbol{\xi}}_{\boldsymbol{T}})), \exists l, |\lambda_l^0 - \hat{\lambda}_{i_l}| > cT^{\delta-1}) \to 0.$$

As the number of change points is bounded, it is sufficient to show that, for each fixed l, we have

$$P(MDL((\hat{K}, \hat{\boldsymbol{\lambda}}_{\boldsymbol{T}}, \hat{\boldsymbol{\xi}}_{\boldsymbol{T}})) < MDL((\hat{K}, \tilde{\boldsymbol{\lambda}}_{\boldsymbol{T}}, \hat{\boldsymbol{\xi}}_{\boldsymbol{T}})), |\lambda_l^0 - \hat{\lambda}_{i_l}| > cT^{\delta-1}) \to 0.$$

Given that $|\lambda_l^0 - \hat{\lambda}_{i_l}| > cT^{\delta-1}$, the difference $MDL((\hat{K}, \hat{\boldsymbol{\lambda}}_{\boldsymbol{T}}, \hat{\boldsymbol{\xi}}_{\boldsymbol{T}})) - MDL((\hat{K}, \tilde{\boldsymbol{\lambda}}_{\boldsymbol{T}}, \hat{\boldsymbol{\xi}}_{\boldsymbol{T}}))$ is either

$$\sum_{k=T_l-[T(\lambda_l^0-\hat{\lambda}_{i_l})]+1}^{T_l}\left(l_{i_l}(\hat{\boldsymbol{\xi}}_{i_l}; x_{k,l}, \boldsymbol{z}_t^*|\boldsymbol{y_{k,l}}) - l_{i_l+1}(\hat{\boldsymbol{\xi}}_{i_l+1}; x_{k,l}, \boldsymbol{z}_t^*|\boldsymbol{y_{k,l}})\right)$$

or

$$\sum_{k=1}^{[T(\hat{\lambda}_{i_l}-\lambda_l^0)]} \left(l_{i_l+1}(\hat{\boldsymbol{\xi}}_{i_l+1}; x_{k,l}, \boldsymbol{z}_t^* | \boldsymbol{y_{k,l}}) - l_{i_l}(\hat{\boldsymbol{\xi}}_{i_l}; x_{k,l}, \boldsymbol{z}_t^* | \boldsymbol{y_{k,l}}) \right).$$

And from Lemma 4.8.0.1, we have the difference is either

$$\Sigma' \left(l_{i_l}(\hat{\boldsymbol{\xi}}_{i_l}; x_{k,l}, \boldsymbol{z}_t^* | \boldsymbol{y_{k,l}}) - l_{i_l+1}(\hat{\boldsymbol{\xi}}_{i_l+1}; x_{k,l}, \boldsymbol{z}_t^* | \boldsymbol{y_{k,l}}) \right) + O_p(1) \tag{4.8.40}$$

or

$$\Sigma'' \left(l_{i_l+1}(\hat{\boldsymbol{\xi}}_{i_l+1}; x_{k,l}, \boldsymbol{z}_t^* | \boldsymbol{y_{k,l}}) - l_{i_l}(\hat{\boldsymbol{\xi}}_{i_l}; x_{k,l}, \boldsymbol{z}_t^* | \boldsymbol{y_{k,l}}) \right) + O_p(1), \tag{4.8.41}$$

where $\Sigma' = \sum_{k=T_l-[T(\lambda_l^0-\hat{\lambda}_{i_l})]+1}^{T_l}$ and $\Sigma'' = \sum_{k=1}^{[T(\hat{\lambda}_{i_l}-\lambda_l^0)]}$. By Lemma 4.8.0.6 and the ergodic theorem, we have for case (4.8.41) is positive and of order no less than $O(T^\delta)$ a.s. For (4.8.40), the ergodic theorem as well as the stationarity in each segment together guarantees the positive of this term and is of order $O(T^\delta)$. Therefore, both the quantities (4.8.40) and (4.8.41) are positive with probability going to 1, and hence, the lemma is proved. □

LEMMA 4.8.0.10. *If $\{X_t\}$ is a sequence of stationary, zero-mean strongly mixing process with geometric rate, and $E(|X_1|^{r+\epsilon}) < \infty$ for some $2 \leq r < \infty$ and $\epsilon > 0$, then*

$$\frac{1}{g(T)} \sum_{t=1}^{g(T)} X_t \xrightarrow{a.s} \mu \quad \textit{and} \quad \frac{1}{g(T)} \sum_{t=n-g(T)+1}^{\top} X_t \xrightarrow{a.s} \mu$$

for any sequence $g(T)_{T\geq 1}$ for integers that satisfies $g(T) > cT^{2/r}$ for some $c > 0$ when T is sufficiently large. Moreover,

$$\sum_{t=1}^{s(T)} X_t = O\left(T^{2/r}\right) \quad \textit{and} \quad \sum_{t=T-s(T)+1}^{\top} X_t = O\left(T^{2/r}\right)$$

a.s., for any sequence $\{s(T)\}_{T\geq 1}$ satisfying $s(T) = O(T^{2/r})$.

PROOF. This lemma is taken from [81], from which the proof is given. □

LEMMA 4.8.0.11. *Recall $\hat{\boldsymbol{\xi}}_T^{(k)}(\lambda_l, \lambda_u) = \arg\max_{\boldsymbol{\xi}_k \in \Theta_n} \tilde{L}_T^{(k)}(\boldsymbol{\xi}_k, \lambda_l, \lambda_u; \mathbf{x}_k)$ .. We have*

$$\hat{\boldsymbol{\xi}}_T^{(k)}(\hat{\lambda}_{k-1}, \hat{\lambda}_k) - \boldsymbol{\xi}_k^0 = O\left(T^{-\frac{1}{2}}\right) \quad \textit{a.s.},$$

where $\boldsymbol{\xi}_k^0$ is the true parameter vector in k-th segment.

PROOF. Denote $(\hat{\lambda}_{k-1}, \hat{\lambda}_k)$ by (λ_l, λ_u) for simplicity. Let $\tilde{L}_T^{'(k)}(\boldsymbol{\xi}_k, \lambda_l, \lambda_u; \mathbf{x}_k)$ and $\tilde{L}_T^{''(k)}(\boldsymbol{\xi}_k, \lambda_l, \lambda_u; \mathbf{x}_k)$ be the first and second partial derivatives of $\tilde{L}_T^{(k)}(\boldsymbol{\xi}_k, \lambda_l, \lambda_u; \mathbf{x}_k)$, respectively. Apply Taylor expansion to $\tilde{L}_T^{'(k)}(\boldsymbol{\xi}_k, \lambda_l, \lambda_u; \mathbf{x}_k)$ around the true parameter vector value $\boldsymbol{\xi}_k^0$, we have

$$\tilde{L}_T^{'(k)}\left(\hat{\boldsymbol{\xi}}_k, \lambda_l, \lambda_u; \mathbf{x}_k\right) = \tilde{L}_T^{'(k)}\left(\boldsymbol{\xi}_k^0, \lambda_l, \lambda_u; \mathbf{x}_k\right) + \tilde{L}_T^{''(k)}\left(\boldsymbol{\xi}_k^+, \lambda_l, \lambda_u; \mathbf{x}_k\right)(\hat{\boldsymbol{\xi}}_k - \boldsymbol{\xi}_k^0), \tag{4.8.42}$$

where $\left|\boldsymbol{\xi}_k^+ - \boldsymbol{\xi}_k^0\right| < \left|\hat{\boldsymbol{\xi}}_k - \boldsymbol{\xi}_k^0\right|$. By definition of $\hat{\boldsymbol{\xi}}_k$, we have $L_T^{'(k)}\left(\hat{\boldsymbol{\xi}}_k, \lambda_l, \lambda_u; \mathbf{x}_k\right) = 0$. Therefore, (4.8.42) is equivalent to

$$\tilde{L}_T^{''(k)}\left(\boldsymbol{\xi}_k^+, \lambda_l, \lambda_u; \mathbf{x}_k\right)\left(\hat{\boldsymbol{\xi}}_k - \boldsymbol{\xi}_k^0\right) = -\tilde{L}_T^{'(k)}\left(\boldsymbol{\xi}_k^0, \lambda_l, \lambda_u; \mathbf{x}_k\right). \tag{4.8.43}$$

So combining Lemma 4.8.0.1, Lemma 4.8.0.9 and Lemma 4.8.0.10, we have

$$\begin{aligned}
\tilde{L}_T^{'(k)}\left(\boldsymbol{\xi}_k^0, \lambda_l, \lambda_u; \mathbf{x}_k\right) &= L_T^{'(k)}\left(\boldsymbol{\xi}_k^0, \lambda_l, \lambda_u; \mathbf{x}_k\right) + O(T) \\
&= \sum_{i=[T\dot{\lambda}_l]+1}^{[T\dot{\lambda}_u]} l_k'\left(\boldsymbol{\xi}_k^0; x_{i,k} | x_{l,k}, l < i\right) + O_p(T^\delta) \\
&= \sum_{i=1}^{[T\dot{\lambda}_u]} l_k'\left(\boldsymbol{\xi}_k^0; x_{i,k} | x_{l,k}, l < i\right) - \sum_{i=1}^{[T\dot{\lambda}_d]} l_k'\left(\boldsymbol{\xi}_k^0; x_{i,k} | x_{l,k}, l < i\right) + O_p(T^\delta),
\end{aligned}$$

where $\dot{\lambda}_l = \max(0, \lambda_l)$ and $\dot{\lambda}_u = \min(1, \lambda_u)$. Since $E\left(l_k'\left(\boldsymbol{\xi}_k^0; x_{i,k} | x_{l,k}, l < i\right)\right) = 0$, so the sequence $\{l_k'\left(\boldsymbol{\xi}_k^0; x_{i,k} | x_{l,k}, l < i\right)_{i\in[N]}\}$ is a stationary ergodic zero-mean martingale difference sequence with finite second moment. Thus, from [108]

$$\sum_{i=1}^{[T\dot{\lambda}_u]} l_k'\left(\boldsymbol{\xi}_k^0; x_{i,k} | x_{l,k}, l < i\right) \quad \text{and} \quad \sum_{i=1}^{[T\dot{\lambda}_d]} l_k'\left(\boldsymbol{\xi}_k^0; x_{i,k} | x_{l,k}, l < i\right)$$

are of order $O_p(T^{\frac{1}{2}})$. Thus, we have $\tilde{L}_T^{'(k)}\left(\boldsymbol{\xi}_k^0, \lambda_l, \lambda_u; \mathbf{x}_k\right) = O_p(1)$ and $L_k''(\boldsymbol{\xi}_k^0)$ is positive definite. Together with Lemma 4.8.0.5 and $\left|\boldsymbol{\xi}_T^+ - \boldsymbol{\xi}_T^0\right| \to O_p(1)$, $\frac{1}{T}\tilde{L}_T^{''(k)}\left(\boldsymbol{\xi}_k^+, \lambda_l, \lambda_u; \mathbf{x}_k\right)\left(\hat{\boldsymbol{\xi}}_k - \boldsymbol{\xi}_k^0\right)$ is positive definite. Combining all the above, the lemma is proved. □

4.8.1.2. *Proof of Theorem 4.3.1.* We are now ready to prove Theorem 4.3.1.

PROOF. From Lemma 4.8.0.11, we have $\left|\hat{\boldsymbol{\xi}}_T^{(k)}(\hat{\lambda}_{k-1}, \hat{\lambda}_k) - \boldsymbol{\xi}_k^0\right| = O\left(T^{-\frac{1}{2}}\right)$ a.s. Following Lemma 4.8.0.8 and Lemma 4.8.0.9, it is suffice to prove that for any integer $d = 1, \cdots, M - K_0$, any $\delta > 0$ and any sequence $\tilde{\boldsymbol{\lambda}}_{\boldsymbol{T}} = (\tilde{\lambda}_1, \cdots, \tilde{\lambda}_{K_0})$ such that $|\lambda_k^0 - \tilde{\lambda}_k| = O(T^{\delta-1})$ for $k = 1, \cdots, K_0$,

$$\underset{\substack{\boldsymbol{\xi}, \boldsymbol{\lambda} \in A_{\epsilon_\lambda}^{(K_0+d)} \\ \tilde{\boldsymbol{\lambda}}_{\boldsymbol{T}} \subset \boldsymbol{\lambda}}}{\arg\min} \left[\frac{2}{T} MDL(K_0 + d, \boldsymbol{\lambda}, \boldsymbol{\xi})\right] - \frac{2}{T} MDL(K_0, \tilde{\boldsymbol{\lambda}}_{\boldsymbol{T}}, \boldsymbol{\xi}^0) \tag{4.8.44}$$

is positive with a probability approaching 1. Denote

$$\hat{\boldsymbol{\lambda}}_T = (\hat{\lambda}_1, \cdots, \hat{\lambda}_{K_0+d+1}) = \underset{\substack{\boldsymbol{\xi}, \boldsymbol{\lambda} \in A_{\epsilon_\lambda}^{(K_0+d)} \\ \tilde{\boldsymbol{\lambda}} \subset \boldsymbol{\lambda}}}{\arg\min} \left[\frac{2}{T} MDL(K_0 + d, \boldsymbol{\lambda}, \boldsymbol{\xi})\right].$$

First note that $\tilde{\boldsymbol{\lambda}}_T \subset \hat{\boldsymbol{\lambda}}_T$ by construction. Using Taylor expansion on the likelihood function, (4.8.44) can be rewritten as

$$
\begin{aligned}
&(4.8.44) = C_1 - C_2 \\
&+ \frac{1}{T}\left(\sum_{k=1}^{K_0+1} \tilde{L}_T^{(k)}\left(\boldsymbol{\xi}_k^0, \tilde{\lambda}_{k-1}, \tilde{\lambda}_k; \mathbf{y}\right) - \sum_{l=1}^{K_0+d+1} \tilde{L}_T^{(l)}\left(\hat{\boldsymbol{\xi}}_T^{(l-1)}, \hat{\lambda}_{l-1}, \hat{\lambda}_l; \mathbf{y}\right)\right) \\
&- \sum_{l=1}^{K_0+d+1} \left(\hat{\boldsymbol{\xi}}_T^{(l)} - \boldsymbol{\xi}_l^*\right)^\top \frac{1}{T} \tilde{L}_T^{\prime\prime(k)}\left(\boldsymbol{\xi}_k^+, \hat{\lambda}_{l-1}, \hat{\lambda}_l; \mathbf{y}\right)\left(\hat{\boldsymbol{\xi}}_T^{(l)} - \boldsymbol{\xi}_l^*\right),
\end{aligned} \tag{4.8.45}
$$

where $C_1 - C_2$ is positive and of order $O(\frac{\log(T)}{T})$ and $\left|\boldsymbol{\xi}_k^+ - \boldsymbol{\xi}_k^0\right| < \left|\hat{\boldsymbol{\xi}}_k - \boldsymbol{\xi}_k^0\right|$.

The third part of (4.8.45) is 0 and since $\left|\hat{\boldsymbol{\xi}}_T^{(l)}(\hat{\lambda}_{k-1}, \hat{\lambda}_k) - \boldsymbol{\xi}_l^0\right| = O\left(T^{-\frac{1}{2}}\right)$. As the fourth part is of order $O_p(T^{-1})$, $C_1 - C_2$ is the dominant part in (4.8.45). Thus (4.8.44) is positive with probability approaching 1. So, the theorem is proved. □

Bibliography

[1] Andreas S Spanias. Speech coding: A tutorial review. *Proceedings of the IEEE*, 82(10):1541–1582, 1994.

[2] Hamid Krim and Mats Viberg. Two decades of array signal processing research: the parametric approach. *IEEE Signal Processing Magazine*, 13(4):67–94, 1996.

[3] Rob Carriere and Randolph L Moses. High resolution radar target modeling using a modified prony estimator. *IEEE Transactions on Antennas and Propagation*, 40(1):13–18, 1992.

[4] Liliana Borcea, George Papanicolaou, Chrysoula Tsogka, and James Berryman. Imaging and time reversal in random media. *Inverse Problems*, 18(5):1247, 2002.

[5] Petre Stoica and Prabhu Babu. Sparse estimation of spectral lines: Grid selection problems and their solutions. *IEEE Transactions on Signal Processing*, 60:962–967, 2011.

[6] Maurice S Bartlett. On the theoretical specification and sampling properties of autocorrelated time-series. *Supplement to the Journal of the Royal Statistical Society*, 8:27–41, 1946.

[7] Peter Welch. The use of fast fourier transform for the estimation of power spectra: a method based on time averaging over short, modified periodograms. *IEEE Transactions on Audio and Electroacoustics*, 15:70–73, 1967.

[8] John Makhoul. Linear prediction: A tutorial review. *Proceedings of the IEEE*, 63:561–580, 1975.

[9] J Cadzow. High performance spectral estimation–a new ARMA method. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 28:524–529, 1980.

[10] Md Mashud Hyder and Kaushik Mahata. Direction-of-arrival estimation using a mixed $l_{2,0}$ norm approximation. *IEEE Transactions on Signal processing*, 58(9):4646–4655, 2010.

[11] Badri Narayan Bhaskar, Gongguo Tang, and Benjamin Recht. Atomic norm denoising with applications to line spectral estimation. *IEEE Transactions on Signal Processing*, 61(23):5987–5999, 2013.

[12] Arian Maleki and David L Donoho. Optimally tuned iterative reconstruction algorithms for compressed sensing. *IEEE Journal of Selected Topics in Signal Processing*, 4(2):330–341, 2010.

[13] Zai Yang, Lihua Xie, and Cishen Zhang. Off-grid direction of arrival estimation using sparse Bayesian inference. *IEEE Transactions on Signal Processing*, 61(1):38–43, 2012.

[14] Jiang Zhu, Qi Zhang, Peter Gerstoft, Mihai-Alin Badiu, and Zhiwei Xu. Grid-less variational Bayesian line spectral estimation with multiple measurement vectors. *Signal Processing*, 161:155–164, 2019.

[15] Joel A Tropp. Greed is good: Algorithmic results for sparse approximation. *IEEE Transactions on Information Theory*, 50(10):2231–2242, 2004.

[16] Emmanuel J Candès, Justin Romberg, and Terence Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Transactions on Information Theory*, 52(2):489–509, 2006.

[17] Jan Hannig, Hari Iyer, Randy CS Lai, and Thomas C. M. Lee. Generalized fiducial inference: A review and new results. *Journal of the American Statistical Association*, 111:1346–1361, 2016.

[18] Ronald A Fisher. Inverse probability. In *Mathematical Proceedings of the Cambridge Philosophical Society*, volume 26, pages 528–535. Cambridge University Press, 1930.

[19] Arthur P Dempster. The Dempster–Shafer calculus for statisticians. *International Journal of approximate reasoning*, 48(2):365–377, 2008.

[20] Paul T Edlefsen, Chuanhai Liu, Arthur P Dempster, et al. Estimating limits from poisson counting data using dempster–shafer analysis. *Annals of Applied Statistics*, 3(2):764–790, 2009.

[21] Jianchun Zhang and Chuanhai Liu. Dempster-shafer inference with weak beliefs. *Statistica Sinica*, pages 475–494, 2011.

[22] Kam-Wah Tsui and Samaradasa Weerahandi. Generalized p-values in significance testing of hypotheses in the presence of nuisance parameters. *Journal of the American Statistical Association*, 84(406):602–607, 1989.

[23] Samaradasa Weerahandi. Generalized confidence intervals. In *Exact Statistical Methods for Data Analysis*, pages 143–168. Springer, 1995.

[24] Andy K L Chiang. A simple general method for constructing confidence intervals for functions of variance components. *Technometrics*, 43(3):356–367, 2001.

[25] Jan Hannig and Thomas C. M. Lee. Generalized fiducial inference for wavelet regression. *Biometrika*, 96:847–860, 2009.

[26] Randy CS Lai, Jan Hannig, and Thomas C. M. Lee. Generalized fiducial inference for ultrahigh-dimensional regression. *Journal of the American Statistical Association*, 110:760–772, 2015.

[27] Qi Gao, Randy CS Lai, Thomas C. M. Lee, and Yao Li. Uncertainty quantification for high-dimensional sparse nonparametric additive models. *Technometrics*, 62(4):513–524, 2020.
[28] Jan Hannig. On generalized fiducial inference. *Statistica Sinica*, pages 491–544, 2009.
[29] Jorma Rissanen. *Information and Complexity in Statistical Modeling*. Springer Science & Business Media, 2007.
[30] Zhenyu Wei, Raymond K. W. Wong, and Thomas C. M. Lee. Extending the use of MDL for high-dimensional problems: Variable selection, robust fitting, and additive modeling. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5707–5711, 2022.
[31] Petre Stoica and Prabhu Babu. SPICE and LIKES: Two hyperparameter-free methods for sparse-parameter estimation. *Signal Processing*, 92(7):1580–1590, 2012.
[32] Prabhu Babu, Peter Stoica, Jian Li, Zhaofu Chen, and Jian Ge. Analysis of radial velocity data by a novel adaptive approach. *The Astronomical Journal*, 139:783, 2010.
[33] Minjie Fan, Jue Wang, Vinay L Kashyap, Thomas CM Lee, David A van Dyk, and Andreas Zezas. Identifying diffuse spatial structures in high-energy photon lists. *The Astronomical Journal*, 165(2):66, 2023.
[34] C Moutou, M Mayor, F Bouchy, C Lovis, F Pepe, D Queloz, NC Santos, S Udry, W Benz, G Lo Curto, et al. The harps search for southern extra-solar planets-iv. three close-in planets around HD 2638, HD 27894 and HD 63454. *Astronomy & Astrophysics*, 439:367–373, 2005.
[35] PC Gregory. A Bayesian kepler periodogram detects a second planet in HD 208487. *Monthly Notices of the Royal Astronomical Society*, 374:1321–1333, 2007.
[36] Eugenio J Rivera, Jack J Lissauer, R Paul Butler, Geoffrey W Marcy, Steven S Vogt, Debra A Fischer, Timothy M Brown, Gregory Laughlin, and Gregory W Henry. A $\sim$7.5 $M_\oplus$ planet orbiting the nearby star, GJ 876. *The Astrophysical Journal*, 634:625, 2005.
[37] Zai Yang, Jian Li, Petre Stoica, and Lihua Xie. Sparse methods for direction-of-arrival estimation. In *Academic Press Library in Signal Processing, Volume 7*, pages 509–581. Elsevier, 2018.
[38] Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58:267–288, 1996.
[39] Xuehu Zhu, Yue Kang, and Junmin Liu. Estimation of the number of endmembers via thresholding ridge ratio criterion. *IEEE Transactions on Geoscience and Remote Sensing*, 58(1):637–649, 2019.
[40] Haixu Ma, Donglin Zeng, and Yufeng Liu. Learning individualized treatment rules with many treatments: A supervised clustering approach using adaptive fusion. *Advances in Neural Information Processing Systems*, 35:15956–15969, 2022.
[41] Arian Maleki, Laura Anitori, Zai Yang, and Richard G Baraniuk. Asymptotic analysis of complex lasso via complex approximate message passing (CAMP). *IEEE Transactions on Information Theory*, 59:4290–4308, 2013.
[42] Ming Yuan and Yi Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68:49–67, 2006.
[43] Han Liu and Jian Zhang. Estimation consistency of the group lasso and its applications. In *Artificial Intelligence and Statistics*, pages 376–383. PMLR, 2009.
[44] Nicolai Meinshausen and Peter Bühlmann. Stability selection. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 72:417–473, 2010.
[45] Petre Stoica, Prabhu Babu, and Jian Li. New method of sparse parameter estimation in separable models and its use for spectral analysis of irregularly sampled data. *IEEE Transactions on Signal Processing*, 59:35–47, 2010.
[46] Yiyuan She, Jiangping Wang, Huanghuang Li, and Dapeng Wu. Group iterative spectrum thresholding for super-resolution sparse spectral selection. *IEEE Transactions on Signal Processing*, 61:6371–6386, 2013.
[47] Haixu Ma, Donglin Zeng, and Yufeng Liu. Learning optimal group-structured individualized treatment rules with many treatments. *Journal of Machine Learning Research*, 24(102):1–48, 2023.
[48] John Y Campbell, George Chacko, Jorge Rodriguez, and Luis M Viceira. Strategic asset allocation in a continuous-time VAR model. *Journal of Economic Dynamics and Control*, 28(11):2195–2214, 2004.
[49] Alessandro Cologni and Matteo Manera. Oil prices, inflation and interest rates in a structural cointegrated VAR model for the g-7 countries. *Energy economics*, 30(3):856–888, 2008.
[50] Xuening Zhu, Rui Pan, Guodong Li, Yuewen Liu, and Hansheng Wang. Network vector autoregression. *The Annals of Statistics*, 45(3):1096–1123, 2017.
[51] Dimitrios Katselis, Carolyn L. Beck, and R. Srikant. Mixing times and structural inference for Bernoulli autoregressive processes. *IEEE Transactions on Network Science and Engineering*, 6(3):364–378, 2019.
[52] Kai Yang, Shaoyu Dou, Pan Luo, Xin Wang, and H. Vincent Poor. Robust group anomaly detection for quasi-periodic network time series. *IEEE Transactions on Network Science and Engineering*, 9(4):2833–2845, 2022.
[53] Hang Yin, Abolfazl Safikhani, and George Michailidis. A general modeling framework for network autoregressive processes. *arXiv preprint arXiv:2110.09596*, 2021.
[54] Marina Knight, Kathryn Leeming, Guy Nason, and Matthew Nunes. Generalised network autoregressive processes and the gnar package. *arXiv preprint arXiv:1912.04758*, 2019.
[55] Hernando Ombao, Rainer Von Sachs, and Wensheng Guo. SLEX analysis of multivariate nonstationary time series. *Journal of the American Statistical Association*, 100(470):519–531, 2005.

[56] Giorgio E Primiceri. Time varying structural vector autoregressions and monetary policy. *The Review of Economic Studies*, 72(3):821–852, 2005.
[57] Jorma Rissanen. *Stochastic Complexity in Statistical Inquiry*. World Scientific, Singapore, 1989.
[58] Haixu Ma, Yufeng Liu, and Guorong Wu. Elucidating multi-stage progression of neuro-degeneration process in alzheimer's disease. *Alzheimer's & Dementia*, 18:e068774, 2022.
[59] Li Chen, Jie Zhou, and Lizhen Lin. Hypothesis testing for populations of networks. *Communications in Statistics-Theory and Methods*, pages 1–24, 2021.
[60] Sayantan Banerjee and Kousik Guhathakurta. Change-point analysis in financial networks. *Stat*, 9(1):e269, 2020.
[61] Jonathan Flossdorf and Carsten Jentsch. Change detection in dynamic networks using network characteristics. *IEEE Transactions on Signal and Information Processing over Networks*, 7:451–464, 2021.
[62] Tingting Zhu, Ping Li, Lanlan Yu, Kaiqi Chen, and Yan Chen. Change point detection in dynamic networks based on community identification. *IEEE Transactions on Network Science and Engineering*, 7(3):2067–2077, 2020.
[63] Leto Peel and Aaron Clauset. Detecting change points in the large-scale structure of evolving networks. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2015.
[64] Sebastian Moreno and Jennifer Neville. Network hypothesis testing using mixed kronecker product graph models. In *2013 IEEE 13th International Conference on Data Mining*, pages 1163–1168. IEEE, 2013.
[65] Yuriy Hulovatyy and Tijana Milenković. SCOUT: simultaneous time segmentation and community detection in dynamic networks. *Scientific reports*, 6(1):1–11, 2016.
[66] Thomas C. M. Lee. Segmenting images corrupted by correlated noise. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(5):481–492, 1998.
[67] Remco R Bouckaert. Probabilistic network construction using the minimum description length principle. In *European conference on symbolic and quantitative approaches to reasoning and uncertainty*, pages 41–48. Springer, 1993.
[68] Rex CY Cheung, Alexander Aue, Seungyong Hwang, and Thomas C. M. Lee. Simultaneous detection of multiple change points and community structures in time series of networks. *IEEE Transactions on Signal and Information Processing over Networks*, 6:580–591, 2020.
[69] Cong Xu and Thomas C. M. Lee. Statistical consistency for change point detection and community estimation in time-evolving dynamic networks. *IEEE Transactions on Signal and Information Processing over Networks*, 8:215–227, 2022.
[70] Alexander Aue, Rex C. Y. Cheung, Thomas C. M. Lee, and Ming Zhong. Segmented model selection in quantile regression using the minimum description length principle. *Journal of the American Statistical Association*, 109:1241–1256, 2014.
[71] Thomas Chun Man Lee. Regression spline smoothing using the minimum description length principle. *Statistics and Probability Letters*, 48:71–82, 2000.
[72] Thomas C. M. Lee. An introduction to coding theory and the two–part minimum description length principle. *International Statistical Review*, 69:169–183, 2001.
[73] Richard A Davis, Thomas C. M. Lee, and Gabriel A Rodriguez-Yam. Structural break estimation for nonstationary time series models. *Journal of the American Statistical Association*, 101(473):223–239, 2006.
[74] Alex S Fraser. Simulation of genetic systems by automatic digital computers i. introduction. *Australian Journal of Biological Sciences*, 10(4):484–491, 1957.
[75] V Scott Gordan and Darrel Whitley. Serial and parallel genetic algorithms as function optimizers. In *Proc. 5th Int. Conf. on Genetic Algorithms, Morgan Kaufmann*, 1993.
[76] Abolfazl Safikhani and Ali Shojaie. Joint structural break detection and parameter estimation in high-dimensional nonstationary VAR models. *Journal of the American Statistical Association*, 117(537):251–264, 2022.
[77] Zhengping Che, Sanjay Purushotham, Kyunghyun Cho, David Sontag, and Yan Liu. Recurrent neural networks for multivariate time series with missing values. *Scientific Reports*, 8(1):6085, 2018.
[78] M Sangeetha and M Senthil Kumaran. Deep learning-based data imputation on time-variant data using recurrent neural network. *Soft Computing*, 24:13369–13380, 2020.
[79] Jinsung Yoon, James Jordon, and Mihaela Schaar. Gain: Missing data imputation using generative adversarial nets. In *International Conference on Machine Learning*, pages 5689–5698. PMLR, 2018.
[80] Yi Han and Thomas C. M. Lee. Uncertainty quantification for sparse estimation of spectral lines. *IEEE Transactions on Signal Processing*, 70:6243–6256, 2022.
[81] Richard A Davis and Chun Yip Yau. Consistency of minimum description length model selection for piecewise stationary time series models. *Electronic Journal of Statistics*, 7:381–411, 2013.
[82] W.F. Stout. *Almost Sure Convergence*. Probability and mathematical statistics. Academic Press, 1974.
[83] Linas Baltrunas, Tadas Makcinskas, and Francesco Ricci. Group recommendations with rank aggregation and collaborative filtering. In *Proceedings of the fourth ACM conference on Recommender Systems*, pages 119–126, 2010.
[84] Carolin Strobl, Florian Wickelmaier, and Achim Zeileis. Accounting for individual differences in Bradley-Terry models by means of recursive partitioning. *Journal of Educational and Behavioral Statistics*, 36:135–153, 2011.

[85] Ian McHale and Alex Morton. A Bradley-Terry type model for forecasting tennis match results. *International Journal of Forecasting*, 27:619–630, 2011.
[86] Manuela Cattelan, Cristiano Varin, and David Firth. Dynamic Bradley–Terry modelling of sports tournaments. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 62:135–150, 2013.
[87] Robin L Plackett. The analysis of permutations. *Journal of the Royal Statistical Society Series C: Applied Statistics*, 24:193–202, 1975.
[88] Cynthia Dwork, Ravi Kumar, Moni Naor, and Dandapani Sivakumar. Rank aggregation methods for the web. In *Proceedings of the 10th International Conference on World Wide Web*, pages 613–622, 2001.
[89] Tzu-Kuo Huang, Chih-Jen Lin, and Ruby C Weng. Ranking individuals by group comparisons. In *Proceedings of the 23rd International Conference on Machine Learning*, pages 425–432, 2006.
[90] Ruby C Weng and Chih-Jen Lin. A Bayesian approximation method for online ranking. *Journal of Machine Learning Research*, 12, 2011.
[91] Nihar B Shah and Martin J Wainwright. Simple, robust and optimal ranking from pairwise comparisons. *Journal of Machine Learning Research*, 18:1–38, 2018.
[92] Michael Pearce and Elena A Erosheva. A unified statistical learning model for rankings and scores with application to grant panel review. *Journal of Machine Learning Research*, 23:1–33, 2022.
[93] M. E. J. Newman. Efficient computation of rankings from pairwise comparisons. *Journal of Machine Learning Research*, 24:1–25, 2023.
[94] Braxton Osting, Christoph Brune, and Stanley J Osher. Optimal data collection for informative rankings expose well-connected graphs. *Journal of Machine Learning Research*, 15:2981–3012, 2014.
[95] Ralph Allan Bradley and Milton E Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39:324–345, 1952.
[96] R Duncan Luce. Individual choice behavior, 1959.
[97] Jianqing Fan, Jikai Hou, and Mengxin Yu. Uncertainty quantification of MLE for entity ranking with covariates. *arXiv preprint arXiv:2212.09961*, 2022.
[98] Mark E Glickman. Parameter estimation in large dynamic paired comparison experiments. *Journal of the Royal Statistical Society Series C: Applied Statistics*, 48:377–394, 1999.
[99] Mark E Glickman. Dynamic paired comparison models with stochastic variances. *Journal of Applied Statistics*, 28:673–689, 2001.
[100] Heejong Bong, Wanshan Li, Shamindra Shrotriya, and Alessandro Rinaldo. Nonparametric estimation in the dynamic Bradley-Terry model. In *International Conference on Artificial Intelligence and Statistics*, pages 3317–3326. PMLR, 2020.
[101] Wanshan Li, Alessandro Rinaldo, and Daren Wang. Detecting abrupt changes in sequential pairwise comparison data. *Advances in Neural Information Processing Systems*, 35:37851–37864, 2022.
[102] Jorma Rissanen. *Stochastic Complexity in Statistical Inquiry*. World Scientific, 1998.
[103] Rebecca Killick, Paul Fearnhead, and Idris A Eckley. Optimal detection of changepoints with a linear computational cost. *Journal of the American Statistical Association*, 107:1590–1598, 2012.
[104] Yi Han and Thomas C. M. Lee. Structural break detection in non-stationary network vector autoregression models. *IEEE Transactions on Network Science and Engineering*, pages 1–14, 2024.
[105] Alexander Aue, Rex CY Cheung, Thomas C. M. Lee, and Ming Zhong. Segmented model selection in quantile regression using the minimum description length principle. *Journal of the American Statistical Association*, 109:1241–1256, 2014.
[106] Thomas C. M. Lee. A minimum description length-based image segmentation procedure, and its comparison with a cross-validation-based segmentation procedure. *Journal of the American Statistical Association*, 95:259–270, 2000.
[107] Thomas C. M. Lee. An introduction to coding theory and the two-part minimum description length principle. *International Statistical Review*, 69:169–183, 2001.
[108] Markus Pauly. Weighted resampling of martingale difference arrays with applications. 2011.