Change Point Detection and Node Clustering for Time Series of Graphs

Cong Xu and Thomas C. M. Lee *Senior Member, IEEE*

***Abstract*—Suppose an undirected graph is observed over time. Its structure (i.e., nodes and edges) remains the same but the measurements taken at the nodes may vary over time. This paper proposes a method that simultaneously performs the following two tasks: (i) it detects change points in the time domain, and (ii) for each time interval between any two consecutive detected change points, it partitions the nodes into different clusters of similar measurements. The method begins with recasting the problem into a model selection problem and employs the minimum description length principle to construct a selection criterion for which the best fitting model is defined as its minimizer. A practical algorithm is developed to (approximately) locate this minimizer. It is shown that the model selection criterion leads to statistically consistent estimates, while numerical experiments show that the method enjoys promising empirical properties. To the best of the authors' knowledge, the proposed method is one of the first that performs simultaneous change point detection and node clustering for time series of graphs.**

***Index Terms*—graph denoising, graph-guided fused lasso, group fused lasso, minimum description length principle, smoothing proximal gradient descent**

I. Introduction

CONSIDER the following situation. Suppose we would like to study criminal activities in a certain region over time. We could first partition the region into different administrative districts, where each district is represented by a node in a graph. Two nodes are connected if their corresponding districts share a common border. For each node weekly measurements are taken over a time period. These measurements can be the weekly total numbers of reported crime incidents in the district, or they can be the numbers of certain crime incidents such as burglary. With this setup, we can model the crime measurements as a time-evolving graph, and see if the crime activities change over time, or if they are spatially correlated in the sense that adjacent districts have similar patterns.

This problem can be formalized as follows. Suppose a time-evolving graph is observed at time $t = 1, \ldots, T$. The number of nodes and the node connectivity (i.e., edges) remain unchanged over time, although the noisy measurements observed at the nodes may change. We use p to denote the number of nodes, $n_{t,i}$ to denote the number of measurements observed in the ith node at time t, and $x_{t,i,j}$ to denote the jth measurement (i.e., $j = 1, \ldots, n_{t,i}$) of the ith node at time t, where $i = 1, \ldots, p$ and $t = 1, \ldots, T$. The values of the $n_{t,i}$'s are typically small, and could even be zero for some t, i; i.e., no measurement. For all $\{t, i, j\}$, we model the measurements $x_{t,i,j}$ as realizations of normal random variables $X_{t,i,j}$ that satisfy

$$X_{t,i,j} \overset{independent}{\sim} \mathcal{N}(\beta_{t,i}, \sigma^2),$$

where $\beta_{t,i}$ is the true signal value for the ith node at time t, and σ^2 is the noise variance. The goal is to, given the data $x_{t,i,j}$, estimate the signal $\beta_{t,i}$. Of course, an unbiased estimator for $\beta_{t,i}$ is $\sum_{j=1}^{n_{t,i}} x_{t,i,j}/n_{t,i}$ (if $n_{t,i} > 0$), the sample average. However, this estimator is of high variance if $n_{t,i}$ is small, which is quite common for many real data problems, where it is typical to have $n_{t,i} = 1$ for some $\{t, i\}$. We consider the setting when both p and T are fixed, while all $n_{t,i}$'s go to infinity at the same linear rate.

To obtain improved estimates for $\beta_{t,i}$, two additional assumptions are imposed. First we assume that the underlying signal is temporally smooth. Specifically, we assume that there exists a sequence of M time points $1 < t_1 < \ldots < t_M \leq T$, called change points, such that *all* the signal $\beta_{t,i}$'s are the same between any two consecutive change points. Write $\boldsymbol{\beta}_t = (\beta_{t,1}, \beta_{t,2}, ..., \beta_{t,p})^\top$, $t_0 = 1$ and $t_{M+1} = T + 1$. This assumption implies that $\boldsymbol{\beta}_s = \boldsymbol{\beta}_t$ if $t_k \leq s, t < t_{k+1}$ for all $k = 0, \ldots, M$.

In addition to temporal smoothness, we also assume the signal is "spatially" smooth, in the sense that two nodes that are connected by an edge tend to have more similar values of $\beta_{t,i}$ than nodes that are not. We formalize this idea by assuming that, at any time point t, the nodes can be partitioned into different connected subgraphs in such a way that all the nodes within the same cluster share the same signal value. In below we shall call these subgraphs *clusters*. In other words, if at time t the ith and lth nodes are in the same cluster, then $\beta_{t,i} = \beta_{t,l}$. Note that the clusters may change at the change points $t_1, \ldots, t_M$.

It is straightforward to estimate the underlying signal $\beta_{t,i}$'s if the change points and the cluster structure are known; it will simply be the average of the relevant $x_{t,i,j}$'s; see (17) below. In this paper, however, we do not assume the change points nor the cluster structure are known, and we will estimate them as well as the $\beta_{t,i}$'s. We first recast this problem as a model selection problem and invoke the minimum description length (MDL) principle [1], [2] to select a best-fitting model as our final answer. As a model selection criterion, MDL defines the best model as the model that compresses the data into the shortest code length for storage. It has been shown to produce excellent results in various model selection problems in signal

Cong Xu and Thomas C. M. Lee are with the Department of Statistics, University of California, Davis, CA 95616 USA e-mail: {cngxu, tcmlee}@ucdavis.edu. The authors are most grateful to the reviewers and the associate editor for their most useful and constructive comments. They also acknowledge the support by the National Science Foundation under DMS-1811405, DMS-1811661, DMS-1916125, CCF-1934568 and DMS-2113605.

and image processing; e.g., [3], [4], [5], [6], [7]. In particular, in the context of image segmentation, it has been shown by [8] and [9] that MDL produces superior solutions when comparing to other popular model selection criteria including AIC, BIC and cross-validation. We believe that similar results will hold for the current problem and therefore MDL is used here to select a best-fitting model.

To the best of the authors' knowledge, the current paper is one of the first that considers the problem of simultaneous change point detection and node clustering for time series of graphs, although various authors have considered other different but similar problems. For example, the MDL principle was used by [10] for offline change point detection and community detection in time series of dynamic networks. Notice that the focus of their work is to model the edge behavior of the networks and no theoretical results are provided. A so-called Spectral Scan Statistic was derived by [11] to test if the signal over a given graph is constant, or is piecewise constant over two subgraphs. Lastly, a commonly studied problem is change point detection for time-varying Gaussian graphical models. A popular approach is to impose different kinds of l_1 type penalties to encourage sparsity and smoothness across time so that the entries of the precision matrix are piecewise constant or slowly varying over time; e.g., [12], [13], [14], [15].

Finally, we note that a general categorization of different types of changes in dynamic networks is proposed by [16]. The changes that we consider in this paper belongs to a specific type called "extra information changes," as the nodes in the networks contain additional information (i.e., the signal value). Another major contribution of [16] is the definitions of different scalar-valued metrics that characterize various crucial network properties at different time points. Based on these metrics, an exponentially weighted moving average control chart was designed and used for online change point detection. Other recent works for online detection for changes in link or edge properties in dynamic networks include [17], [18].

II. Methodology

To make the presentation more digestible, we begin with deriving in Section II-A the MDL solution for the case when there is no change point; i.e., the homogeneous case. Then in Section II-B we will extend to the general case that allows for change points.

A. Homogeneous Case

This subsection assumes the cluster structure stays the same across different times. That is, there is no change point and $\beta_{t,i} = \beta_i$ for all $\{t, i\}$. The task is to estimate the cluster structure, which includes the number of clusters as well as the cluster membership for each node; i.e., which cluster the node belongs to. Let d be the number of clusters (so $1 \le d \le p$) and write the cluster membership for the ith node as c_i; i.e., the ith node belongs to the c_ith cluster, where $1 \le c_i \le d$. Let $\boldsymbol{c} = \{c_1, c_2, ..., c_p\}$ and $\mathcal{P} = \{\beta_1, \beta_2, ..., \beta_p\}$. For the homogeneous case the goal is to estimate d, $\boldsymbol{c}$ and $\mathcal{P}$.

As mentioned before, MDL defines the best fitting model as the one that enables the best compression of the data, or in other words, the one that produces the shortest code length of the data. This idea can be formalized as follows. If we write $\mathrm{CL}(z)$ as the code length of z, then the code length $\mathrm{CL}(\text{"data"})$ of the observed data can be decomposed into two parts, a model $\mathcal{F}$ plus the corresponding residuals $\hat{\mathcal{E}}$:

$$\mathrm{CL}(\text{"data"}) = \mathrm{CL}(\mathcal{F}) + \mathrm{CL}(\hat{\mathcal{E}}|\mathcal{F}), \tag{1}$$

and the best model is the one that minimizes $\mathrm{CL}(\text{"data"})$. Here $\mathcal{F} = \{d, \boldsymbol{c}, \mathcal{P}\}$ and note that the dependence of $\hat{\mathcal{E}}$ on $\mathcal{F}$ is stressed in the notation of the last term.

To minimize (1) we need a computable expression for $\mathrm{CL}(\text{"data"})$ and we begin by calculating $\mathrm{CL}(\mathcal{F})$, which can be further decomposed into

$$\mathrm{CL}(\mathcal{F}) = \mathrm{CL}(d) + \mathrm{CL}(\boldsymbol{c}) + \mathrm{CL}(\mathcal{P}). \tag{2}$$

According to [1], it takes approximately $\log(I)$ bits to encode an integer I with upper bound unknown, and approximately $\log(I_u)$ bits with a known upper bound I_u. To encode the number of clusters d, we assume $d = O(p)$, which aligns with our computational algorithms below. This gives

$$\mathrm{CL}(d) = \log(d). \tag{3}$$

For $\boldsymbol{c}$, it takes $\log(d)$ bits to encode each c_i. Then we have

$$\mathrm{CL}(\boldsymbol{c}) = \sum_{i=1}^{p} \log(d) = p \log(d). \tag{4}$$

Next we calculate $\mathrm{CL}(\mathcal{P})$, and we need the maximum likelihood estimate (MLE) of the β_i's. If the ith node belongs to the rth cluster (i.e., $c_i = r$), the MLE of β_i is

$$\hat{\beta}_i = \frac{\sum_{t=1}^{T} \sum_{q, c_q = r} \sum_{j=1}^{n_{t,q}} x_{t,q,j}}{\sum_{t=1}^{T} \sum_{q, c_q = r} n_{t,q}}, \tag{5}$$

which is simply the average of all the observations of all the nodes belonging to the rth cluster at all time. By [1], to encode an MLE, the code length is $\frac{1}{2} \log N$ if N observations are used for the estimation. For $\hat{\beta}_i$, this number is given by the denominator of (5), and hence

$$\mathrm{CL}(\mathcal{P}) = \sum_{r=1}^{d} \frac{1}{2} \log(\sum_{t=1}^{T} \sum_{i, c_i = r} n_{t,i}). \tag{6}$$

Notice that although there are p of the $\hat{\beta}_i$'s, there are only d distinct values of them, as there are only $d \le p$ clusters around. Therefore the upper limit of the first summation in (6) is d not p.

Now substitute (3), (4) and (6) into (2), we have

$$\mathrm{CL}(\mathcal{F}) = (p+1) \log(d) + \sum_{r=1}^{d} \frac{1}{2} \log(\sum_{t=1}^{T} \sum_{i, c_i = r} n_{t,i}). \tag{7}$$

Lastly we calculate the last term $\mathrm{CL}(\hat{\mathcal{E}}|\mathcal{F})$ of (1), which, according to [1], is given by the negative log (base 2) of the likelihood of the fitted model. With the Gaussianity assumption $X_{t,i,j} \sim \mathcal{N}(\beta_i, \sigma^2)$ for all $\{t, i, j\}$, the negative log-likelihood is

$$\frac{n}{2} \log(2\pi) + \frac{n}{2} \log(\sigma^2) + \frac{1}{2\sigma^2} \sum_{t=1}^{T} \sum_{i=1}^{p} \sum_{j=1}^{n_{t,i}} (x_{t,i,j} - \beta_i)^2, \tag{8}$$

where $n = \sum_{t=1}^{T}\sum_{i=1}^{p} n_{t,i}$ is the total number of observations. The MLE $\hat{\beta}_i$ for β_i is given by (5), while the MLE for σ^2 is

$$\hat{\sigma}^2 = \frac{1}{n}\sum_{r=1}^{d} \text{SSE}_r,$$

where $\text{SSE}_r = \sum_{t=1}^{T}\sum_{i,c_i=r}\sum_{j=1}^{n_{t,i}}(x_{t,i,j} - \hat{\beta}_i)^2$ is the sum of squared errors of the rth cluster.

Plugging these MLEs $\hat{\beta}_i$ and $\hat{\sigma}^2$ into (8), we obtain the code length of the residuals $\hat{\mathcal{E}}$

$$\text{CL}(\hat{\mathcal{E}}|\mathcal{F}) = \frac{n}{2}\log(2\pi) + \frac{n}{2}\log(\frac{1}{n}\sum_{r=1}^{d}\text{SSE}_r) + \frac{n}{2}, \quad (9)$$

and from (1), (7) and (9), the overall code length is

$$\begin{aligned}&\text{CL}(\text{“data”}) = \text{CL}(\mathcal{F}) + \text{CL}(\hat{\mathcal{E}}|\mathcal{F})\\ =&(p+1)\log(d) + \sum_{r=1}^{d}\frac{1}{2}\log(\sum_{t=1}^{T}\sum_{i,c_i=r} n_{t,i}) + \frac{n}{2}\log(2\pi)\\ &+ \frac{n}{2}\log(\frac{1}{n}\sum_{r=1}^{d}\text{SSE}_r) + \frac{n}{2}.\end{aligned}$$

Ignoring constant terms we arrive at the following MDL criterion for the homogeneous case, and the best-fitting model is defined as its minimizer:

$$(p+1)\log(d) + \sum_{r=1}^{d}\frac{1}{2}\log(\sum_{t=1}^{T}\sum_{i,c_i=r} n_{t,i}) + \frac{n}{2}\log(\frac{1}{n}\sum_{r=1}^{d}\text{SSE}_r). \quad (10)$$

B. Heterogeneous Case

This subsection considers the heterogeneous case where the cluster structure and the signal values are allowed to change at change points. The number M and the locations $\mathcal{T} = \{t_1, t_2, ..., t_M\}$ of such change points are unknown and need to be estimated, and we will continue to use MDL. With M change points, the time line is partitioned into $M+1$ intervals, where the mth interval is $[t_{m-1}, t_m)$ for $m = 1, \ldots, M+1$. We write the number of clusters in the mth interval as $d^{(m)}$ and the cluster membership as $\boldsymbol{c}^{(m)} = \{c_1^{(m)}, c_2^{(m)}, ..., c_p^{(m)}\}$; i.e., in the mth interval the ith node belongs to the $c_i^{(m)}$th cluster. We write $\mathcal{C} = \{\boldsymbol{c}^{(1)}, \boldsymbol{c}^{(2)}, ..., \boldsymbol{c}^{(M+1)}\}$ and $\mathcal{P} = \{\boldsymbol{\beta}_1, \boldsymbol{\beta}_2, ..., \boldsymbol{\beta}_T\}$, and hence the model is $\mathcal{F} = \{\mathcal{T}, \mathcal{C}, \mathcal{P}\}$, which leads to the code length decomposition:

$$\text{CL}(\mathcal{F}) = \text{CL}(\mathcal{T}) + \text{CL}(\mathcal{C}) + \text{CL}(\mathcal{P}). \quad (11)$$

To encode $\mathcal{T}$, we first need to encode the number of the change points and then the actual locations of the change points. As there are M change points, the code length is $\log(M+1)$, where the additional 1 is used to distinguish $M = 0$ and $M = 1$. The locations of change points $\mathcal{T}$ can be encoded by using the length of the time intervals $(t_m - t_{m-1})$'s. Therefore combining the two we have

$$\text{CL}(\mathcal{T}) = \log(M+1) + \sum_{m=1}^{M}\log(t_m - t_{m-1}). \quad (12)$$

It is understood that in (12) the last term of $\text{CL}(\mathcal{T})$ reduces to 0 when there is no change point (i.e., $M = 0$), which implies $\text{CL}(\mathcal{T}) = 0$ when $M = 0$.

Once $\mathcal{T}$ is encoded, it becomes the homogeneous case for each time interval. Using similar arguments as before, we have

$$\text{CL}(\mathcal{C}) = \sum_{m=1}^{M+1}(p+1)\log(d^{(m)}) \quad (13)$$

and

$$\text{CL}(\mathcal{P}) = \sum_{m=1}^{M+1}\sum_{r=1}^{d^{(m)}}\frac{1}{2}\log(\sum_{t=t_{m-1}}^{t_m-1}\sum_{i,c_i^{(m)}=r} n_{t,i}). \quad (14)$$

Combining (11) to (14), we have

$$\begin{aligned}\text{CL}(\mathcal{F}) =& \log(M+1) + \sum_{m=1}^{M}\log(t_m - t_{m-1})\\ &+ \sum_{m=1}^{M+1}(p+1)\log(d^{(m)})\\ &+ \sum_{m=1}^{M+1}\sum_{r=1}^{d^{(m)}}\frac{1}{2}\log(\sum_{t=t_{m-1}}^{t_m-1}\sum_{i,c_i^{(m)}=r} n_{t,i}).\end{aligned} \quad (15)$$

Similarly, the code length of the residuals $\hat{\mathcal{E}}$ is

$$\text{CL}(\hat{\mathcal{E}}|\mathcal{F}) = \frac{n}{2}\log(2\pi) + \frac{n}{2}\log(\frac{1}{n}\sum_{m=1}^{M+1}\sum_{r=1}^{d^{(m)}}\text{SSE}_r^{(m)}) + \frac{n}{2}, \quad (16)$$

where

$$\text{SSE}_r^{(m)} = \sum_{t=t_{m-1}}^{t_m-1}\sum_{i,c_i^{(m)}=r}\sum_{j=1}^{n_{t,i}}(x_{t,i,j} - \hat{\beta}_{t,i})^2$$

with $\hat{\beta}_{t,i}$ being the MLE of $\beta_{t,i}$

$$\hat{\beta}_{t,i} = \frac{\sum_{s=t_{m-1}}^{t_m-1}\sum_{q,c_q^{(m)}=r}\sum_{j=1}^{n_{s,q}} x_{s,q,j}}{\sum_{s=t_{m-1}}^{t_m-1}\sum_{q,c_q^{(m)}=r} n_{s,q}}. \quad (17)$$

Now adding (15) and (16) together and omitting constant terms, the MDL criterion for the heterogeneous case is

$$\begin{aligned}\text{MDL}(\mathcal{T}, \mathcal{C}) =& \log(M+1) + \sum_{m=1}^{M}\log(t_m - t_{m-1})\\ &+ \sum_{m=1}^{M+1}(p+1)\log(d^{(m)})\\ &+ \sum_{m=1}^{M+1}\sum_{r=1}^{d^{(m)}}\frac{1}{2}\log(\sum_{t=t_{m-1}}^{t_m-1}\sum_{i,c_i^{(m)}=r} n_{t,i})\\ &+ \frac{n}{2}\log(\frac{1}{n}\sum_{m=1}^{M+1}\sum_{r=1}^{d^{(m)}}\text{SSE}_r^{(m)}).\end{aligned} \quad (18)$$

Note that in the notation of the above MDL criterion, $\mathcal{P}$ is dropped from its argument list. It is because once $\mathcal{T}$ and $\mathcal{C}$ are specified, $\mathcal{P}$ can be uniquely estimated by (17). Note also that the dependence on M of the MDL criterion is not explicitly

shown in its notation, but it is implicitly implied through $\mathcal{T}$. Lastly, notice that when there is no change point, MDL$(\mathcal{T}, \mathcal{C})$ reduces to (10).

To sum up, we propose to estimate the change points $\mathcal{T} = \{t_1, t_2, ..., t_M\}$ and the cluster structures $\mathcal{C} = \{\boldsymbol{c}^{(1)}, \boldsymbol{c}^{(2)}, ..., \boldsymbol{c}^{(M+1)}\}$ (and the signal $\mathcal{P} = \{\boldsymbol{\beta}_1, \boldsymbol{\beta}_2, ..., \boldsymbol{\beta}_T\}$) as the minimizer of MDL$(\mathcal{T}, \mathcal{C})$:

$$\{\hat{\mathcal{T}}, \hat{\mathcal{C}}\} = \arg\min_{\mathcal{T}, \mathcal{C}} \frac{1}{n} \text{MDL}(\mathcal{T}, \mathcal{C}). \tag{19}$$

Since in general (19) cannot be minimized in real time, the proposed method is for offline change point detection. This is different from online monitoring of change points, where a carefully designed control chart is typically used.

C. Theoretical Properties

This subsection establishes the statistical consistency of the MDL solution $\{\hat{\mathcal{T}}, \hat{\mathcal{C}}\}$ defined by (19). The proofs of the following results can be found in Appendixes A to E.

We need the following regularity conditions. First, for all $N > 0$, it is assumed that there exist an $N_0 > 0$ such that whenever $n > N_0$,

$$n_{t,i} > N \quad \text{for all} \quad 1 \le i \le p, 1 \le t \le T. \tag{20}$$

This condition guarantees that the numbers of observations in all node $n_{t,i}$'s go to infinity when the total number of observations n goes to infinity. Second, it is assumed that

$$\lim_{n \to \infty} \frac{n_{t,i}}{n} = \gamma_{t,i} \quad \text{for all} \quad 1 \le i \le p, 1 \le t \le T, \tag{21}$$

where the $\gamma_{t,i}$'s are some non-negative constants that sum to one. This condition ensures that the numbers of observations $n_{t,i}$'s for the nodes grow at the same linear rate.

We also assume the conditions that were listed at the beginning of Section II for the change points and signal. We denote the true model as $\{\mathcal{T}^0, \mathcal{C}^0\}$: $\mathcal{T}^0 = (t_1^0, t_2^0, ..., t_{M^0}^0)$ and $\mathcal{C}^0 = \{\boldsymbol{c}^{0(1)}, \boldsymbol{c}^{0(2)}, ..., \boldsymbol{c}^{0(M+1)}\}$, where $\boldsymbol{c}^{0(m)} = \{c_1^{0(m)}, c_2^{0(m)}, ..., c_p^{0(m)}\}$. We have the following lemma.

Lemma 1. *Suppose the total number of clusters $\sum_{m=1}^{M+1} d^{(m)}$ is known. Under the model assumptions and Conditions (20) and (21), the MDL criterion (19) gives*

$$\hat{\mathcal{T}} \to \mathcal{T}^0 \ a.s. \quad \textit{and} \quad \hat{\mathcal{C}} \to \mathcal{C}^0 \ a.s.$$

Lemma 1 is based on the assumption that the total number of clusters is known, which can be unrealistic for many real data problems. This assumption can be relaxed.

Theorem 1. *Assume the conditions of Lemma 1 with the exception that the total number of clusters $\sum_{m=1}^{M+1} d^{(m)}$ is unknown. The MDL criterion (19) gives*

$$\hat{\mathcal{T}} \to \mathcal{T}^0 \ a.s. \quad \textit{and} \quad \hat{\mathcal{C}} \to \mathcal{C}^0 \ a.s.$$

III. Practical Minimization of MDL$(\mathcal{T}, \mathcal{C})$

Even for moderate sizes of p and T, direct minimization of (18) is by no means a trivial task. This section develops a practical procedure to tackle this task. The idea is to first construct a function that can be used to generate a set of good candidate models relatively quick, and then select the final model from these candidate models as the one that gives the smallest value of MDL$(\mathcal{T}, \mathcal{C})$. We shall call such a function a *candidate model generating function*. The idea is similar to, in the context of variable selection in linear models, first apply lasso to quickly generate a set of candidate models on its solution path, and then use a model selection criterion such as BIC to select the best model from these candidate models.

We need some notation to proceed. Let $y_{t,i}$ be the average of all the observations within the ith node at time t; i.e., $y_{t,i} = \bar{x}_{t,i} = \frac{1}{n_{t,i}} \sum_{j=1}^{n_{t,i}} x_{t,i,j}$, and write $\boldsymbol{Y}_t = (y_{t,1}, y_{t,2}, ..., y_{t,p})^\top$ for $t = 1, ..., T$, and $\boldsymbol{Y} = (\boldsymbol{Y}_1^\top, \boldsymbol{Y}_2^\top, \ldots, \boldsymbol{Y}_T^\top)^\top$; hence $\boldsymbol{Y}$ is a vector of length $p \times T$. Let $\boldsymbol{n} = (n_{1,1}, \ldots, n_{1,p}, n_{2,1}, \ldots n_{2,p}, \ldots, n_{T,1}, \ldots, n_{T,p})^\top$ be the vector of the numbers of observations for the nodes, and $\boldsymbol{\beta} = (\boldsymbol{\beta}_1^\top, \boldsymbol{\beta}_2^\top, \ldots, \boldsymbol{\beta}_T^\top)^\top$ be the vector of the true signal, where $\boldsymbol{\beta}_t = (\beta_{t,1}, \beta_{t,2}, ..., \beta_{t,p})^\top$ for $t = 1, ..., T$. Then the goal is to retrieve the underlying signal $\boldsymbol{\beta}$ from its noisy version $\boldsymbol{Y}$

$$y_{t,i} = \beta_{t,i} + e_{t,i}, \quad e_{t,i} \overset{i.i.d.}{\sim} \mathcal{N}(0, \frac{\sigma^2}{n_{t,i}})$$

under the assumptions of temporal and spatial smoothness.

A. Construction of A Candidate Model Generating Function

The goal of a candidate model generating (CMG) function is to quickly generate a set of good candidate models with small values of MDL$(\mathcal{T}, \mathcal{C})$. Thus for the current problem, a good CMG function should produce models that are both temporally and spatially smooth, and yet maintain good data fidelity. One natural way to construct such a function is to combine three terms together: a penalty term that encourages temporal smoothness, a second penalty term that encourages spatial smoothness, and lastly a loss term that measures data fidelity.

We begin with the temporal smoothness assumption, which prefers signals close in time to have similar values (except at the change points); i.e., $\boldsymbol{\beta}_{t+1} \approx \boldsymbol{\beta}_t$. This suggests the following penalty term

$$\Omega_1(\boldsymbol{\beta}) = \lambda_1 \sum_{t=1}^{T-1} \|\boldsymbol{\beta}_{t+1} - \boldsymbol{\beta}_t\|_2, \tag{22}$$

where λ_1 is a tuning parameter and $\|.\|_2$ is the vector l_2-norm. This is in similar spirit as the penalty used in the fused lasso of [19] and the generalized total variation denoising method of [20].

For the spatial smoothness assumption, we borrow the idea from graph-guided-fused-lasso [21], [22] to construct the penalty term. First, let $\boldsymbol{E}$ be the set of all connected edges in the graph:

$$\boldsymbol{E} = \{(i, k) : \text{the } i\text{th and } k\text{th nodes are connected}, 1 \le i, k \le p\}.$$

Recall that the node connectivity of our graph is assumed constant over time, so $\boldsymbol{E}$ does not change over time. Next, define a matrix $\boldsymbol{G}$ in such way that if $(i,k) \in \boldsymbol{E}$, then one row of $\boldsymbol{G}$ is all zeros except the ith entry is 1 and the kth entry is -1. Note that $\boldsymbol{G}$ is of size $|\boldsymbol{E}| \times p$, and is not unique as its rows can be permuted, but it will not affect the final results. Here we suggest using the following penalty term for spatial smoothness:

$$\Omega_2(\boldsymbol{\beta}) = \lambda_2 \sum_{t=1}^{T} \|\boldsymbol{G}\boldsymbol{\beta}_t\|_1, \tag{23}$$

where λ_2 is a tuning parameter and $\|.\|_1$ is the vector l_1-norm.

Lastly, we need a data fidelity term and a natural candidate is the loss

$$l(\boldsymbol{\beta}|\boldsymbol{Y},\boldsymbol{n}) = \sum_{t=1}^{T}\sum_{i=1}^{p} \frac{n_{t,i}}{2}(y_{t,i} - \beta_{t,i})^2. \tag{24}$$

Combining (22), (23) and (24), our CMG function is

$$\begin{aligned} f(\boldsymbol{\beta}|\boldsymbol{Y},\boldsymbol{n}) &= l(\boldsymbol{\beta}|\boldsymbol{Y},\boldsymbol{n}) + \Omega_1(\boldsymbol{\beta}) + \Omega_2(\boldsymbol{\beta}) \\ &= l(\boldsymbol{\beta}|\boldsymbol{Y},\boldsymbol{n}) + \Omega(\boldsymbol{\beta}), \end{aligned} \tag{25}$$

where $\Omega(\boldsymbol{\beta}) = \Omega_1(\boldsymbol{\beta}) + \Omega_2(\boldsymbol{\beta})$. Thus, given a pair of (λ_1, λ_2), one can generate a good candidate model by minimizing (25).

B. Generating Candidate Models with the CMG Function

Although the penalty $\Omega(\boldsymbol{\beta})$ is not smooth, (25) can still be approximately minimized in the following manner. First, using the smoothing proximal gradient method of [23], we obtain a smooth approximation of $\Omega(\boldsymbol{\beta})$ so that its gradient with respect to $\boldsymbol{\beta}$ can be derived. Then we apply the fast iterative shrinkage-thresholding algorithm (FISTA) of [24] to carry out the minimization. This procedure is summarized in Algorithm 1, and technical details such as the smooth approximation of $\Omega(\boldsymbol{\beta})$ are referred to Appendixes F to J.

The time complexity of Algorithm 1 can be as low as $O(k_{\max}T(|\boldsymbol{E}| + p))$, as long as sparsity is utilized in those matrix multiplications involved in the algorithm. Notice that this algorithm needs to be applied multiple times for different pairs of (λ_1, λ_2). Also notice that a global optimization is virtually infeasible, as the time complexity for change point detection is $O(T^2)$ if one uses dynamic programming [25], and the time complexity for node clustering for each interval is of polynomial rate. Therefore, the use of Algorithm 1 offers substantial computational advantages.

We note that the output from Algorithm 1 does not produce exactly the same value for $\beta_{t,i}$'s that belong to the same time interval and cluster. For example, suppose for a certain node Algorithm 1 returns $\tilde{\boldsymbol{\beta}} = (1.0, 1.1, 0.9, 2.3, 2.2, 2.3, 2.2)$ for $t = 1, \ldots 7$, which signifies there is a change point at $t = 4$. To circumvent this issue, we conduct a fast scanning operation that will adjust the values to $\hat{\boldsymbol{\beta}} = (1.0, 1.0, 1.0, 2.25, 2.25, 2.25, 2.25)$. Details of the scanning operation are summarized as Algorithms 2 and 3 in Appendix K. Note that the time complexity for this scanning operation is $O(Tp)$.

Thus, by performing the above steps, we can quickly obtain a good candidate model $\{\hat{\mathcal{T}}, \hat{\mathcal{C}}\}$ for a given pair of (λ_1, λ_2). As an optional step, we can generate more good candidate models by perturbing $\{\hat{\mathcal{T}}, \hat{\mathcal{C}}\}$, such as removing a change point in $\hat{\mathcal{T}}$.

Lastly, we comment on the choice of (λ_1, λ_2), for which in practice depends on the scale of the observations. Specifically, a large T usually requires large values of λ_1, while λ_2 depends on the number of edges $|\boldsymbol{E}|$ of the pre-specified graph $\boldsymbol{G}$. One may start with choosing a range of values $(\lambda_1^{\min}, \ldots, \lambda_1^{\max})$ for λ_1 and another range $(\lambda_2^{\min}, \ldots, \lambda_2^{\max})$ for λ_2. Then calculate the MDL$(\mathcal{T}, \mathcal{C})$ values of all the candidate models obtained from every pair of (λ_1, λ_2). If the minimum value of MDL$(\mathcal{T}, \mathcal{C})$ occurs with a candidate model that corresponds to $\lambda_1 = \lambda_1^{\min}$ or $\lambda_1 = \lambda_1^{\max}$, one would need to decrease the value of $\lambda_1^{\min}$ or increase the value of $\lambda_1^{\max}$; similarly for λ_2. Otherwise, one can deem that the original choices for the ranges for λ_1 and λ_2 are reasonable. Also, after identifying such suitable ranges for λ_1 and λ_2, one can try more choices of (λ_1, λ_2) within these ranges (i.e. higher resolution) to achieve better results. See also [26] [27] for practical methods for choosing the initial ranges of values for λ_1 and λ_2.

Algorithm 1 FISTA for minimizing (25)

Require: $\boldsymbol{Y}$, $\boldsymbol{n}$, $\boldsymbol{C}$ derived by (43), (44) and (45), $\boldsymbol{\beta}^{[0]}$, Lipschitz constant L derived by (49) or (50), D derived by (46), desired accuracy ε, λ_1, λ_2, $k_{\max}$

$\mu = \frac{\varepsilon}{2D}$, $\theta_0 = 1$
for $k = 0, 1, \ldots$ until $\boldsymbol{\beta}^{[k]}$ converges or $k \geq k_{\max}$ **do**
  Compute $\boldsymbol{\alpha}^{*[k]}$ based on $\boldsymbol{\beta}^{[k]}$ by (47) and (48)
  $\nabla h(\boldsymbol{w}^{[k]}) \leftarrow \boldsymbol{n}(\boldsymbol{w}^{[k]} - \boldsymbol{X}) + \boldsymbol{C}^\top \boldsymbol{\alpha}^{*[k]}$
  $\boldsymbol{\beta}^{[k+1]} \leftarrow \boldsymbol{w}^{[k]} - \frac{1}{L}\nabla h(\boldsymbol{w}^{[k]})$
  $\theta_{k+1} \leftarrow \frac{2}{k+3}$
  $\boldsymbol{w}^{[k+1]} \leftarrow \boldsymbol{\beta}^{[k+1]} + \frac{1-\theta_k}{\theta_k}\theta_{k+1}(\boldsymbol{\beta}^{[k+1]} - \boldsymbol{\beta}^{[k]})$
end for
return $\tilde{\boldsymbol{\beta}} = \boldsymbol{\beta}^{[k+1]}$

C. Summary

The minimization for MDL$(\mathcal{T}, \mathcal{C})$ defined by (18) can be summarized by the following steps:

1) Given (λ_1, λ_2), apply Algorithm 1 to minimize (25) to obtain $\tilde{\boldsymbol{\beta}}$.
2) Apply Algorithms 2 and 3 to $\tilde{\boldsymbol{\beta}}$ to obtain a good candidate model $\{\hat{\mathcal{T}}, \hat{\mathcal{C}}\}$.
3) (Optional) Perturb $\hat{\mathcal{T}}$ to generate more $\{\hat{\mathcal{T}}, \hat{\mathcal{C}}\}$'s.
4) Repeat Steps 1 to 3 with different values of (λ_1, λ_2) to obtain more $\{\hat{\mathcal{T}}, \hat{\mathcal{C}}\}$'s.
5) Calculate the MDL$(\mathcal{T}, \mathcal{C})$ values for all $\{\hat{\mathcal{T}}, \hat{\mathcal{C}}\}$'s obtained from Step 4. Take the one that gives the smallest value as the minimizer of MDL$(\mathcal{T}, \mathcal{C})$.

IV. Simulation Experiments

A. Setting 1: Regular Grid

In this first experiment the graph structure was a square image of size 8×8. That is, there were $p = 64$ nodes arranged as an 8×8 two-dimensional grid, and each node

was connected to its 4 neighboring nodes, except for those nodes at the edges and corners of the grid, where they were connected to, respectively, 3 and 2 neighboring nodes. We set $T = 100$ and had change points at $t = 25$, 50, 60 and 90. The nodes were partitioned into two groups and for each time segment, all the nodes within the same group share the same true signal $\beta_{t,i}$ value. The true signal values are reported in Table I and they are visually displayed in Figure 1. All the $n_{t,i}$'s were set to 1.

segment	interval	cluster sizes	values
1	[0, 25)	16, 48	2, 1
2	[25, 50)	16, 48	2.2, 1
3	[50, 60)	26, 38	2.1, 1
4	[60, 90)	26, 38	2.4, 1
5	[90, 100)	35, 29	2.4, 1

TABLE I: True signal values used for Experimental Setting 1.

Gaussian noise with variance $\sigma^2 \in \{0.1^2, 0.2^2, 0.3^2, 0.4^2\}$ was added to the true signal to generate the noisy observations $x_{t,i,j}$, with 100 repetitions for each value of σ^2. For each noisy data set, 25 combined values of $\lambda_1 \in \{0.5, 1, 2, 4, 8, 16\}$ and $\lambda_2 \in \{0.5, 1, 2, 4, 8, 16\}$ were used in Algorithm 1 to obtain the MDL solution.

Figure 2 presents the results of this numerical experiment. The histograms show the locations of all the detected change points for the 100 repetitions - recall that these change points are defined as the joint minimizer of (19). As expected, the larger the noise variance, the more difficult to detect the change points. This phenomenon is more obvious for those change points where the changes of the true signal values were small: $t = 25$ and 60. To be more specific, the only difference in the true signal before and after the change point at $t = 25$ was the value for the top-left region. As the noise level increases, it becomes more difficult to detect this change point. A similar phenomenon was observed for the change point at $t = 60$.

Apart from reporting the histograms of the detected change points, we also evaluated the quality of the signal estimates $\hat{\beta}_{t,i}$ in terms of mean squared error (MSE):

$$\text{MSE} = \frac{1}{\sum_{t=1}^{T}\sum_{i=1}^{p} n_{t,i}} \sum_{t=1}^{T}\sum_{i=1}^{p} n_{t,i}(\hat{\beta}_{t,i} - \beta_{t,i})^2.$$

We report the MSE results in a similar fashion as [28]. First, define the negative signal-to-noise ratio (SnR) as

$$10\log_{10}\left[\left\{\sum_{t=1}^{T}\sum_{i=1}^{p}\frac{\sigma^2}{n_{t,i}}\right\} \Big/ \left\{\sum_{t=1}^{T}\sum_{i=1}^{p}(\beta_{t,i} - \bar{\beta})^2\right\}\right].$$

Thus, the negative SnR increases as the noise level increases. Next, define the denoised negative SnR as

$$10\log_{10}\left[\text{MSE} \Big/ \left\{\frac{1}{Tp}\sum_{t=1}^{T}\sum_{i=1}^{p}(\beta_{t,i} - \bar{\beta})^2\right\}\right],$$

and hence the smaller the denoised negative SnR is, the better the estimates $\beta_{t,i}$'s are. We compared the results obtained from the proposed method with their corresponding saturated models: here a saturated model was the model with a separate parameter $\beta_{t,i}$ fitted for each node. In order to verify the importance of both the temporal and spatial smoothness assumptions, we also compared the performances of the version of the proposed method without the temporal smoothness assumption (i.e., forcing $\lambda_1 = 0$) and the version without the spatial smoothness assumption (i.e., forcing $\lambda_2 = 0$). The results are also reported in Figure 2. As the noise level increases, the denoised negative SnRs for both the MDL fitted model and the saturated model increase. Compared with the saturated models, the denoised negative SnRs for the MDL fitted models are smaller, even more so for those cases with high noise levels. Note also that, from Figure 2(e), both the versions of no smoothness assumption and no temporal smoothness gave inferior performances when compared with the proposed method. Similar results can also be seen in the next two experiments.

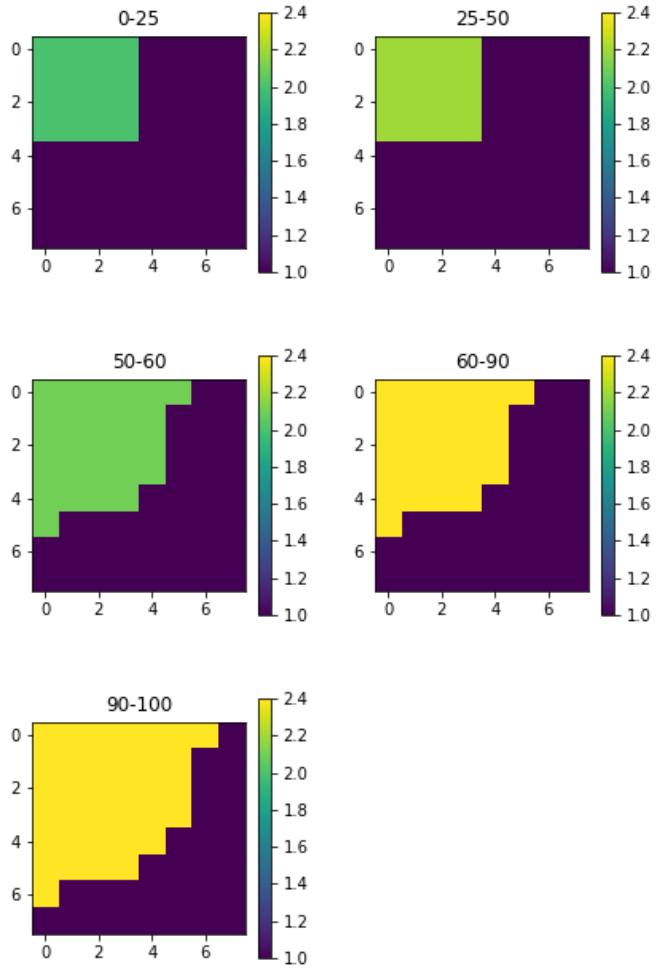
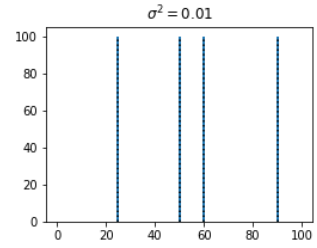


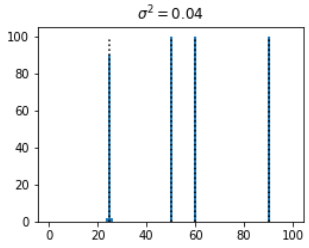
Fig. 1: True signal values for Experimental Setting 1.

B. Setting 2: Graph based on California Counties

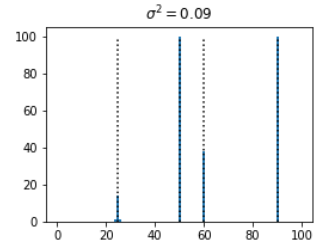
In this second experiment the graph structure was defined by the 58 counties in California. Each county was a node, and two nodes were connected if the two corresponding counties share a common border. So there were 58 nodes and 136 edges; see Figure 7(a). We partitioned the nodes into 4 groups, and the number of time points was $T = 60$ with change points at



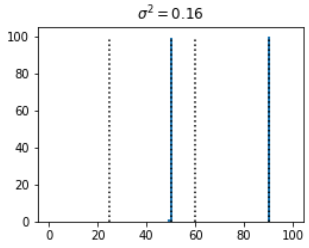
(a) change points, $\sigma^2 = 0.01$



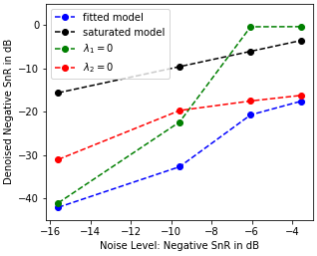
(b) change points, $\sigma^2 = 0.04$



(c) change points, $\sigma^2 = 0.09$



(d) change points, $\sigma^2 = 0.16$



(e) negative SnR

Fig. 2: (a)-(d) Histograms of the detected change points under different noise levels. (e) Denoised negative SnRs for different noise levels. Recall that a saturated model is a model with a separate parameter fitted for each node.

$t = 10, 20, 35$ and 45. For each time segment, all the nodes within the same group share the same true signal $\beta_{t,i}$ value; see Table II and Figure 3. Note that these signal values were selected so that the overall signal averages were the same for all the time intervals. Consequently, any univariate change point detection method will fail when it is applied to the (univariate) time series of combined signal values for all time points, as the important graph structure information is ignored.

segment	interval	cluster sizes	values
1	[0, 10)	10, 17, 12, 19	10, 20, 30, 40
2	[10, 20)	10, 17, 12, 19	10, 31.18, 40, 20
3	[20, 35)	10, 17, 12, 19	20, 31.18, 31.67, 20
4	[35, 45)	10, 17, 12, 19	20, 40, 20, 19.47
5	[45, 50)	10, 17, 12, 19	30, 25.29, 31.67, 20

TABLE II: True signal values used for Experimental Setting 2.

We tested the proposed method with 6 difference noise variance $\sigma^2 \in \{1^2, 2^2, 3^2, 4^2, 6^2, 8^2\}$ and 36 combined values of $\lambda_1 \in \{2, 4, 8, 16, 32, 64\}$ and $\lambda_2 \in \{0.5, 1, 2, 4, 8, 16\}$. As before, the number of repetitions was 100. The histograms of the detected change points are given in Figure 4, as well as the denoised negative SnRs. Similar empirical conclusions can be drawn as before: the larger the noise level, the more difficult to detect the change points.

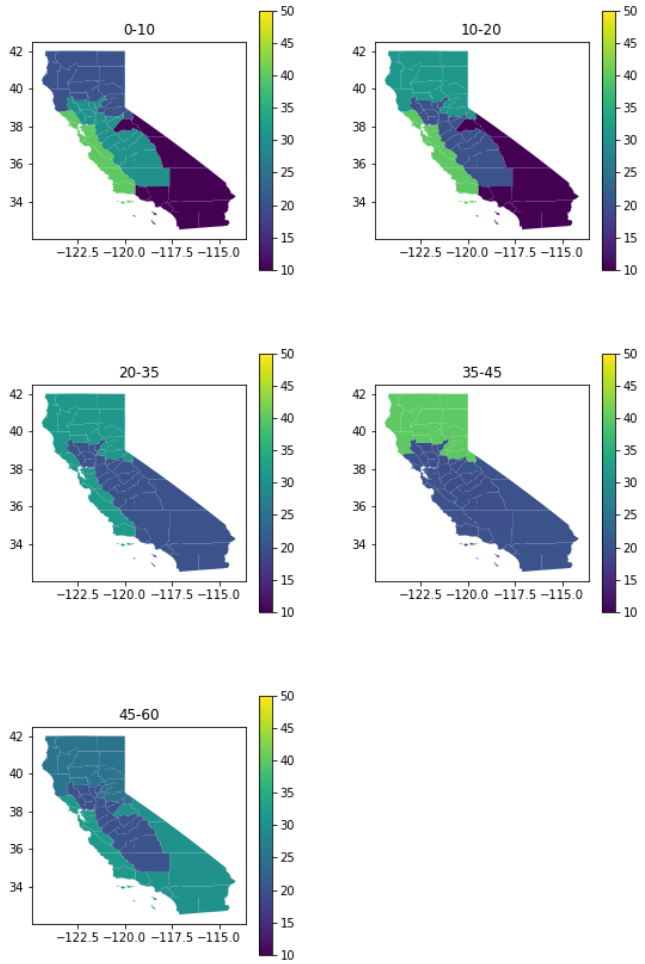


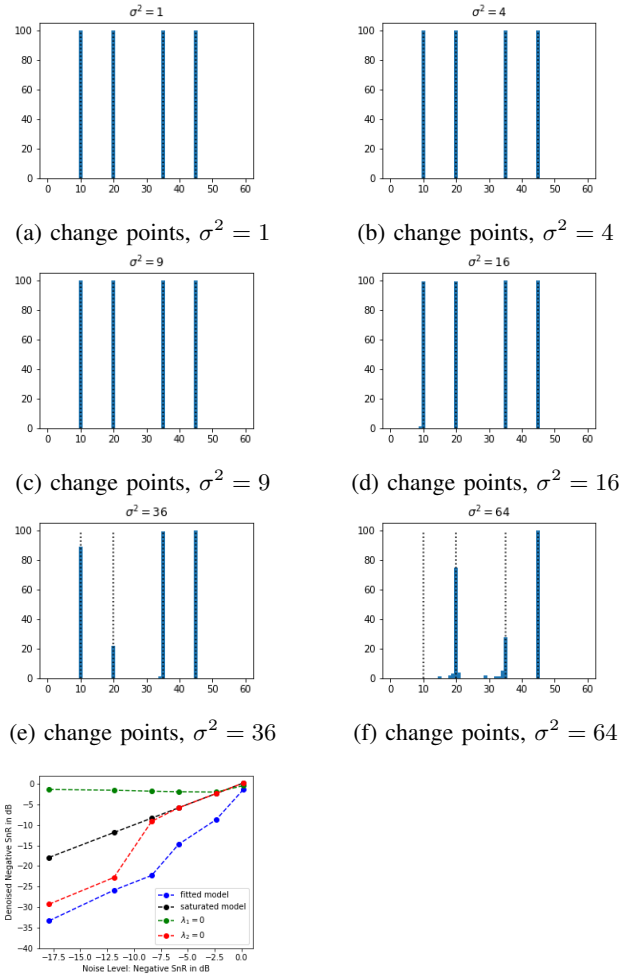
Fig. 3: True signal values for Experimental Setting 2.

C. Setting 3: Regular Grid with No Change Points

In this last experiment we tested the performance of the proposed algorithm when there were no change points. The graph structure is the same as that in Setting 1. The number of time points was $T = 60$ and all $n_{t,i}$'s were set to 1. We partitioned the nodes into 3 groups, and their true signal values are displayed in Figure 5.

One hundred Gaussian noisy data sets were generated for each $\sigma^2 \in \{0.1^2, 0.2^2, 0.3^2, 0.4^2\}$. For each simulated data set, 63 combined values of $\lambda_1 \in \{0, 0.5, 1, 2, 4, 8, 16\}$ and $\lambda_2 \in \{0, 0.125, 0.25, 0.5, 1, 2, 4, 8, 16\}$ were used in Algorithm 1 to obtain the MDL solution.

The proposed method and the version without the spatial smoothness assumption (i.e., $\lambda_2 = 0$) performed perfectly in terms of change point detection; i.e. no change point detected. However, the version without the temporal smoothness assumption (i.e., $\lambda_1 = 0$) detected many false change points before perturbing $\hat{\mathcal{T}}$. The denoised SnRs can be found in Figure 5. One can see that the proposed method outperformed the version without spatial smoothness.



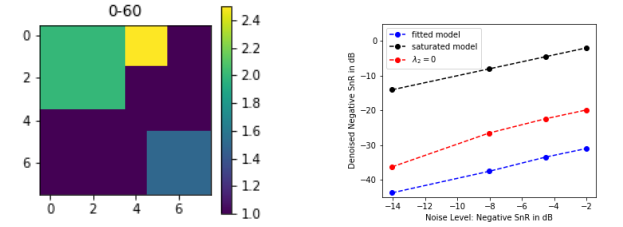
(a) change points, $\sigma^2 = 1$ (b) change points, $\sigma^2 = 4$

(c) change points, $\sigma^2 = 9$ (d) change points, $\sigma^2 = 16$

(e) change points, $\sigma^2 = 36$ (f) change points, $\sigma^2 = 64$

(g) negative SnR

Fig. 4: (a)-(f) Histograms of the detected change points under different noise levels. (g) Denoised negative SnRs for different noise levels.



(a) true signal values (b) negative SnR

Fig. 5: (a) True signal values for Experimental Setting 3. (b) Denoised negative SnRs for different noise levels.

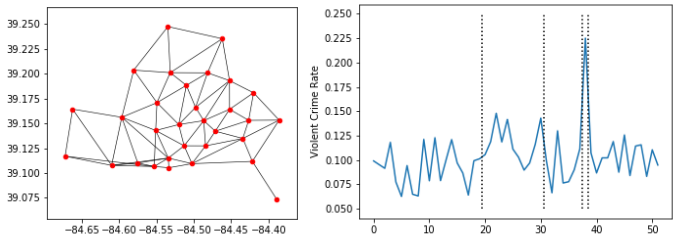
V. Real Data Applications

A. Violent Crime in Cincinnati, OH

The data set in this subsection concerns reported crime incidents in Cincinnati, OH. It contains dates, times, locations, and other information about the reported events. We considered weekly crime rates from December 31, 2018, to December 29, 2019; i.e., $T = 52$. The data can be obtained from this website[1].

Each crime event has an FBI Uniform Crime Reporting code that describes its type. As similar to [29], we used this code to classify each crime event into violent crime or non-violent crime: a violent crime can be homicide, rape, aggravated assault, or robbery, while all the other types of crimes are non-violent.

The nodes were defined by ZIP Code Tabulation Areas in Cincinnati, and edges were defined by geographical neighborhoods. There were 31 nodes and 77 edges in the graph; see Figure 6(a). During the t-th week and at the i-th node, the number of observations $n_{t,i}$ was the total number of reported crime events, while the j-th measurement $x_{t,i,j}$ was 1 if the j-th crime was violent, and 0 otherwise. Thus, the data was in fact binomial and we modified the likelihood function in the MDL criterion (18) to reflect this. The justification for this extension can be found in Appendix L.



(a) (b)

Fig. 6: (a) The graph structure defined by the ZIP Code Tabulation Areas in Cincinnati, OH. (b) Violent crime rate for each week from 2018-12-31 to 2019-12-29 in Cincinnati, OH. Vertical lines denote detected change point locations .

Change points were detected at 2019-05-20, 2019-08-05, 2019-09-23 and 2019-09-30. The weekly overall violent crime rates, together with these 4 change points, are displayed in Figure 6(b). [30] studied the relationships between temperature and different kinds of crimes. The author concluded that higher temperatures lead to statistically significant increases in all types of crimes. However, the rate of increase is approximately constant for violent crimes, while for non-violent crimes, the rate of increase starts to slow down around 50 °F. Therefore, the first detected change point (late May) signifies the beginning of summer and hence an increased rate of violent crime. The second detected change point (early August) was close to the end of the peak travel season which may explain the drop in violent crime rates. The last two change points together actually suggest that the week in between was an outlier. In fact, that week included the last weekend before Halloween, and it is known that the violent crime rate (e.g., robbery and sexual assault) increases shortly before or at Halloween.

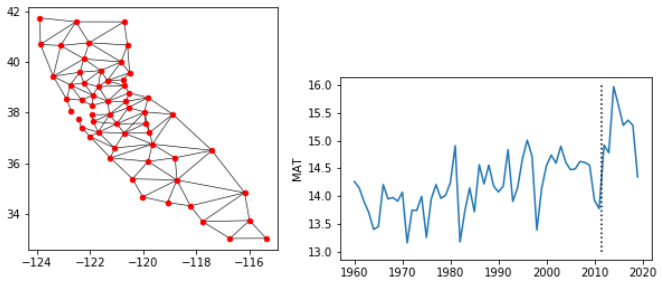
B. Temperatures in Counties in California

The data set is the output of PRISM (parameter-elevation regressions on independent slopes model), a combination of

[1] https://data.cincinnati-oh.gov/Safety/PDI-Police-Data-Initiative-Crime-Incidents/k59e-2pvf

statistical and human-expert methods for climate mapping [31]. It contains different readings such as temperatures and precipitation. In this study, we considered mean annual temperatures from 1960 to 2019 in 58 counties in California. We collected data at the grids of 0.2×0.2 degrees of longitude/latitude from this website[2].

The graph structure was defined by the 58 counties in California, in the same manner as in Section IV-B; see Figure 7(a). The proposed method was applied and detected one change point in the year 2012. We plotted the average temperature of the whole of California in Figure 7(b), together with the detected change point. It seems that the mean annual temperatures after the change points are higher than those before the change point, hence supporting a warming trend in California [32].



(a) (b)

Fig. 7: (a) The graph structure defined by the counties in California. (b) Mean annual temperatures (°C) of California in 1960-2019. The vertical line indicates the detected change point at 2012.

VI. Concluding Remarks

This paper proposed a method for simultaneous change point detection and node clustering for time-evolving graphs. The method is composed of two major components: (i) an MDL criterion for which the best fitting model is defined as its minimizer, and (ii) a practical algorithm for finding this minimizer. It is shown that the MDL criterion yields statistically consistent estimates, while simulation results suggest that the method also enjoys highly desirable empirical properties.

Future work includes extending the piecewise constant assumption to piecewise linear or even quadratic fitting, for accommodating more signal trends. Another possible extension is to relax the iid noise assumption. For example, different time intervals can have different noise levels, or the noise can be temporally and/or spatially correlated. One could also allow for outliers in the observations, or place different weights on the nodes. Lastly, if the Gaussian noise assumption is violated, say verified by performing diagnostic checking with the residuals, then other noise assumptions could be used in place. A good example was given in Section V-A, where a binomial distribution was used to model the crime incident data. In general, it should be relatively straightforward to derive a tailored MDL criterion for any of these extensions. The major challenge is then, how to practically minimize the criterion.

[2]http://www.prism.oregonstate.edu/explorer/map.php

Appendix

A. Proof of Lemma 1

Let $\mathcal{B}$ be a probability 1 set. For each $\omega \in \mathcal{B}$, suppose on the contrary $\hat{\mathcal{T}} \nrightarrow \mathcal{T}^0$ or $\hat{\mathcal{C}} \nrightarrow \mathcal{C}^0$. As the numbers of time points and nodes are finite, the possible values for $\mathcal{T}$ and $\mathcal{C}$ are finite. Therefore, there exists a subsequence $\{n_k\}$ such that $\hat{\mathcal{T}} \to \mathcal{T}^*$ and $\hat{\mathcal{C}} \to \mathcal{C}^*$ for some $\mathcal{T}^*$ and $\mathcal{C}^*$ as k increases.

To simplify the notation for the set of indices in the same cluster, $\sum_{\{i|1\le i\le p, c_i^{(m)}=r\}}$ is written as $\sum_{i,c_i^{(m)}=r}$ in below. Similarly, $\sum_{\{i|1\le i\le p, c_i^{*(m)}=r, c_i^{0(s)}=l\}}$ is written as $\sum_{i,c_i^{*(m)}=r,c_i^{0(s)}=l}$.

It is convenient to define the set $R^*(m,r)$ that collects all the time and node indices belonging to the m-th interval and r-th cluster:

$$R^*(m,r) = \{(t,i)|t^*_{m-1}+1 \le t \le t^*_m, c_i^{*(m)} = r\}. \quad (26)$$

Therefore if during the interval $\{t^*_{m-1}+1, \le t^*_m\}$ the ith node belongs to the rth cluster (i.e., $c_i^{*(m)} = r$), then its signal estimate $\hat{\beta}_i^{*(m)}$ is given by the sample mean of all the observations $x_{t,i,j}$'s such that $(t,i) \in R^*(m,r)$. We denote this sample mean as $\hat{\beta}(R^*(m,r))$, and we have

$$\hat{\beta}_i^{*(m)} = \hat{\beta}(R^*(m,r)) = \frac{\sum_{t=t^*_{m-1}}^{t^*_m-1}\sum_{i,c_i^{*(m)}=r}\sum_{j=1}^{n_{t,i}} x_{t,i,j}}{\sum_{t=t^*_{m-1}}^{t^*_m-1}\sum_{i,c_i^{*(m)}=r} n_{t,i}}. \quad (27)$$

To simplify notations, we replace n_k by n. For large enough n,

$$\begin{aligned}
\frac{1}{n}\mathrm{MDL}(\hat{\mathcal{T}},\hat{\mathcal{C}}) &= \frac{1}{n}\log(M+1) + \frac{1}{n}\sum_{m=1}^{M}\log(t^*_m - t^*_{m-1}) \\
&\quad + \frac{1}{n}\sum_{m=1}^{M+1}(p+1)\log(d^{(m)}) \\
&\quad + \frac{1}{n}\sum_{m=1}^{M+1}\sum_{r=1}^{d^{(m)}}\frac{1}{2}\log\Big(\sum_{t=t^*_{m-1}}^{t^*_m-1}\sum_{i,c_i^{*(m)}=r} n_{t,i}\Big) \\
&\quad + \frac{1}{n}\frac{n}{2}\log\Big(\frac{1}{n}\sum_{m=1}^{M+1}\sum_{r=1}^{d^{(m)}}\mathrm{SSE}_r^{*(m)}\Big) \\
&= c_n + \frac{1}{2}\log\Big(\frac{1}{n}\sum_{m=1}^{M+1}\sum_{r=1}^{d^{(m)}}\mathrm{SSE}_r^{*(m)}\Big). \quad (28)
\end{aligned}$$

In the above c_n is of order $O(\log(n)/n)$ and

$$\mathrm{SSE}_r^{*(m)} = \sum_{t=t^*_{m-1}}^{t^*_m-1}\sum_{i,c_i^{*(m)}=r}\sum_{j=1}^{n_{t,i}}(x_{t,i,j} - \hat{\beta}_i^{*(m)})^2. \quad (29)$$

As $(\mathcal{T}^*,\mathcal{C}^*) \neq (\mathcal{T}^0,\mathcal{C}^0)$, for each $R^*(m,r)$, there are two possible cases, to be examined below.

1) Case 1: If $R^*(m,r) \subseteq R^0(s,l)$, that is to say, $R^*(m,r)$ is totally within a true $R^0(s,l) = \{(t,i)|t^0_{s-1}+1 \le t \le t^0_s, c_i^{0(s)} = l\}$, then $\forall (t,i) \in R^*(m,r) \subseteq R^0(s,l)$, $x_{t,i,j} \sim \mathcal{N}(\beta^{(l)}_{(s)}, \sigma^2)$ *i.i.d.* ($\beta^{(l)}_{(s)}$ denotes the common mean shared by all the nodes in $R^0(s,l)$). Then from (27) and the strong law of large number, $\hat{\beta}_i^{*(m)} = \hat{\beta}(R^*(m,r)) \to \beta^{0(l)}_{(s)}$ *a.s.* Also, from (29)

$$\begin{aligned}\frac{1}{n}\text{SSE}_r^{*(m)} &= \frac{1}{n}\sum_{t=t^*_{m-1}}^{t^*_m-1}\sum_{i,c_i^{*(m)}=r}\sum_{j=1}^{n_{t,i}}(x_{t,i,j}-\hat{\beta}_i^{*(m)})^2 \\ &\to \sum_{t=t^*_{m-1}}^{t^*_m-1}\sum_{i,c_i^{*(m)}=r}\gamma_{t,i}\sigma^2 \ a.s., \end{aligned} \tag{30}$$

where $\gamma_{t,i}$ is defined in (20).

2) Case 2: If $R^*(m,r) \subseteq \cup_{(s,l)\in \boldsymbol{S}} R^0(s,l)$ and $R^*(m,r) \cap R^0(s,l) \neq \emptyset, \forall (s,l) \in \boldsymbol{S}$, which is that same as saying $R^*(m,r)$ has nontrivial intersection with more than one true $R^0(s,l)$, then

$$\begin{aligned}&\hat{\beta}_i^{*(m)} = \hat{\beta}(R^*(m,r)) \\ =&\frac{\sum_{t=t^*_{m-1}}^{t^*_m-1}\sum_{i,c_i^{*(m)}=r}\sum_{j=1}^{n_{t,i}} x_{t,i,j}}{\sum_{t=t^*_{m-1}}^{t^*_m-1}\sum_{i,c_i^{*(m)}=r} n_{t,i}} \\ =&\frac{\sum_{(s,l)\in\boldsymbol{S}}\sum_{t=\max\{t^*_{m-1},t^*_{s-1}\}}^{\min\{t^*_m,t^*_s\}-1}\sum_{i,c_i^{*(m)}=r,c_i^{0(s)}=l}\sum_{j=1}^{n_{t,i}} x_{t,i,j}}{\sum_{(s,l)\in\boldsymbol{S}}\sum_{t=\max\{t^*_{m-1},t^0_{s-1}\}}^{\min\{t^*_m,t^0_s\}-1}\sum_{i,c_i^{*(m)}=r,c_i^{0(s)}=l} n_{t,i}} \\ \to&\frac{\sum_{(s,l)\in\boldsymbol{S}}\sum_{t=\max\{t^*_{m-1},t^0_{s-1}\}}^{\min\{t^*_m,t^0_s\}-1}\sum_{i,c_i^{*(m)}=r,c_i^{0(s)}=l}\gamma_{t,i}\beta^{0(l)}_{(s)}}{\sum_{(s,l)\in\boldsymbol{S}}\sum_{t=\max\{t^*_{m-1},t^0_{s-1}\}}^{\min\{t^*_m,t^0_s\}-1}\sum_{i,c_i^{*(m)}=r,c_i^{0(s)}=l}\gamma_{t,i}} \ a.s.\end{aligned}$$

And

$$\begin{aligned}&\frac{1}{n}\text{SSE}_r^{*(m)} \\ =&\frac{1}{n}\sum_{t=t^*_{m-1}}^{t^*_m-1}\sum_{i,c_i^{*(m)}=r}\sum_{j=1}^{n_{t,i}}(x_{t,i,j}-\hat{\beta}_i^{*(m)})^2 \\ =&\frac{1}{n}\sum_{(s,l)\in\boldsymbol{S}}\sum_{t=\max\{t^*_{m-1},t^0_{s-1}\}}^{\min\{t^*_m,t^0_s\}-1}\sum_{i,c_i^{*(m)}=r,c_i^{0(s)}=l} \\ &\sum_{j=1}^{n_{t,i}}(x_{t,i,j}-\hat{\beta}_i^{*(m)})^2 \\ \ge&\frac{1}{n}\sum_{(s,l)\in\boldsymbol{S}}\sum_{t=\max\{t^*_{m-1},t^0_{s-1}\}}^{\min\{t^*_m,t^0_s\}-1}\sum_{i,c_i^{*(m)}=r,c_i^{0(s)}=l} \\ &\sum_{j=1}^{n_{t,i}}(x_{t,i,j}-\hat{\beta}_{(l)}^{0(s)})^2 \\ \to&\sum_{(s,l)\in\boldsymbol{S}}\sum_{t=\max\{t^*_{m-1},t^0_{s-1}\}}^{\min\{t^*_m,t^0_s\}-1}\sum_{i,c_i^{*(m)}=r,c_i^{0(s)}=l} \\ &\sum_{j=1}^{n_{t,i}}\gamma_{t,i}\sigma^2 \ a.s.\end{aligned} \tag{31}$$

Here the strict inequalities hold for at least one (m,r) because $(\mathcal{T}^*,\mathcal{C}^*) \neq (\mathcal{T}^0,\mathcal{C}^0)$ and the total number of clusters $\sum_{m=1}^{M+1} d^{(m)}$ is known.

Thus, combining (28), (30) and (31), for large enough n, $\frac{1}{n}\text{MDL}(\hat{\mathcal{T}},\hat{\mathcal{C}}) = c_n + \frac{1}{2}\log(\frac{1}{n}\sum_{m=1}^{M+1}\sum_{r=1}^{d^{(m)}}\text{SSE}_r^{*(m)}) > c_n + \frac{1}{2}\log(\sigma^2) = \frac{1}{n}\text{MDL}(\mathcal{T}^0,\mathcal{C}^0) \ge \frac{1}{n}\text{MDL}(\hat{\mathcal{T}},\hat{\mathcal{C}})$, which is a contradiction. This comes to the conclusion that $(\hat{\mathcal{T}},\hat{\mathcal{C}}) \to (\mathcal{T}^0,\mathcal{C}^0)$ *a.s.* when the total number of clusters $\sum_{m=1}^{M+1} d^{(m)}$ is known.

B. Lemma 2 and Its Proof

Lemma 2. *Assume the setting of Lemma 1 with the exception that the total number of clusters $\sum_{m=1}^{M+1} d^{(m)}$ is unknown. If the change points and the cluster structures are estimated by (19), then*

1) *The number of change points cannot be underestimated; i.e., $\hat{M} \ge M^0$ for large enough n.*
2) *The true change points $\mathcal{T}^0$ are a subset of the estimated $\hat{\mathcal{T}}$; i.e., the true change points can be identified for large enough n.*
3) *For large enough n and each $1 \le m \le \hat{M}$ with its corresponding s such that $t_{s-1}+1 \le t_{m-1}+1 < t_m \le t_s$, there exists a true $R^0(s,l)$ such that*

$$\hat{R}(m,r) \subseteq R^0(s,l)$$

for any of the fitted $\hat{R}(m,r)$. (Here $\hat{R}(m,r)$ and $R^0(s,l)$ are defined in the similar manner as (26).) In other words, the cluster structure cannot be underestimated.

The proof of Lemma 2 follows the proof of Lemma 1. If Case 2 applies, there will be a contradiction. This finishes the proof.

C. Lemma 3 and Its Proof

Lemma 3. *For k independent $\hat{U}_i \sim \mathcal{N}(\mu, \frac{\sigma^2}{n_i})$, let $\hat{U} = \frac{1}{n}\sum_{i=1}^k n_i\hat{U}_i$, where $n = \sum_{i=1}^k n_i$. We have $\sum_{i=1}^k n_i(\hat{U}_i - \hat{U})^2 \sim \sigma^2\chi^2_{k-1}$.*

Proof: Let $V_i = \sqrt{n_i}\hat{U}_i \sim \mathcal{N}(\sqrt{n_i}\mu, \sigma^2)$ and $\boldsymbol{V} = (V_1,...,V_k)^\top$. Define an orthonormal matrix $\boldsymbol{A} = (a_1,...,a_k)^\top$ with $a_1^\top = (\frac{\sqrt{n_1}}{\sqrt{n}},...,\frac{\sqrt{n_k}}{\sqrt{n}})$. Therefore, $\hat{U} = \frac{1}{\sqrt{n}}a_1^\top\boldsymbol{V} \sim \mathcal{N}(\mu, \frac{\sigma^2}{n})$. By the property of orthonormal matrices, $E(a_i^\top\boldsymbol{V}) = a_i^\top(\sqrt{n_1},...,\sqrt{n_k})^\top = a_i^\top\sqrt{n}a_1 = 0$, for $i = 2,...,k$. Hence $\boldsymbol{W} = \boldsymbol{A}\boldsymbol{V} \sim \mathcal{N}(\mu(\sqrt{n},0,...,0)^\top, \sigma^2\boldsymbol{I}_k)$. By the definition of χ^2 distribution, $\sum_{i=2}^k W_i^2 \sim \sigma^2\chi^2_{k-1}$ and $\sum_{i=1}^k W_i^2 = \boldsymbol{W}^\top\boldsymbol{W} = (\boldsymbol{A}\boldsymbol{V})^\top\boldsymbol{A}\boldsymbol{V} = \boldsymbol{V}^\top\boldsymbol{V} = \sum_{i=1}^k V_i^2$. Then, $\sum_{i=1}^k n_i(\hat{U}_i-\hat{U})^2 = \sum_{i=1}^k n_i\hat{U}_i^2 - n\hat{U}^2 = \sum_{i=1}^k V_i^2 - (a_1^\top\boldsymbol{V})^2 = \sum_{i=2}^k W_i^2 \sim \sigma^2\chi^2_{k-1}$, which completes the proof.

D. Lemma 4 and Its Proof

Lemma 4. *For large enough n, if $(\hat{\mathcal{T}},\hat{\mathcal{C}}) \neq (\mathcal{T}^0,\mathcal{C}^0)$, then the difference Δ between the penalty terms in MDL$(\hat{\mathcal{T}},\hat{\mathcal{C}})$ and that in MDL$(\mathcal{T}^0,\mathcal{C}^0)$ is positive and of order $O(\log n)$.*

Proof: Let $\mathcal{B}$ be a probability 1 set. For each $\omega \in \mathcal{B}$, suppose on the contrary $\hat{\mathcal{T}} \nrightarrow \mathcal{T}^0$ or $\hat{\mathcal{C}} \nrightarrow \mathcal{C}^0$. For large enough n, The penalty term of the MDL for the fitted model is

$$\log(M^*+1)+\sum_{m=1}^{M^*}\log(t^*_m-t^*_{m-1})+\sum_{m=1}^{M^*+1}(p+1)\log(d^{*(m)}) + \sum_{m=1}^{M^*+1}\sum_{r=1}^{d^{*(m)}}\frac{1}{2}\log(\sum_{t=t^*_{m-1}}^{t^*_m-1}\sum_{i,c_i^{*(m)}=r} n_{t,i}), \tag{32}$$

and the penalty term of the MDL for the true model is

$$\log(M^0+1)+\sum_{m=1}^{M^0}\log(t^0_m-t^0_{m-1})+\sum_{m=1}^{M^0+1}(p+1)\log(d^{0(m)}) + \sum_{m=1}^{M^0+1}\sum_{r=1}^{d^{0(m)}}\frac{1}{2}\log(\sum_{t=t^0_{m-1}}^{t^0_m-1}\sum_{i,c_i^{0(m)}=r} n_{t,i}). \tag{33}$$

Define Δ as the difference between (32) and (33).

As $M^0 \le M^* \le T$, $d^{0(m)} \le p,\ \forall m$ and $d^{*(m)} \le p,\ \forall m$, the first part of Δ

$$\begin{aligned}&\Big[\log(M^*+1)+\sum_{m=1}^{M^*}\log(t^*_m-t^*_{m-1}) \\ &+\sum_{m=1}^{M^*+1}(p+1)\log(d^{*(m)})\Big]-\Big[\log(M^0+1) \\ &+\sum_{m=1}^{M^0}\log(t^0_m-t^0_{m-1})+\sum_{m=1}^{M^0+1}(p+1)\log(d^{0(m)})\Big]\end{aligned} \tag{34}$$

is finite.

By Lemma 2 with large enough n, for each of the fitted $\hat{R}(m,r)=R^*(m,r)$, there must exist a true $R^0(s,l)$, such that $R^*(m,r)\subseteq R^0(s,l)$. Without loss of generality, we assume that there exists a true set $R^0(s,l)=\cup_{(m,r)\in S}R^*(m,r)$, which means that this set is over segmented. And for all the other true sets, we have $R^0(s',l')=R^*(m',r')$; that is, the fitted model is the same as the true model in all the other sets.

Therefore, the second part of Δ can be written in the following format:

$$\begin{aligned}&\sum_{m=1}^{M^*+1}\sum_{r=1}^{d^{*(m)}}\frac{1}{2}\log(\sum_{t=t^*_{m-1}}^{t^*_m-1}\sum_{i,c_i^{*(m)}=r} n_{t,i}) \\ &-\sum_{m=1}^{M^0+1}\sum_{r=1}^{d^{0(m)}}\frac{1}{2}\log(\sum_{t=t^0_{m-1}}^{t^0_m-1}\sum_{i,c_i^{0(m)}=r} n_{t,i}) \\ =&\sum_{(m,r)\in S}\frac{1}{2}\log(\sum_{t=t^*_{m-1}}^{t^*_m-1}\sum_{i,c_i^{*(m)}=r} n_{t,i}) \\ &-\frac{1}{2}\log(\sum_{t=t^0_{s-1}}^{t^0_s-1}\sum_{i,c_i^{0(s)}=l} n_{t,i}).\end{aligned} \tag{35}$$

Here we have

$$\sum_{(m,r)\in S}\sum_{t=t^*_{m-1}}^{t^*_m-1}\sum_{i,c_i^{*(m)}=r} n_{t,i}=\sum_{t=t^0_{s-1}}^{t^0_s-1}\sum_{i,c_i^{0(s)}=l} n_{t,i}. \tag{36}$$

As n is large enough, combining (36) with the assumption (21), it can be seen that the second part of Δ defined by (35) is positive and of order $O(\log(n))$. As in Δ, the other part (34) is finite, the second part dominates Δ, which finishes the proof.

E. Proof of Theorem 1

By Lemma 4, $\frac{1}{n}\Delta$ is positive and of order $O(\log(n)/n)$. The difference between the negative log-likelihood terms in $\frac{1}{n}\text{MDL}(\mathcal{T}^0,\mathcal{C}^0)-\frac{1}{n}\text{MDL}(\mathcal{T}^*,\mathcal{C}^*)$ is

$$\frac{1}{2}\log(\frac{1}{n}\sum_{m=1}^{M^0+1}\sum_{r=1}^{d^{0(m)}}\text{SSE}_r^{(m)})-\frac{1}{2}\log(\frac{1}{n}\sum_{m=1}^{M^*+1}\sum_{r=1}^{d^{*(m)}}\text{SSE}_r^{*(m)}).$$

By Lemma 2, this difference is positive. To prove the theorem, it is sufficient to show that the difference is of order $o(\log(n)/n)$. We begin with calculating

$$\begin{aligned}&\frac{1}{2}\log(\frac{1}{n}\sum_{m=1}^{M^0+1}\sum_{r=1}^{d^{0(m)}}\text{SSE}_r^{(m)})-\frac{1}{2}\log(\frac{1}{n}\sum_{m=1}^{M^*+1}\sum_{r=1}^{d^{*(m)}}\text{SSE}_r^{*(m)}) \\ =&\frac{1}{2}\log(\frac{\sum_{m=1}^{M^0+1}\sum_{r=1}^{d^{0(m)}}\text{SSE}_r^{(m)}}{\sum_{m=1}^{M^*+1}\sum_{r=1}^{d^{*(m)}}\text{SSE}_r^{*(m)}}) \\ =&\frac{1}{2}\log(1+ \\ &\frac{\sum_{m=1}^{M^0+1}\sum_{r=1}^{d^{0(m)}}\text{SSE}_r^{(m)}-\sum_{m=1}^{M^*+1}\sum_{r=1}^{d^{*(m)}}\text{SSE}_r^{*(m)}}{\sum_{m=1}^{M^*+1}\sum_{r=1}^{d^{*(m)}}\text{SSE}_r^{*(m)}}) \\ \le&\frac{1}{2}\frac{\sum_{m=1}^{M^0+1}\sum_{r=1}^{d^{0(m)}}\text{SSE}_r^{(m)}-\sum_{m=1}^{M^*+1}\sum_{r=1}^{d^{*(m)}}\text{SSE}_r^{*(m)}}{\sum_{m=1}^{M^*+1}\sum_{r=1}^{d^{*(m)}}\text{SSE}_r^{*(m)}}.\end{aligned} \tag{37}$$

Without loss of generality, we use the same idea in the proof of Lemma 4. Let

$$\begin{aligned}\text{SSE}^0_{s,l}&=\sum_{t=t^0_{s-1}}^{t^0_s-1}\sum_{i,c_i^0=l}\sum_{j=1}^{n_{t,i}}(x_{t,i,j}-\hat{\beta}(R^0(s,l)))^2, \\ \text{SSE}^*_{m,r}&=\sum_{t=t^*_{m-1}}^{t^*_m-1}\sum_{i,c_i^*=r}\sum_{j=1}^{n_{t,i}}(x_{t,i,j}-\hat{\beta}(R^*(m,r)))^2,\end{aligned}$$

where

$$\begin{aligned}\hat{\beta}(R^0(s,l))&=\frac{\sum_{t=t^0_{s-1}}^{t^0_s-1}\sum_{i,c_i^0=l}\sum_{j=1}^{n_{t,i}}x_{t,i,j}}{\sum_{t=t^0_{s-1}}^{t^0_s-1}\sum_{i,c_i^0=l}n_{t,i}}, \\ \hat{\beta}(R^*(m,r))&=\frac{\sum_{t=t^*_{m-1}}^{t^*_m-1}\sum_{i,c_i^*=r}\sum_{j=1}^{n_{t,i}}x_{t,i,j}}{\sum_{t=t^*_{m-1}}^{t^*_m-1}\sum_{i,c_i^*=r}n_{t,i}}.\end{aligned} \tag{38}$$

Then the numerator of (37) can be written as

$$\begin{aligned}
&\text{SSE}^0_{s,l} - \sum_{(m,r)\in S} \text{SSE}^*_{m,r} \\
&= \sum_{(m,r)\in S} \sum_{t=t^*_{m-1}}^{t^*_m-1} \sum_{i,c^*_i=r} \sum_{j=1}^{n_{t,i}} (x_{t,i,j} - \hat{\beta}(R^0(s,l)))^2 \\
&\quad - \sum_{(m,r)\in S} \text{SSE}^*_{m,r} \\
&= \sum_{(m,r)\in S} \sum_{t=t^*_{m-1}}^{t^*_m-1} \sum_{i,c^*_i=r} \sum_{j=1}^{n_{t,i}} (x_{t,i,j} - \hat{\beta}(R^*(m,r)) + \\
&\quad \hat{\beta}(R^*(m,r)) - \hat{\beta}(R^0(s,l)))^2 - \sum_{(m,r)\in S} \text{SSE}^*_{m,r} \\
&= \sum_{(m,r)\in S} \sum_{t=t^*_{m-1}}^{t^*_m-1} \sum_{i,c^*_i=r} \sum_{j=1}^{n_{t,i}} (x_{t,i,j} - \hat{\beta}(R^*(m,r)))^2 \\
&\quad + 2 \sum_{(m,r)\in S} \sum_{t=t^*_{m-1}}^{t^*_m-1} \sum_{i,c^*_i=r} \sum_{j=1}^{n_{t,i}} [(x_{t,i,j} - \hat{\beta}(R^*(m,r))) \\
&\quad (\hat{\beta}(R^*(m,r)) - \hat{\beta}(R^0(s,l)))] \\
&\quad + \sum_{(m,r)\in S} \sum_{t=t^*_{m-1}}^{t^*_m-1} \sum_{i,c^*_i=r} \sum_{j=1}^{n_{t,i}} (\hat{\beta}(R^*(m,r)) - \hat{\beta}(R^0(s,l)))^2 \\
&\quad - \sum_{(m,r)\in S} \text{SSE}^*_{m,r} \\
&= \sum_{(m,r)\in S} \text{SSE}^*_{m,r} + 0 \\
&\quad + \sum_{(m,r)\in S} \sum_{t=t^*_{m-1}}^{t^*_m-1} \sum_{i,c^*_i=r} \sum_{j=1}^{n_{t,i}} (\hat{\beta}(R^*(m,r)) - \hat{\beta}(R^0(s,l)))^2 \\
&\quad - \sum_{(m,r)\in S} \text{SSE}^*_{m,r} \\
&= \sum_{(m,r)\in S} \sum_{t=t^*_{m-1}}^{t^*_m-1} \sum_{i,c^*_i=r} \sum_{j=1}^{n_{t,i}} (\hat{\beta}(R^*(m,r)) - \hat{\beta}(R^0(s,l)))^2 \\
&= \sum_{(m,r)\in S} \Big(\sum_{t=t^*_{m-1}}^{t^*_m-1} \sum_{i,c^*_i=r} n_{t,i} \Big) (\hat{\beta}(R^*(m,r)) - \hat{\beta}(R^0(s,l)))^2.
\end{aligned} \tag{39}$$

By (38), we have

$$\hat{\beta}(R^*(m,r)) \sim \mathcal{N}(\beta^{0(s)}_{(l)}, \frac{\sigma^2}{\sum_{t=t^*_{m-1}}^{t^*_m-1} \sum_{i,c^*_i=r} n_{t,i}}) \tag{40}$$

are independent for different $(m,r) \in S$. Also

$$\hat{\beta}(R^0(s,l)) = \frac{\sum_{(m,r)\in S} (\sum_{t=t^*_{m-1}}^{t^*_m-1} \sum_{i,c^*_i=r} n_{t,i}) \hat{\beta}(R^*(m,r))}{\sum_{(m,r)\in S} (\sum_{t=t^*_{m-1}}^{t^*_m-1} \sum_{i,c^*_i=r} n_{t,i})}. \tag{41}$$

By (40), (41) and Lemma 3, we have

$$\begin{aligned}
&\sum_{(m,r)\in S} \Big(\sum_{t=t^*_{m-1}}^{t^*_m-1} \sum_{i,c^*_i=r} n_{t,i} \Big) (\hat{\beta}(R^*(m,r)) - \hat{\beta}(R^0(s,l)))^2 \\
&\sim \sigma^2 \chi^2_{|S|-1}.
\end{aligned}$$

As $|S| \leq Tp$, we can conclude that (39) is of order $O(1)$.

In addition, the denominator of (37), $\sum_{m=1}^{M^*+1} \sum_{r=1}^{d^{*(m)}} \text{SSE}^{*(m)}_r$, satisfies

$$\frac{1}{n} \sum_{m=1}^{M^*+1} \sum_{r=1}^{d^{*(m)}} \text{SSE}^{*(m)}_r \to \sigma^2 \quad a.s.$$

Furthermore, we can show that the numerator and the denominator of (37) are independent. That is to say,

$$\begin{aligned}
0 <& \frac{1}{2} \log\Big(\frac{1}{n} \sum_{m=1}^{M^0+1} \sum_{r=1}^{d^{0(m)}} \text{SSE}^{(m)}_r\Big) - \frac{1}{2} \log\Big(\frac{1}{n} \sum_{m=1}^{M^*+1} \sum_{r=1}^{d^{*(m)}} \text{SSE}^{*(m)}_r\Big) \\
=& o(\log(n)/n).
\end{aligned} \tag{42}$$

Then for large enough n, combining Lemma 4 and (42) gives

$$\begin{aligned}
&\frac{1}{n}\text{MDL}(\mathcal{T}^0, \mathcal{C}^0) - \frac{1}{n}\text{MDL}(\hat{\mathcal{T}}, \hat{\mathcal{C}}) \\
=& -\frac{1}{n}\Delta + \frac{1}{2} \log\Big(\frac{1}{n} \sum_{m=1}^{M^0+1} \sum_{r=1}^{d^{0(m)}} \text{SSE}^{(m)}_r\Big) \\
&- \frac{1}{2} \log\Big(\frac{1}{n} \sum_{m=1}^{M^*+1} \sum_{r=1}^{d^{*(m)}} \text{SSE}^{*(m)}_r\Big) < 0,
\end{aligned}$$

which is a contradiction. This finishes the proof.

F. An Alternative Expression for $\Omega_1(\cdot)$ in (22)

Let $\boldsymbol{\alpha}_{(1)} = (\boldsymbol{\alpha}_1^\top, \boldsymbol{\alpha}_2^\top, ..., \boldsymbol{\alpha}_{T-1}^\top)^\top$ and $\mathcal{Q}_1 = \{\boldsymbol{\alpha}_{(1)} | \|\boldsymbol{\alpha}_t\|_2 \leq 1, t = 1, ..., T-1\}$. Notice that for any vector $\boldsymbol{v}$, $\|\boldsymbol{v}\|_2 = \max_{\|\boldsymbol{\alpha}\|_2 \leq 1} \alpha^\top \boldsymbol{v}$, where $\boldsymbol{\alpha}$ is a vector that has the same dimension as $\boldsymbol{v}$. Then $\Omega_1(\boldsymbol{\beta})$ can be written as

$$\begin{aligned}
\Omega_1(\boldsymbol{\beta}) &= \lambda_1 \sum_{t=1}^{T-1} \|\boldsymbol{\beta}_{t+1} - \boldsymbol{\beta}_t\|_2 = \lambda \sum_{t=1}^{T-1} \max_{\|\boldsymbol{\alpha}_t\|_2 \leq 1} \boldsymbol{\alpha}_t^\top (\boldsymbol{\beta}_{t+1} - \boldsymbol{\beta}_t) \\
&= \lambda \max_{\boldsymbol{\alpha}_{(1)} \in \mathcal{Q}_1} \sum_{t=1}^{T-1} \boldsymbol{\alpha}_t^\top (\boldsymbol{\beta}_{t+1} - \boldsymbol{\beta}_t) = \max_{\boldsymbol{\alpha}_{(1)} \in \mathcal{Q}_1} \alpha_{(1)}^\top \boldsymbol{C}_1 \boldsymbol{\beta},
\end{aligned}$$

where the matrix $\boldsymbol{C}_1 \in \mathcal{R}^{(T-1)p \times Tp}$ is defined as

$$\boldsymbol{C}_1 = \lambda_1 \begin{pmatrix} -\boldsymbol{I} & \boldsymbol{I} & & \\ & -\boldsymbol{I} & \boldsymbol{I} & \\ & & \ddots & \ddots \\ & & & -\boldsymbol{I} \quad \boldsymbol{I} \end{pmatrix} \tag{43}$$

with $\boldsymbol{I} = \boldsymbol{I}_p$ being the p-dimensional identity matrix.

G. An Alternative Expression for $\Omega_2(\cdot)$ in (23)

Let $\boldsymbol{\alpha}_{(2)} \in \mathcal{R}^{T|E|}$ and $\mathcal{Q}_2 = \{\boldsymbol{\alpha} | \|\boldsymbol{\alpha}\|_\infty \leq 1\}$, and notice that $\|\boldsymbol{v}\|_1 = \max_{\|\boldsymbol{\alpha}\|_\infty \leq 1} \boldsymbol{\alpha}^\top \boldsymbol{v}$. Then $\Omega_2(\boldsymbol{\beta})$ can be written as

$$\Omega_2(\boldsymbol{\beta}) = \lambda_2 \sum_{t=1}^{T} \|\boldsymbol{G}\boldsymbol{\beta}_t\|_1 = \|C_2\boldsymbol{\beta}\|_1 = \max_{\alpha_{(2)} \in \mathcal{Q}_2} \alpha_{(2)}^\top \boldsymbol{C}_2 \boldsymbol{\beta},$$

where

$$\boldsymbol{C}_2 = \lambda_2 \begin{pmatrix} \boldsymbol{G} & & \\ & \ddots & \\ & & \boldsymbol{G} \end{pmatrix}. \tag{44}$$

H. A Smooth Approximation of $\Omega(\cdot) = \Omega_1(\cdot) + \Omega_2(\cdot)$

Let $\boldsymbol{\alpha} = (\boldsymbol{\alpha}_{(1)}^\top, \boldsymbol{\alpha}_{(2)}^\top)^\top$ and

$$\boldsymbol{C} = (\boldsymbol{C}_1^\top, \boldsymbol{C}_2^\top)^\top. \tag{45}$$

The penalty term $\Omega(\boldsymbol{\beta})$ can be written as

$$\Omega(\boldsymbol{\beta}) = \max_{\boldsymbol{\alpha}_{(1)} \in \mathcal{Q}_1} \boldsymbol{\alpha}_{(1)}^\top \boldsymbol{C}_1 \boldsymbol{\beta} + \max_{\boldsymbol{\alpha}_{(2)} \in \mathcal{Q}_2} \boldsymbol{\alpha}_{(2)}^\top \boldsymbol{C}_2 \boldsymbol{\beta} = \max_{\boldsymbol{\alpha} \in \mathcal{Q}} \boldsymbol{\alpha}^\top \boldsymbol{C} \boldsymbol{\beta},$$

where $\mathcal{Q} = \{\boldsymbol{\alpha} = (\boldsymbol{\alpha}_{(1)}^\top, \boldsymbol{\alpha}_{(2)}^\top)^\top | \boldsymbol{\alpha}_{(1)} \in \mathcal{Q}_1 \text{ and } \boldsymbol{\alpha}_{(2)} \in \mathcal{Q}_2\}$.

By [33], The smooth approximation of $\Omega(\boldsymbol{\beta})$ can be constructed as $g_\mu(\boldsymbol{\beta}) = \max_{\boldsymbol{\alpha} \in \mathcal{Q}} (\boldsymbol{\alpha}^\top \boldsymbol{C} \boldsymbol{\beta} - \mu d(\boldsymbol{\alpha}))$, where μ is a positive smoothness parameter and $d(\boldsymbol{\alpha}) = \frac{1}{2}\|\boldsymbol{\alpha}\|_2^2$. Therefore, the original penalty term $\Omega(\boldsymbol{\beta})$ can be viewed as $g_0(\boldsymbol{\beta})$.

Let $D = \max_{\alpha \in \mathcal{Q}} d(\boldsymbol{\alpha})$, then by [33], $g_0(\boldsymbol{\beta}) - \mu D \leq g_\mu(\boldsymbol{\beta}) \leq g_0(\boldsymbol{\beta})$, which means that $g_\mu(\boldsymbol{\beta})$ is an approximation of $g_0(\boldsymbol{\beta})$ with a maximum gap of μD. [23] suggested that $\mu = \frac{\varepsilon}{2D}$ achieves the best convergence rate for the given desired accuracy ε. For the current problem

$$\begin{aligned} D &= \max_{\alpha \in \mathcal{Q}} d(\boldsymbol{\alpha}) = \max_{\boldsymbol{\alpha}_{(1)} \in \mathcal{Q}_1} \frac{1}{2}\|\boldsymbol{\alpha}_{(1)}\|_2^2 + \max_{\boldsymbol{\alpha}_{(2)} \in \mathcal{Q}_2} \frac{1}{2}\|\boldsymbol{\alpha}_{(2)}\|_2^2 \\ &= \frac{1}{2}(T-1) + \frac{1}{2}T|\boldsymbol{E}|. \end{aligned} \tag{46}$$

Also, by Theorem 1 in [23], for $\mu > 0$, $g_\mu(\boldsymbol{\beta})$ is convex and continuously-differentiable with respect to $\boldsymbol{\beta}$, with gradient

$$\nabla g_\mu(\boldsymbol{\beta}) = \boldsymbol{C}^\top \boldsymbol{\alpha}^*,$$

where $\boldsymbol{\alpha}^* = \arg\max_{\alpha \in \mathcal{Q}} \alpha^\top \boldsymbol{C}\boldsymbol{\beta} - \mu d(\boldsymbol{\alpha})$. Here $\nabla g_\mu(\boldsymbol{\beta})$ is Lipschitz continuous with Lipschitz constant $L_\mu = \frac{1}{\mu}\|\boldsymbol{C}\|^2$, where $\|.\|$ is the matrix spectral norm. ($\|\boldsymbol{C}\| \equiv \max_{\|\boldsymbol{v}\|_2 \leq 1} \|\boldsymbol{C}\boldsymbol{v}\|_2$).

As $\boldsymbol{\alpha}^* = ((\boldsymbol{\alpha}_{(1)}^*)^\top, (\boldsymbol{\alpha}_{(2)}^*)^\top)^\top$, by [23], we have

$$\begin{aligned} \boldsymbol{\alpha}_{(1)}^* &= (\boldsymbol{\alpha}_{(1),1}^*, ..., \boldsymbol{\alpha}_{(1),(T-1)}^*)^\top \\ \boldsymbol{\alpha}_{(1),t}^* &= S_1(\frac{\lambda_1}{\mu}(\boldsymbol{\beta}_{t+1} - \boldsymbol{\beta}_t)), \quad t = 1, \ldots, T-1, \end{aligned} \tag{47}$$

where S_1 is the projection operator that projects a vector onto l_2 unit ball:

$$S_1(\boldsymbol{u}) = \begin{cases} \frac{\boldsymbol{u}}{\|\boldsymbol{u}\|_2} & \|\boldsymbol{u}\|_2 \geq 1, \\ \boldsymbol{u} & \|\boldsymbol{u}\|_2 < 1. \end{cases}$$

In addition,

$$\begin{aligned} \boldsymbol{\alpha}_{(2)}^* &= (\boldsymbol{\alpha}_{(2),1}^*, ..., \boldsymbol{\alpha}_{(2),T}^*)^\top \\ \boldsymbol{\alpha}_{(2),t}^* &= S_2(\frac{\lambda_2}{\mu} G\boldsymbol{\beta}_t), \quad t = 1, \ldots, T, \end{aligned} \tag{48}$$

where S_2 is the projection operator defined as

$$S_2(x) = \begin{cases} x & x \in [-1, 1] \\ -1 & x < -1 \\ 1 & x > 1. \end{cases}$$

And for any vector $\boldsymbol{u}$, the projection $S_2(\boldsymbol{u})$ is defined as applying S_2 element-wise. So the operator can be viewed as the projection operator that projects a vector onto l_∞ unit ball.

I. Smoothing Proximal Gradient Descent

By replacing the penalty term $\Omega(\boldsymbol{\beta})$ with $g_\mu(\boldsymbol{\beta})$, we obtain the following optimization problem

$$\min_{\boldsymbol{\beta}} h(\boldsymbol{\beta}) \equiv l(\boldsymbol{\beta}|X, \boldsymbol{n}) + g_\mu(\boldsymbol{\beta}).$$

The gradient of $h(\boldsymbol{\beta})$ is $\nabla h(\boldsymbol{\beta}) = \boldsymbol{n}(\boldsymbol{\beta} - \boldsymbol{X}) + \boldsymbol{C}^\top \boldsymbol{\alpha}^*$, which is Lipschitz continuous with the Lipschitz constant

$$L = n_{\max} + L_\mu = n_{\max} + \frac{1}{\mu}\|\boldsymbol{C}\|^2, \tag{49}$$

where $n_{\max}$ is the largest element of vector $\boldsymbol{n}$.

J. Computation of the Lipschiz Constant

To use the smoothing proximal gradient descent algorithm, one needs to compute the Lipschiz constant L (49). However, it is difficulty to calculate the spectral norm $\|\boldsymbol{C}\|$ when the dimension of $\boldsymbol{C}$ is high. Therefore, following [23], we replace it with an upper bound. We begin by calculating

$$\begin{aligned} \|\boldsymbol{C}\|^2 &= \left\| \begin{pmatrix} \boldsymbol{C}_1 \\ \boldsymbol{C}_2 \end{pmatrix} \right\|^2 = \max_{\|\boldsymbol{v}\|_2 \leq 1} \left\| \begin{pmatrix} \boldsymbol{C}_1\boldsymbol{v} \\ \boldsymbol{C}_2\boldsymbol{v} \end{pmatrix} \right\|_2^2 \\ &= \max_{\|\boldsymbol{v}\|_2 \leq 1} \|\boldsymbol{C}_1\boldsymbol{v}\|_2^2 + \|\boldsymbol{C}_2\boldsymbol{v}\|_2^2 \\ &\leq \max_{\|\boldsymbol{v}\|_2 \leq 1} \|\boldsymbol{C}_1\boldsymbol{v}\|_2^2 + \max_{\|\boldsymbol{v}\|_2 \leq 1} \|\boldsymbol{C}_2\boldsymbol{v}\|_2^2. \end{aligned}$$

Let $\boldsymbol{v} = (\boldsymbol{v}_1^\top, \boldsymbol{v}_2^\top, \ldots, \boldsymbol{v}_T^\top)^\top$, where $\boldsymbol{v}_t = (v_{t,1}, v_{t,2}, ..., v_{t,p})^\top$, $t = 1, ..., T$. Now calculate

$$\begin{aligned} \|\boldsymbol{C}_1\boldsymbol{v}\|_2^2 &= \lambda_1^2 \sum_{t=1}^{T-1} \|\boldsymbol{v}_{t+1} - \boldsymbol{v}_t\|_2^2 \\ &= \lambda_1^2 \sum_{t=1}^{T-1} (\|\boldsymbol{v}_{t+1}\|_2^2 - 2\boldsymbol{v}_{t+1} \cdot \boldsymbol{v}_t + \|\boldsymbol{v}_t\|_2^2) \\ &\leq \lambda_1^2 \sum_{t=1}^{T-1} 2(\|\boldsymbol{v}_{t+1}\|_2^2 + \|\boldsymbol{v}_t\|_2^2) \\ &\leq \lambda_1^2 \sum_{t=1}^{T} 4\|\boldsymbol{v}_t\|_2^2 = 4\lambda_1^2 \|\boldsymbol{v}\|_2^2. \end{aligned}$$

Therefore, $\max_{\|\boldsymbol{v}\|_2 \leq 1} \|\boldsymbol{C}_1\boldsymbol{v}\|_2^2 \leq 4\lambda_1^2$. Next calculate

$$\|\boldsymbol{C}_2\boldsymbol{v}\|_2^2 = \lambda_2^2 \sum_{t=1}^{T} \|\boldsymbol{G}\boldsymbol{v}_t\|_2^2 \leq \lambda_2^2 \sum_{t=1}^{T} d_1^2 \|\boldsymbol{v}_t\|_2^2 = \lambda_2^2 d_1^2 \|\boldsymbol{v}\|_2^2,$$

where d_1 is the largest (non-negative) singular value of $\boldsymbol{G}$, or $d_1 = \|\boldsymbol{G}\|$. So $\max_{\|\boldsymbol{v}\|_2 \leq 1} \|\boldsymbol{C}_2\boldsymbol{v}\|_2^2 = \lambda_2^2 d_1^2$.

Combining the above, we have

$$L = n_{\max} + \frac{1}{\mu}\|\boldsymbol{C}\|^2 \leq n_{\max} + \frac{1}{\mu}(4\lambda_1^2 + \lambda_2^2 \|\boldsymbol{G}\|^2). \tag{50}$$

K. Processing Output from Algorithm 1

As mentioned in Section III-B, the output from Algorithm 1 does not produce exactly the same signal values $\beta_{t,i}$'s for nodes belonging to the same time interval and cluster. To circumvent this issue, we apply Algorithm 2 to the output from Algorithm 1. Briefly, Algorithm 2 compares the fitted signal values (from Algorithm 1) between any two time points with a pre-set threshold to determine if a change point exists, and if yes, sets all the relevant fitted signal values to the same value. It employs Algorithm 3 recursively to compare connected nodes, in a depth-first manner. Nodes with very similar fitted signal values are assigned to the same cluster.

Algorithm 2 To convert output from Algorithm 1 into a final fitted model

Require: fitted coefficients $\tilde{\boldsymbol{\beta}}$, threshold ϵ, edges of the graph $\boldsymbol{E}$, tolerance γ

$\hat{\mathcal{T}} \leftarrow \emptyset$, $\hat{\mathcal{C}} \leftarrow \emptyset$
$c_1 \leftarrow \gamma\sqrt{p(2\epsilon)^2}$
for $t = 1, 2, \ldots, T-1$ **do**
  if $\|\tilde{\boldsymbol{\beta}}_{t+1} - \tilde{\boldsymbol{\beta}}_t\| > c_1$ **then**
    Add t to $\hat{\mathcal{T}}$
  end if
end for
for $m = 1, \ldots, |\hat{\mathcal{T}}| + 1$ **do**
  $t_m \leftarrow m$th element in $\hat{\mathcal{T}}$, $(t_{|\hat{\mathcal{T}}|+1} \leftarrow T+1)$
  $t_{m-1} \leftarrow (m-1)$th element in $\hat{\mathcal{T}}$, $(t_0 \leftarrow 1)$
  $c_2 \leftarrow \gamma\sqrt{(t_k - t_{k-1})(2\epsilon)^2}$
  $l \leftarrow (-1, -1, ..., -1) \in \mathcal{R}^p$
  $c \leftarrow 0$
  for $i = 1, \ldots, p$ **do**
    if $l_i = -1$ **then**
      Apply Algorithm 3 with i, $\tilde{\boldsymbol{\beta}}$, c_2, $\boldsymbol{E}$, l, c and (t_{m-1}, t_m)
      $c \leftarrow c + 1$
    end if
  end for
  Add l to $\hat{\mathcal{C}}$
end for
return fitted change points $\hat{\mathcal{T}}$, set of fitted membership vectors $\hat{\mathcal{C}}$

L. Justification for binomial log likelihood

Recall that in Section V-A the crime data set was modeled with a binomial distribution and we modified the MDL criterion by replacing the Gaussian likelihood with a binomial likelihood. Here we provide further details.

Suppose the observations are binomial counts; i.e. $y_{t,i} \sim Binomial(n_{t,i}, \beta_{t,i})$, where $n_{t,i}$ is known, $\beta_{t,i} \in [\epsilon_0, 1-\epsilon_0]$ for some positive ϵ_0, and $\beta_{t,i}$ satisfies both the temporally and spatially smoothness assumptions.

An MDL criterion for binomial data can be derived in the same manner as in Section II. Notice that $y_{t,i}$ can be viewed as the summation of $n_{t,i}$ iid Bernoulli trails $x_{t,i,j}$, and that the asymptotic properties hold when $n_{t,i}$ satisfies (20) and (21).

Algorithm 3 Use a depth-first search strategy to compare connected nodes, and nodes with similar fitted signal values to the coefficients are labelled the same.

Require: current index i, fitted coefficients $\tilde{\boldsymbol{\beta}}$, threshold c_2, edges of the graph $\boldsymbol{E}$, current membership vector l,, current label c, time interval (t_{m-1}, t_m)

$l_i \leftarrow c$
$\tilde{\boldsymbol{\beta}}_{(t_{m-1},t_m),i} \leftarrow (\tilde{\boldsymbol{\beta}}_{t_{m-1},i}, \tilde{\boldsymbol{\beta}}_{t_{m-1}+1,i}, ..., \tilde{\boldsymbol{\beta}}_{t_m-1,i})^\top$
$\tilde{\boldsymbol{\beta}}_{(t_{m-1},t_m),j} \leftarrow (\tilde{\boldsymbol{\beta}}_{t_{m-1},j}, \tilde{\boldsymbol{\beta}}_{t_{m-1}+1,j}, ..., \tilde{\boldsymbol{\beta}}_{t_m-1,j})^\top$
for $j = 1, \ldots, p$ **do**
  if $(i,j) \in \boldsymbol{E}$ **and** $l_i = -1$ **and** $\|\tilde{\boldsymbol{\beta}}_{(t_{m-1},t_m),i} - \tilde{\boldsymbol{\beta}}_{(t_{m-1},t_m),j}\|_2 < c_2$ **then**
    Apply Algorithm 3 with j, $\tilde{\boldsymbol{\beta}}$, c_2, $\boldsymbol{E}$, l, c, (t_{m-1}, t_m)
  end if
end for
return updated membership vector l

When $\{t, i\}$ belongs to the rth cluster in the mth time interval, the MLE of $\beta_{t,i}$ is

$$\hat{\beta}_{t,i} = \frac{\sum_{s=t_{m-1}}^{t_m-1} \sum_{q, c_q^{(m)}=r} y_{s,q}}{\sum_{s=t_{m-1}}^{t_m-1} \sum_{q, c_q^{(m)}=r} n_{s,q}}.$$

Ignoring constant terms, the negative log-likelihood is

$$-\sum_{t=1}^{T}\sum_{i=1}^{p} y_{t,i}\log(\beta_{t,i}) - \sum (n_{t,i} - y_{t,i})\log(1-\beta_{t,i}).$$

After plugging in the MLE for $\beta_{t,i}$, the code length of the residuals $\text{CL}(\hat{\mathcal{E}}|\mathcal{F})$ can be obtained, which leads to the following MDL criterion for binomial data

$$\begin{aligned}\text{MDL}(\mathcal{T}, \mathcal{C}) =& \log(M+1) + \sum_{m=1}^{M}\log(t_m - t_{m-1}) \\ &+ \sum_{m=1}^{M+1}(p+1)\log(d^{(m)}) \\ &+ \sum_{m=1}^{M+1}\sum_{r=1}^{d^{(m)}} \frac{1}{2}\log(\sum_{t=t_{m-1}}^{t_m-1}\sum_{i, c_i^{(m)}=r} n_{t,i}) \\ &- \sum_{t=1}^{T}\sum_{i=1}^{p} y_{t,i}\log(\hat{\beta}_{t,i}) \\ &- \sum_{t=1}^{T}\sum_{i=1}^{p}(n_{t,i} - y_{t,i})\log(1-\hat{\beta}_{t,i}).\end{aligned} \tag{51}$$

For simplicity, we re-express the above MDL criterion as

$$\begin{aligned}\text{MDL}(\mathcal{T}, \mathcal{C}) =& O(\log(n)) \\ &- \sum_{m=1}^{M+1}\sum_{r=1}^{d^{(m)}}[(\sum_{t=t_{m-1}}^{t_m-1}\sum_{i, c_i^{(m)}=r} n_{t,i}) \\ &\sigma(\frac{(\sum_{t=t_{m-1}}^{t_m-1}\sum_{i, c_i^{(m)}=r} y_{t,i})}{(\sum_{t=t_{m-1}}^{t_m-1}\sum_{i, c_i^{(m)}=r} n_{t,i})})],\end{aligned} \tag{52}$$

where $O(\log(n))$ is a term with order $\log(n)$ and $\sigma(x) = x\log(x) + (1-x)\log(1-x)$ for $x \in (0,1)$.

To establish the theoretical properties of (52), we will show that Lemma 1 and Theorem 1 are also true under the binomial setting. In addition, Lemma 2 and Lemma 4 also hold for (52) and their proofs are exactly the same.

Proof for Lemma 1:

We follow the same arguments as that in Appendix A.

$$\begin{aligned}&\frac{1}{n}\text{MDL}(\hat{\mathcal{T}},\hat{\mathcal{C}})\\ =&c_n-\sum_{m=1}^{M+1}\sum_{r=1}^{d^{(m)}}[(\frac{\sum_{t=t^*_{m-1}}^{t^*_m-1}\sum_{i,c_i^{*(m)}=r}n_{t,i}}{n})\sigma(\hat{\beta}(R^*(m,r)))]\end{aligned} \tag{53}$$

where

$$\hat{\beta}(R^*(m,r))=\frac{\sum_{t=t^*_{m-1}}^{t^*_m-1}\sum_{i,c_i^{*(m)}=r}y_{t,i}}{\sum_{t=t^*_{m-1}}^{t^*_m-1}\sum_{i,c_i^{*(m)}=r}n_{t,i}}.$$

Similarly we will discuss two cases for each $R^*(m,r)$.

If $R^*(m,r)\subseteq R^0(s,l)$, then by the strong law of large number, we have $\hat{\beta}(R^*(m,r))\to\beta_{(s)}^{0(l)}$. In addition, with (21) we have $\frac{1}{n}\sum_{t=t^*_{m-1}}^{t^*_m-1}\sum_{i,c_i^{*(m)}=r}n_{t,i}\to\sum_{t=t^*_{m-1}}^{t^*_m-1}\sum_{i,c_i^{*(m)}=r}\gamma_{t,i}$. Therefore,

$$\begin{aligned}&(\frac{\sum_{t=t^*_{m-1}}^{t^*_m-1}\sum_{i,c_i^{*(m)}=r}n_{t,i}}{n})\sigma(\hat{\beta}(R^*(m,r)))\\ \to&(\sum_{t=t^*_{m-1}}^{t^*_m-1}\sum_{i,c_i^{*(m)}=r}\gamma_{t,i})\sigma(\beta_{(s)}^{0(l)})\ a.s.\end{aligned} \tag{54}$$

On the other hand, if $R^*(m,r)\subseteq\cup_{(s,l)\in\boldsymbol{S}}R^0(s,l)$ and $R^*(m,r)\cap R^0(s,l)\neq\emptyset,\forall(s,l)\in\boldsymbol{S}$, we can show that

$$\begin{aligned}&\hat{\beta}(R^*(m,r))\to\\ &\frac{\sum_{(s,l)\in\boldsymbol{S}}\sum_{t=\max\{t^*_{m-1},t^0_{s-1}\}}^{\min\{t^*_m,t^0_s\}-1}\sum_{i,c_i^*=r,c_i^0=l}\gamma_{t,i}\beta_{(s)}^{0(l)}}{\sum_{(s,l)\in\boldsymbol{S}}\sum_{t=\max\{t^*_{m-1},t^0_{s-1}\}}^{\min\{t^*_m,t^0_s\}-1}\sum_{i,c_i^*=r,c_i^0=l}\gamma_{t,i}}\ a.s.\end{aligned}$$

Then

$$\begin{aligned}&(\frac{\sum_{t=t^*_{m-1}}^{t^*_m-1}\sum_{i,c_i^{*(m)}=r}n_{t,i}}{n})\sigma(\hat{\beta}(R^*(m,r)))\\ \to&(\sum_{(s,l)\in\boldsymbol{S}}\sum_{t=\max\{t^*_{m-1},t^0_{s-1}\}}^{\min\{t^*_m,t^0_s\}-1}\sum_{i,c_i^*=r,c_i^0=l}\gamma_{t,i})\\ &\sigma(\frac{\sum_{(s,l)\in\boldsymbol{S}}\sum_{t=\max\{t^*_{m-1},t^0_{s-1}\}}^{\min\{t^*_m,t^0_s\}-1}\sum_{i,c_i^*=r,c_i^0=l}\gamma_{t,i}\beta_{(s)}^{0(l)}}{\sum_{(s,l)\in\boldsymbol{S}}\sum_{t=\max\{t^*_{m-1},t^0_{s-1}\}}^{\min\{t^*_m,t^0_s\}-1}\sum_{i,c_i^*=r,c_i^0=l}\gamma_{t,i}})\ a.s.\\ \le&\sum_{(s,l)\in\boldsymbol{S}}(\sum_{t=\max\{t^*_{m-1},t^0_{s-1}\}}^{\min\{t^*_m,t^0_s\}-1}\sum_{i,c_i^*=r,c_i^0=l}\gamma_{t,i})\sigma(\beta_{(s)}^{0(l)}).\end{aligned} \tag{55}$$

The last inequality was obtained because of the convexity of $\sigma(.)$, together with Jensen's inequality

$$\sigma(\frac{\sum\alpha_ix_i}{\sum\alpha_i})\le\frac{\sum\alpha_i\sigma(x_i)}{\sum\alpha_i}$$

for any positive weights α_i.

Same as the argument in Appendix A, there would be a contradiction if Lemma 1 under the binomial setting does not hold. This finishes the proof.

Proof for Theorem 1:

By Lemma 2 with large enough n, without loss of generality, we assume that there is only one real cluster that is composed of multiple fitted clusters; i.e. $R^0(s,l)=\cup_{(m,r)\in\boldsymbol{S}}R^*(m,r)$. And all the other real clusters are not overfitted. Then the difference between the negative log-likelihood terms in $\frac{1}{n}\text{MDL}(\mathcal{T}^0,\mathcal{C}^0)-\frac{1}{n}\text{MDL}(\mathcal{T}^*,\mathcal{C}^*)$ is

$$\begin{aligned}&-\frac{1}{n}l(\hat{\beta}(R^0(s,l));\{y_{t,i}|(t,i)\in R^0(s,l)\})\\ &+\sum_{(m,r)\in\boldsymbol{S}}\frac{1}{n}l(\hat{\beta}(R^*(m,r));\{y_{t,i}|(t,i)\in R^*(m,r)\}),\end{aligned} \tag{56}$$

where

$$\begin{aligned}&l(\beta;\{y_{t,i}|(t,i)\in R(s,l)\})\\ =&\sum_{(t,i)\in\in R(s,l)}y_{t,i}\log(\beta)+(n_{t,i}-y_{t,i})\log(1-\beta)\end{aligned} \tag{57}$$

is the log-likelihood function.

By the strong law of large number we have $\hat{\beta}(R^*(m,r))\to\beta_{(s)}^{0(l)}\ a.s.\forall(m,r)\in\boldsymbol{S}$ and $\hat{\beta}(R^0(s,l))\to\beta_{(s)}^{0(l)}\ a.s.$ We also have

$$\begin{aligned}l'(\hat{\beta}(R^0(s,l));\{y_{t,i}|(t,i)\in R^0(s,l)\})=0\\ l'(\hat{\beta}(R^*(m,r));\{y_{t,i}|(t,i)\in R^*(m,r)\})=0\end{aligned} \tag{58}$$

as $\hat{\beta}(R^0(s,l))$ and $\hat{\beta}(R^*(m,r))$ are MLEs. The Taylor expansion of $l(\hat{\beta}(R^0(s,l);\{y_{t,i}|(t,i)\in R^0(s,l)\})$ around $\hat{\beta}(R^0(s,l))$ gives

$$\begin{aligned}&l(\hat{\beta}(R^0(s,l));\{y_{t,i}|(t,i)\in R^0(s,l)\})\\ =&l(\beta_{(s)}^{0(l)};\{y_{t,i}|(t,i)\in R^0(s,l)\})\\ &-\frac{1}{2}l''(\tilde{\beta}(R^0(s,l));\{y_{t,i}|(t,i)\in R^0(s,l)\})\\ &\times(\beta_{(s)}^{0(l)}-\hat{\beta}(R^0(s,l)))^2\end{aligned} \tag{59}$$

and

$$\begin{aligned}&l(\hat{\beta}(R^*(m,r)));\{y_{t,i}|(t,i)\in R^*(m,r))\})\\ =&l(\beta_{(s)}^{0(l)};\{y_{t,i}|(t,i)\in R^*(m,r))\})\\ &-\frac{1}{2}l''(\tilde{\beta}(R^*(m,r)));\{y_{t,i}|(t,i)\in R^*(m,r)\})\\ &\times(\beta_{(s)}^{0(l)}-\hat{\beta}(R^*(m,r)))^2,\end{aligned} \tag{60}$$

where $\tilde{\beta}(R^0(s,l))$ is between $\hat{\beta}(R^0(s,l))$ and $\beta_{(s)}^{0(l)}$, while $\tilde{\beta}(R^*(m,r))$ is between $\hat{\beta}(R^*(m,r))$ and $\beta_{(s)}^{0(l)}$.

Since $\hat{\beta}(R^*(m,r))-\beta_{(s)}^{0(l)}=O(\frac{1}{\sqrt{n}})$ and $\hat{\beta}(R^0(s,l))-\beta_{(s)}^{0(l)}=O(\frac{1}{\sqrt{n}})$ by the central limit theorem, and $\frac{1}{n}l''(\tilde{\beta}(R^0(s,l));\{y_{t,i}|(t,i)\in R^0(s,l)\})$ and $\frac{1}{n}l''(\tilde{\beta}(R^*(m,r)));\{y_{t,i}|(t,i)\in R^*(m,r)\})$ are stochastically bounded, (56) is of $o(\log(n)/n)$, which finishes the proof.

REFERENCES

[1] J. Rissanen, *Stochastic Complexity in Statistical Inquiry*. World Scientific, Singapore, 1989.

[2] ——, *Information and Complexity in Statistical Modeling*. Springer Science & Business Media, 2007.

[3] T. C. M. Lee, "Segmenting images corrupted by correlated noise," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, pp. 481–492, 1998.

[4] T. Roos, P. Myllymaki, and J. Rissanen, "MDL denoising revisited," *IEEE Transactions on Signal Processing*, vol. 57, pp. 3347–3360, 2009.

[5] D. F. Schmidt and E. Makalic, "The consistency of MDL for linear regression models with increasing signal-to-noise ratio," *IEEE Transactions on Signal Processing*, vol. 60, pp. 1508–1510, 2011.

[6] S. Kallummil and S. Kalyani, "High SNR consistent linear model order selection and subset selection," *IEEE Transactions on Signal Processing*, vol. 64, pp. 4307–4322, 2016.

[7] R. C. Y. Cheung, A. Aue, and T. C. M. Lee, "Consistent estimation for partition-wise regression and classification models," *IEEE Transactions on Signal Processing*, vol. 65, pp. 3662–3674, 2017.

[8] T. 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*, vol. 95, no. 449, pp. 259–270, 2000.

[9] A. Aue and T. C. M. Lee, "On image segmentation using information theoretic criteria," *The Annals of Statistics*, vol. 39, no. 6, p. 2912–2935, Dec 2011.

[10] R. C. Y. Cheung, A. Aue, S. Hwang, and T. 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*, vol. 6, pp. 580–591, 2020.

[11] J. Sharpnack, A. Singh, and A. Rinaldo, "Changepoint detection over graphs with the spectral scan statistic," in *International Conference on Artificial Intelligence and Statistics*, vol. 16, 2013, pp. 545–553.

[12] M. Kolar and E. P. Xing, "Estimating networks with jumps," *Electronic Journal of Statistics*, vol. 6, pp. 2069–2106, 2012.

[13] A. J. Gibberd and J. D. B. Nelson, "Regularized estimation of piecewise constant Gaussian graphical models: The group-fused graphical lasso," *Journal of Computational and Graphical Statistics*, vol. 26, no. 3, pp. 623–634, 2017.

[14] D. Hallac, Y. Park, S. Boyd, and J. Leskovec, "Network inference via the time-varying graphical lasso," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2017, p. 205–213.

[15] J. Yang and J. Peng, "Estimating time-varying graphical models," *Journal of Computational and Graphical Statistics*, vol. 29, no. 1, pp. 191–202, 2020.

[16] J. Flossdorf and C. Jentsch, "Change detection in dynamic networks using network characteristics," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 7, pp. 451–464, 2021.

[17] E. M. Farahani, R. B. Kazemzadeh, R. Noorossana, and G. Rahimian, "A statistical approach to social network monitoring," *Communications in Statistics - Theory and Methods*, vol. 46, no. 22, pp. 11 272–11 288, 2017.

[18] J. D. Wilson, N. T. Stevens, and W. H. Woodall, "Modeling and detecting change in temporal networks via the degree corrected stochastic block model," *Quality and Reliability Engineering International*, vol. 35, no. 5, pp. 1363–1378, 2019.

[19] R. Tibshirani, M. Saunders, S. Rosset, J. Zhu, and K. Knight, "Sparsity and smoothness via the fused lasso," *Journal of the Royal Statistical Society Series B*, vol. 67, pp. 91–108, 02 2005.

[20] K. Bleakley and J.-P. Vert, "The group fused lasso for multiple change-point detection," pp. arXiv:1106.4199 [q–bio.QM], 2011.

[21] X. Chen, S. Kim, Q. Lin, J. G. Carbonell, and E. P. Xing, "Graph-structured multi-task regression and an efficient optimization method for general fused lasso," p. arXiv:1005.3579 [stat.ML], 2010.

[22] S. Kim, K.-A. Sohn, and E. P. Xing, "A multivariate regression approach to association analysis of a quantitative trait network," *Bioinformatics*, vol. 25, no. 12, pp. i204–i212, 05 2009.

[23] X. Chen, Q. Lin, S. Kim, J. G. Carbonell, and E. P. Xing, "Smoothing proximal gradient method for general structured sparse regression," *The Annals of Applied Statistics*, vol. 6, no. 2, pp. 719–752, 2012.

[24] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM Journal on Imaging Sciences*, vol. 2, no. 1, pp. 183–202, 2009.

[25] C. Truong, L. Oudre, and N. Vayatis, "Selective review of offline change point detection methods," *Signal Processing*, vol. 167, p. 107299, 2020.

[26] L. Meier, S. V. D. Geer, and P. Bühlmann, "The group lasso for logistic regression," *Journal of the Royal Statistical Society Series B*, vol. 70, no. 1, pp. 53–71, February 2008.

[27] J. H. Friedman, T. Hastie, and R. Tibshirani, "Regularization paths for generalized linear models via coordinate descent," *Journal of Statistical Software*, vol. 33, no. 1, p. 1–22, 2010.

[28] Y.-X. Wang, J. Sharpnack, A. J. Smola, and R. J. Tibshirani, "Trend filtering on graphs," *Journal of Machine Learning Research*, vol. 17, no. 105, pp. 1–41, 2016.

[29] M. A. Taddy, "Autoregressive mixture models for dynamic spatial poisson processes: Application to tracking intensity of violent crime," *Journal of the American Statistical Association*, vol. 105, no. 492, pp. 1403–1417, 2010.

[30] M. Ranson, "Crime, weather, and climate change," *Journal of Environmental Economics and Management*, vol. 67, no. 3, pp. 274–302, 2014.

[31] C. Daly, R. P. Neilson, and D. L. Phillips, "A statistical-topographic model for mapping climatological precipitation over mountainous terrain," *Journal of Applied Meteorology*, vol. 33, pp. 140–158, 1994.

[32] M. Anderson, "Hydroclimate report water year 2015," *Office of the State Climatologist*, 2016.

[33] Y. Nesterov, "Smooth minimization of non-smooth functions," *Mathematical Programming*, vol. 103, pp. 127–152, 05 2005.

Cong Xu received the B.A. degree in Economics and the B.S. degree in Mathematics and Applied Mathematics in 2016 from Peking University, Beijing, China, and the Ph.D. in Statistics in 2021 from the University of California, Davis, USA. His research interests include change point detection, network analysis, and statistical applications in other disciplines.

Thomas C. M. Lee received the B.App.Sc. (Math) degree in 1992, and the B.Sc. (Hons) (Math) degree with University Medal in 1993, all from the University of Technology, Sydney, Australia. In 1997 he completed a Ph.D. degree jointly at Macquarie University and CSIRO Mathematical and Information Sciences, Sydney, Australia.

Currently, he is Professor of Statistics and Associate Dean of the Faculty in Mathematical and Physical Sciences at the University of California, Davis. He is an elected Fellow of the American Association for the Advancement of Science (AAAS), the American Statistical Association (ASA), and the Institute of Mathematical Statistics (IMS). From 2013 to 2015, he served as the Editor-in-Chief for the *Journal of Computational and Graphical Statistics*, and from 2015 to 2018, he served as the Chair of the Department of Statistics at UC Davis. His research interests include inference methods, image and signal processing, and statistical applications in other scientific disciplines.