Advances in Statistical Network Modeling and Nonlinear Time Series Modeling

Advances in Statistical Network Modeling and Nonlinear Time Series Modeling

A DISSERTATION PRESENTED
BY
QIUYI HAN
TO
THE DEPARTMENT OF STATISTICS

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY
IN THE SUBJECT OF
STATISTICS

HARVARD UNIVERSITY
CAMBRIDGE, MASSACHUSETTS

MAY 2018

Dissertation advisor: Professor Edoardo M. Airoldi Qiuyi Han

Advances in Statistical Network Modeling and Nonlinear Time Series Modeling

Abstract

The thesis is composed of two independent topics: statistical network modeling and nonlinear time series modeling. With the increasing demand of network data analysis, we present two statistical network models and inferences, analyzing both theoretical and real data performance. The first model targets static networks while the second model targets dynamic networks. The time series modeling focuses on sequential, nonlinear and high dimensional prediction problem which arises with the era of big data.

Contents

Dedicated to my family.

Acknowledgments

I owe my deepest gratitude to my advisor Prof. Edoardo M. Airoldi, who introduced me to my thesis topic and guided me through my six year long research journey. It has been my great pleasure and fortune to learn from his broad knowledge and insightful ideas. Without his constant support and encouragement, I would not have completed the thesis.

I thank Prof. Vahid Tarokh for advising me on the time series modeling. It has been my honor to work with him. I thank Prof. Jun S. Liu and Prof. Joseph K. Blitzstein for joining my committee and providing valuable inputs to my thesis.

I thank my dear friends and colleages Jie Ding, Xufei Wang, Hyungsuk Tak, Justin Yang, Alexander D'Amour and many others, who have helped me during my study.

Last but not least, I express my sincere gratitude to my parents for their unconditional love and support.

1 Introduction

The thesis consists of two independent topics: network modeling and time series modeling.

Network data arise in many applications, such as social science, neural science and etc. The statistical modeling and inference of network data has become a hot topic in the recent years[1, 2, 3, 4, 5, 6, 7]. In the first two chapters of thesis, two statistical models and inference methods are investigated. They are shown to have good theoretical properties. Their applications are demonstrated in real data examples.

In the last chapter, a method for sequential nonlinear time series prediction is proposed. Detailed introductions about each chapter are explained as follow.

1.1 Outline and Contributions

The outlines and contributions of each chapter are given below.

Contribution of Chapter 2: Exchangeable graph models (ExGM) are a nonparametric approach to modeling network data that subsumes a number of popular models. The key object that defines an ExGM is often referred to as a graphon, or graph kernel. Here, we make three contributions to advance the theory of estimation of graphons. We determine conditions under which a unique canonical representation for a graphon exists and it is identifiable. We propose a 3-step procedure to estimate the canonical graphon of any ExGM that satisfies these conditions. We then focus on a specific estimator, built using the proposed 3-step procedure, which combines probability matrix estimation by Universal Singular Value Thresholding (USVT) and empirical degree sorting of the observed adjacency matrix. We prove that this estimator is consistent. We illustrate how the proposed theory and methods can be used to develop hypothesis testing procedures for models of network data.

The paper [8] is a joint work with Justin Yang. I contributed to the development of theory and empirical data analysis.

Contribution of Chapter 3: Significant progress has been made recently on theoretical analysis of estimators for the stochastic block model (SBM). In this chapter, we consider the multi-graph SBM, which serves as a foundation for many application settings including dynamic and multi-layer networks. We explore the asymptotic properties of two estimators for the multi-graph SBM, namely spectral clustering and the maximum-likelihood estimate (MLE), as the number of layers of the multi-graph increases. We derive sufficient conditions for consistency of both estimators and propose a variational approximation to the MLE that is computationally feasible for large networks. We verify the sufficient conditions via simulation and demonstrate that they are practical. In addition, we apply the model to two real data sets: a dynamic social network and a multi-layer social network with several types of relations.

I am the main contributor of the paper [9] .

Contribution of Chapter 4: We propose a method for adaptive nonlinear sequential mod-

eling of time series data. Data are modeled as a nonlinear function of past values corrupted by noise, and the underlying nonlinear function is assumed to be approximately expandable in a spline basis. We cast the modeling of data as finding a good fit representation in the linear span of multidimensional spline basis, and use a variant of l_1-penalty regularization in order to reduce the dimensionality of representation. Using adaptive filtering techniques, we design our online algorithm to automatically tune the underlying parameters based on the minimization of the regularized sequential prediction error. We demonstrate the generality and flexibility of the proposed approach on both synthetic and real-world datasets. Moreover, we analytically investigate the performance of our algorithm by obtaining both bounds on prediction errors and consistency in variable selection.

The paper [10] is a joint work with Jie Ding. I contributed to the algorithm and data analysis.

2

Nonparametric estimation and testing of exchangeable graph models

Exchangeable graph models (ExGM) [11, 12, 13, 14] are nonparametric approaches to model network data which subsume a number of popular models, for example, stochastic block model (SBM). The key object that defines an ExGM is often referred to as a *graphon*, or *graph kernel*. Traditionally, the graphon estimation problem is often formulated as seeking the probability matrix, which generates the observed adjacency matrix according to independent Bernoulli trials. This line of research has been studied extensively recently [15, 16]. However, one of the deficiencies estimating the probability is that the estimated graphons always lack global structural information of the generating graphon.

We make three contributions to advance the theory of estimation of graphons. First,

we determine conditions under which a *unique* canonical representation for a graphon exists and is identifiable. Second, we propose a 3-step procedure to estimate the canonical graphon of any ExGM that satisfies these conditions. Third, we build a specific estimator using the proposed 3-step procedure, which combines probability matrix estimation by Universal Singular Value Thresholding (USVT) with empirical degree sorting of the observed adjacency matrix. We prove that the proposed estimator is consistent. We illustrate how the proposed theory and methods can be used to develop hypothesis testing procedures for models of network data.

Outline:

In Section 2.1, we discuss in details the identifiability issue to propose a functional form estimate for the generating graphon. In Section 2.2, we propose the 3-step procedure to construct estimates of a graphon. In Section 2.3, we focus on a specific choice of estimator and derive its asymptotic properties. In Section 2.4, we demonstrate the power of pursuing a functional form estimate in the context of classical hypothesis testing. We offer some remarks in Section 2.5.

2.1 Identifiability of exchangeable graph models

In this section, we first present the problem setup and define key notations. Then, we discuss the identifiability issues of graphons and conclude with a special but flexible subclass of ExGM, which we will focus on in the remainder of the paper.

2.1.1 Basic setup

Let $U_1, \ldots, U_N$ be i.i.d. uniform random variables on the closed interval $[0, 1]$, and let $W : [0, 1]^2 \to [0, 1]$ be an unknown symmetric measurable function. The observed data is an undirected simple graph described by an adjacent matrix A, which is an $N \times N$ symmetric

random matrix with binary elements such that, for $\mathcal{U}_N \triangleq \sigma(U_1, \ldots, U_N)$,

$$\begin{aligned} \mathcal{A}_{ii} &= 0, \text{ for all } i, \text{ and} \\ \mathcal{A}_{ij}|\mathcal{U}_N &\backsim \text{Ber}\left(W\left(U_i, U_j\right)\right), \text{ for } i < j, \end{aligned}$$

where $\mathcal{A}_{ij}$'s are, conditionally on $\mathcal{U}_N$, independent of each other for $i < j$. The unknown symmetric parameter matrix

$$\begin{aligned} P_{ii} &\triangleq 0, \text{ for all } i, \text{ and} \\ P_{ij} &\triangleq W\left(U_i, U_j\right), \text{ for } i < j \end{aligned}$$

is defined as the probability matrix. From this definition, we refer to the model specification on the observed undirected graph as the exchangeable graph model (ExGM). Call W as a graphon generating this ExGM, and $U_1, \ldots, U_N$ as the latent variables for the observed graph.

The estimation problem of interest is to draw inferences about the unknown graphon W from the observed adjacency matrix $\mathcal{A}$. In the literature, this problem is often formulate as follows:

Estimation problem 1 (P1). Build a matrix estimator $\hat{P}$ of P, independently of the latent variables.

(P1) is a popular problem formulation. However, it often leads to an estimator $\hat{P}$ that is unable to describe structural information encoded by the generating graphon W. This is undesirable for many practical situations, such as model similarity checking or prediction inferences.

Alternatively, we consider the following formulation.

Estimation problem 2 (P2). Build a nonparametric estimator $\hat{W}(u,v) \equiv W(\hat{u},\hat{v})$ of $W(u,v)$, as a plug-in estimator that relies on estimating latent variables.

However, there is an unavoidable well-posedness issues before we further study (P2). Due to the highly symmetric structure as a result of the exchangeability of ExGM, it is possible to have multiple graphons that generate the same ExGM. Therefore, (P2) cannot be well-posed unless there is a unique and identifiable representation among all graphons that generate the same underlying ExGM.

To study the identifiability issue, we say that two ExGMs $\mathbb{P}_1$ and $\mathbb{P}_2$ are the same if, for any binary and symmetric $N \times N$ matrix realization A and any $N \in \mathbb{N}$, $\mathbb{P}_1(A) = \mathbb{P}_2(A)$. We discuss this issue next.

2.1.2 Identifiability of graphons

The discussion of the identifiability of (P2) starts from a non-trivial statement, which might seem true at first glance. The statement is that for any measure preserving mapping $\phi : [0,1] \to [0,1]$,

$$W'(u,v) \triangleq W(\phi(u), \phi(v)) \tag{2.1}$$

for almost everywhere* (a.e.) $(u,v) \in [0,1]^2$, generates the same ExGM as W. The difficult question is the converse of the statement. Suppose that both W and W' generate the same ExGM $\mathbb{P}$, does there exist a measure preserving mapping ϕ such that equation (2.1) holds?

While in literature it is often thought that the converse is true, we can find counter examples to disprove this thought. As suggested by [17] , if we consider two graphons,

$$\begin{aligned} W(u,v) &= uv \text{ and} \\ W'(u,v) &\triangleq (2u \bmod 1)(2v \bmod 1). \end{aligned}$$

*For $[0,1]^d$ space, we always refer the term almost everywhere with respect to the complete Lebesgue measure on it.

then these two graphons will generate the same ExGM but there exists no such a measure preserving mapping ϕ satisfying equation (2.1). A more accurate condition for idetifiability is given in Theorem 7.1 by [17], which states that W and W' generate the same ExGM if and only if there exist two—rather than one—measure preserving mappings ϕ and ϕ' such that

$$W\left(\phi\left(u\right),\phi\left(v\right)\right)=W'\left(\phi'\left(u\right),\phi'\left(v\right)\right)\text{ a.e. }\left(u,v\right)\in\left[0,1\right]^{2}.$$

Note that an alternative and equivalent characterization that ensures W and W' generate the same ExGM is

$$\delta_{\square}\left(W,W'\right)=0,$$

where $\delta_{\square}$ is the so-called cut-metric [18].

However, the result by [17] does not mean that (2.1) should be fully abandoned. For such a relationship to hold among graphons generating the same ExGM, the following condition must be satisfied:

Definition 1 (Twin-free condition) *Two graphons are twin-free if there exists no such a pair* $\left(u_1,u_2\right)$ *in* $\left[0,1\right]$ *such that* $W\left(u_1,v\right)=W\left(u_2,v\right)$ *for a.e.* $v\in\left[0,1\right]$.

For any two twin-free graphons W_1 and W_2 generating the same ExGM, [19] proved that there is a measure preserving bijection $\phi_{12}:\left[0,1\right]\to\left[0,1\right]$ such that

$$W_1\left(u,v\right)=W_2\left(\phi_{12}\left(u\right),\phi_{12}\left(v\right)\right)\text{ for a.e. }\left(u,v\right)\in\left[0,1\right]^{2}. \tag{2.2}$$

Thus, for those papers misusing equation (2.1) as the only condition to define graphons generating the same ExGM, a simple fix would be to rephrase the results by limiting their interests to a subclass of ExGMs generated by twin-free graphons, which we call *twin-freely* generated ExGMs. Unfortunately, to the best of our knowledge, there is no known result

that states an appropriate way to choose a unique representation for graphons that generate a twin-free ExGM.

If there is no general result that hold for twin-free graphons, we then ask the question: would it be possible to resolve the identifiability issue for a subclass of ExGM? Such approach has been considered in [20], where they attempted to solve the identifiability issue by claiming that, for any ExGM $\mathbb{P}$ generated by a graphon W, one can find a measure preserving mapping ϕ such that, for $W^{\mathbb{P}}_{\text{can}} \triangleq W(\phi(u), \phi(v))$,

$$g^{\mathbb{P}}_{\text{can}}(u) \triangleq \int_0^1 W^{\mathbb{P}}_{\text{can}}(u, v)\, dv$$

is monotone non-decreasing for $u \in [0, 1]$. They also argued that the so-called canonical form $W^{\mathbb{P}}_{\text{can}}$ of the graphon W is uniquely determined for a.e. $(u, v) \in [0, 1]^2$.

We expand their condition by assuming the following:

Definition 2 (Degree-identifiability condition) *Let U be a uniform random variable on $[0, 1]$. Then the degree proportion*

$$g(U) \triangleq \int_0^1 W(U, v)\, dv$$

is an absolutely continuous random variable[†] *on* $[0, 1]$.

In their later work, [21] also rely on a similar assumption to ensure identifiability of the graphon.

An easy example to check why this extension is necessary is given by the following two

[†]We should note that the random variable $g(U)$ here is uniquely determined by the ExGM $\mathbb{P}$ in the distribution sense.

graphons

$$
\begin{aligned}
W(u,v) &\triangleq \mathbb{1}_{[0,1/2]^2}(u,v) + \mathbb{1}_{[1/2,1]^2}(u,v), \\
W'(u,v) &\triangleq \mathbb{1}_{[0,1/2]\times[1/2,1]}(u,v) + \mathbb{1}_{[1/2,1]\times[0,1/2]}(u,v),
\end{aligned}
$$

which give monotone non-decreasing $g(u) \equiv g'(u) \equiv 1/2$, generate a same ExGM, yet are different for a.e. $(u,v) \in [0,1]^2$. There is no canonical choice between W and W' in this example.

Therefore, estimation problem (P2) stated above will be well-posed as long as we focus on this subclass of ExGMs generated by degree-identifiable graphons, and if we treat the uniquely defined canonical graphon $W^{\mathbb{P}}_{\text{can}}$ associated with a degree-identifiable ExGM $\mathbb{P}$ as the major estimand of interest. The next section will discuss a strategy to develop estimation procedures under this context.

Remark 1 *For notation simplicity we denote W and g as the canonical graphon of a degree-identifiable ExGM and its marginal integral.*

Remark 2 *There are three equivalent characterizations for a degree-identifiable ExGM $\mathbb{P}$:*

1. *$g(U)$ is an absolutely continuous random variable;*
2. *$g^{\mathbb{P}}_{\text{can}}$ is strictly increasing on $[0,1]$.*
3. *The cumulative distribution function (CDF) of $g(U)$ is absolutely continuous and hence is continuous.*

Remark 3 *The strictly increasing property of $g^{\mathbb{P}}_{\text{can}}$ automatically makes sure that $W^{\mathbb{P}}_{\text{can}}$ is also twin-free, so degree-identifiable ExGM is indeed a subclass of twin-free ExGMs. Further-*

more, it's a proper subclass. We can justify this by considering the following graphon

$$W(u,v) \triangleq (u+v) \mod 1, \tag{2.3}$$

which is clearly twin-free but has a constant marginal integral $g(u) \equiv 1/2$, so this graphon will generate a twin-freely generated ExGM but not a degree-identifiable ExGM, which requires $g(U)$ to be an absolutely continuous random variable. It might be an interesting future research to define a canonical representation for twin-free graphons like equation (2.2).

Remark 4 *The requirement of a graphon's being twin-free will reject any cluster structure on nodes because every two nodes in the same class is by definition a pair of twin. Therefore, twin-free ExGM is in its nature incompatible to stochastic blockmodels, and so is the same for degree-identifiable ExGM as being a proper subclass of twin-freely generated ExGM.*

2.2 Three-step estimation of exchangeable graph models

In this Section, we discuss how, in a three steps procedure, to construct a flexible class of functional form or nonparametric estimates for the canonical graphon generating a degree-identifiable ExGM. Then we will demonstrate some comparative simulation studies and finally conclude with a special choice of nonparametric estimate.

The main idea behind the estimation procedure is to exploit the degree-identifiable feature of the canonical graphon and make use of empirical degree sorting to infer the unknown latent variables. Our proposed method consists of three steps.

Step 1: Probability matrix estimation. Perform any P1 estimation $\hat{P}$ for the probability matrix P.

Step 2: Latent variables estimation. Construct an empirical CDF of degree proportions using another P1 estimation $\hat{P}'$, which may or may not be the same as $\hat{P}$, and then let $\hat{U}_i$'s be the estimators of the unknown latent variables U_i's defined as the values of the empirical CDF evaluating at the degree proportions of i-th node in $\hat{P}'$.

The rationale of doing this Step 2 is explained in the following. From our simulation results, we find that the empirical CDF $\hat{F}(x)$ of degree proportions seems to describe the CDF of $g(U)$, which we denote as $g^{-1}(x)$, quite accurately when the number of nodes N is large enough. On the other hand, the law of large numbers can somehow guarantee that the degree proportions in P' at i-th node, $\frac{1}{N}\sum_{j=1}^{N} P'_{ij}$, will be a good approximation to $g(U_i)$ (assuming that given U_i, P'_{ij}'s are roughly i.i.d. from the distribution $W(U_i, U)$). As for a degree-identifiable ExGM, which requires the canonical graphon marginal integral g to be strictly increasing, we must have $u = g^{-1}(g(u))$ for every $u \in [0, 1]$, so we can trust the estimation of the latent variables $U_i = g^{-1}(g(U_i))$ by $\hat{U}_i \triangleq \hat{F}\left(\frac{1}{N}\sum_{j=1}^{N} P'_{ij}\right)$.

Remark 5 *In the descriptions above, we temporarily assume that there is no overlapping for degree proportions in $\hat{P}'$, that is, we assume that the elements of*

$$\left\{\frac{1}{N}\sum_{j=1}^{N} P'_{ij}\right\}_{i=1}^{N}$$

are distinct. We will resolve this overlapping issue later after we commit to a specific choice for $\hat{P}'$.

Once Step 1 and Step 2 are completed, we can construct a functional form estimate $\hat{W}(u, v)$. For now, we already have a set of three dimensional points

$$\left(\hat{U}_i, \hat{U}_j, \hat{W}\left(\hat{U}_i, \hat{U}_j\right)\right) \triangleq \left(\hat{U}_i, \hat{U}_j, \hat{P}_{ij}\right),$$

which we should treat as a noisy realization[‡] of the unknown canonical graphon plane at $\left(U_i, U_j, W\left(U_i, U_j\right)\right)$. To build a functional form estimation $\hat{W}(u, v)$ from those three dimensional points, we can either use a linear interpolation or a stepwise approximation as the pre-smoothing estimate. We majorly focus on the later one in this study, so the pre-smoothing estimate now takes the form of a step function

$$\hat{W}(u, v) \triangleq \sum_{1 \leq i,j \leq N} \hat{P}_{ij} \mathbb{I}_{\left(\hat{U}_i - 1/N, \hat{U}_i\right] \times \left(\hat{U}_j - 1/N, \hat{U}_j\right]}(u, v) .$$

Step 3: Smoothing. (Optional.) Apply any smoothing algorithm to the estimate to get a smoothed estimate.

Here are several notes related to this Step 3. First, it's a an optional step, and the choice of whether to include this step or not and how to conduct it depends on researchers' goal for inferences on network data and acceptability to those unavoidably additional assumptions on the canonical graphon.

Because both Step 1 and 2 can take any kind of estimator for problem (P1) to proceed, we need to know how to explicitly specify $\hat{P}$ and $\hat{P}'$.

Based on a comparative simulation study, which we omit for the sake of space, we have selected Universal Singular Value Thresholding (USVT) [22] as the solution estimation problem no. 1 (in Step 1), and the adjacency matrix A itself as the basis for degree sorting (in Step 2). In the remaining parts of the paper we focus on this combination for estimation in order to seek the least assumptions on W; we refer informally to this method as the USVT-A estimation procedure.

In this paper, we especially limit our choice for the two problem no. 1 estimations to be either the method of Universal Singular Value Thresholding (USVT) proposed by [22] or the adjacency matrix A itself. We describe the USVT method in the following Subsection.

[‡]With noise coming from not only the z-direction, but also the x- and y-directions as well.

2.2.1 Probability Matrix Estimation using Universal Singular Value Thresholding (USVT)

[22] proposed a general method for attacking estimation problems no. 1. His method requires only one observation data, can be easily implemented with tremendous speed, and assume almost non-criterions on the underlying graphon except for its measurability in a purely nonparametric sense (so no model bias actually). When the observation is large enough, USVT method provides a very promising estimation results. However, it's still addressing the P1 formulation, in which no complete form for the W_{can} function can be written down as one seeks in a problem no. 2 formulation.

where we will discussion different combinations later.

2.2.2 Comparative Simulation Study

In this subsection, we present two simulations to illustrate the performance of different combinations of graphon estimations constructed from the 3-step procedure.

In each simulation, we calculate the root of the mean square error (RMSE) between the constructed estimator (using N ranging from 300 to 3000) and the true graphon, where only two cases are considered here: the quadratic graphon $W(u,v) = \left(u^2 + v^2\right)/4$ and the logistic graphon $W(u,v) = \text{logistic}\left(-5 + 5\left(u+v\right)\right)$, where $\text{logistic}\left(x\right) \triangleq \left(1+\exp\left(-x\right)\right)^{-1}$.

The simulation results are shown in Table 1 and 2. In these Tables, the naming convention for each combination is to report the methods that were used for each of the steps separated by a dash; for example, "Step 1 method"-"Step 2 method"-"Step 3 method". The last smoothing step is optional–for example, USVT-$\mathcal{A}$-TVM method stands for using USVT in Step 1 for probability matrix estimation, using the plain adjacency matrix $\mathcal{A}$ in Step 2 for

Table 2.1: RMSE Simulation for Quadratic Graphon

N	300	900	1500
A-A	0.344657	0.359469	0.357767
USVT-USVT	0.035505	0.024235	0.017006
USVT-A	0.037397	0.024614	0.017132
A-A-TVM	0.02453	0.013044	0.009418
USVT-USVT-TVM	0.040357	0.01202	0.008492
USVT-A-TVM	0.040601	0.011967	0.008526
N	1800	2400	3000
A-A	0.351921	0.360604	0.358457
USVT-USVT	0.01695	0.013922	0.012097
USVT-A	0.017059	0.013995	0.012168
A-A-TV	0.010491	0.00895	0.007025
USVT-USVT-TV	0.009912	0.00813	0.006128
USVT-A-TV	0.009926	0.008101	0.006066

the empirical degree sorting, and finally using the total variation smoothing (TVM) in Step 3 [23]. We also include the combination A-A as a baseline estimation procedure.

From the results, we see that, for Step 1, using USVT is better than using A alone; for Step 2, sorting according to USVT estimate gives approximately the same result as (sometimes worse than) sorting according to the plain A; for Step 3, TVM smoothing can be helpful and reduce some mean square errors. It is interesting that A-A-TVM method gives a fairly good performance as both USVT-USVT-TVM and USVT-A-TVM methods. This observation motivated recent follow-up work [23].

We note that while adding a third smoothing step is helpful in these two specific examples, the *histogram estimator* recently proposed in [23] requires an additional smoothness assumption on the underlying canonical graphon W. As comparison, our proposed USVT-A estimation has slightly larger RMSE than [23], and the two methods have similar decay rate. Therefore, it is safe to argue that the proposed USVT-A method has a fairly good performance compared to the histogram estimator, but it requires less constraints on the

Table 2.2: RMSE Simulation of Logistic Graphon

N	300	900	1500
A-A	0.38003	0.3812	0.383409
USVT-USVT	0.102721	0.065759	0.034483
USVT-A	0.106061	0.069602	0.034192
A-A-TVM	0.085428	0.034208	0.02423
USVT-USVT-TVM	0.084122	0.051107	0.023318
USVT-A-TVM	0.075824	0.043535	0.02324
N	1800	2400	3000
A-A	0.380116	0.379474	0.379676
USVT-USVT	0.03164	0.024988	0.019176
USVT-A	0.031406	0.024771	0.019082
A-A-TVM	0.023198	0.019551	0.014232
USVT-USVT-TVM	0.023003	0.019187	0.014117
USVT-A-TVM	0.022963	0.019178	0.014113

graphon.

In the remainder of the paper, we focus on the USVT-A estimation in order to seek the least assumptions on W. Its theoretical property is discussed in the next Section.

2.3 Consistency

In this Section, we study the theoretical consistency of the USVT-A estimation procedure, which is defined through a combination of probability matrix estimation using USVT and the latent variables estimation using the empirical CDF of observed degrees proportions in A.

The main theoretical result of this paper is as follows:

Theorem 1 (USVT-A Consistency) *Assume that W is the canonical graphon of a degree-identifiable ExGM. If W is continuous on $[0,1]^2$, then the $\hat{W}$ constructed by the USVT-A*

method is consistent for estimating W in the sense that

$$\mathbb{E}\left(\int_0^1 \int_0^1 \left(\hat{W}(u,v) - W(u,v)\right)^2 dudv\right) \to 0.$$

The proof of this Theorem is detailed in the appendix.

Below are two results that make our main result hold, both of which correspond to the consistency of Step 1 and Step 2 in our proposed three steps procedure in Section 2.2.

Theorem 2 (USVT Consistency) *Let*

$$\hat{M} \triangleq \sum_{i \in \left\{s_i \geq 1.01\sqrt{N}\right\}} s_i u_i u_i^T \text{ and } \hat{P}_{ij} \triangleq \left(\left(\hat{M}_{ij}\right) \wedge 1\right) \vee 0,$$

be the USVT estimation of probability matrix P, where $\sum_{i=1}^{N} s_i u_i u_i^T$ is the singular value decomposition of adjacency matrix $\mathcal{A}$. Then

$$\mathbb{E}\left(\frac{1}{N^2}\sum_{i,j=1}^{N}\left|\hat{P}_{ij} - P_{ij}\right|^2\right) \to 0. \tag{2.4}$$

Proof: See [22][Theorem 4.11].

Now we define D_i as the observed degree proportions in $\mathcal{A}$,

$$D_i \triangleq \frac{1}{N}\sum_{j=1}^{N} \mathcal{A}_{ij},$$

with their empirical CDF defined as

$$\hat{F}(x) \triangleq \frac{1}{N}\sum_{i=1}^{N} 1_{\{D_i \leq x\}}.$$

Theorem 3 (Degree Sorting Consistency) *Let the empirical degree sorting estimate of the latent variables to be*

$$\hat{U}_i \triangleq \hat{F}(D_i)\,, \tag{2.5}$$

$$\tilde{U}_i \triangleq \hat{U}_i - \frac{\varkappa_i - 1}{N} \tag{2.6}$$

in the proposed USVT-A estimation, where $\varkappa_i$, given all of $\hat{U}_i$, is jointly distributed as follows: let $C_1, \ldots, C_M$ be those unique values of D_i's, and, if $D_{i_1} = \cdots = D_{i_{k_m}} = C_m$, then $\varkappa_{i_1}, \ldots, \varkappa_{i_{k_m}}$ are a uniform resampling of the set $\{1, 2, \ldots, k_m\}$. Then we have, for each $i = 1, \ldots, N$, $\left|U_i - \hat{U}_i\right| \to 0$ and $\left|\tilde{U}_i - \hat{U}_i\right| \to 0$ in probability, and hence $\left|U_i - \tilde{U}_i\right| \to 0$ in probability.

Theorem 3 describes the consistency of latent variables estimation via empirical degree sorting using the observed adjacency matrix A.

A proof for Theorem 1 using the two key consistency results described above is given in the Appendix.

2.4 Hypothesis testing

In this Section, we illustrate how the proposed USVT-A estimator can be used to study a hypothesis testing procedure to analyze network data. There is ample room for improvement of the procedure we describe here.

Hypothesis testing is a powerful procedure with limited literature in network data analysis. [24] presents likelihood ratio tests on three examples with a novel flavor: (i) ErdHos-Rényi graph [25, 26], (ii) stochastic blockmodel [27, 28], and (iii) the fixed-degree graph model [29]. However, such a method lacks the flexibility to cope with more sophisticated models, such as exchangeable graph models.

There are mainly two reasons why it is difficult to extend classical hypothesis testing theory to network data. The first is that modeling network data often involves latent variables. In case of ExGM, the U_i's are especially hard to handle. The second reason is the high computational cost of fitting existing methods, so it is often untenable to get the sampling distribution of the test statistic under the null hypothesis, using simulations.

The proposed USVT-$\mathcal{A}$ procedure captures the structure in exchangeable graph models by design. It is also computationally efficient so that Monte Carlo can be employed to obtain the sampling distribution. A simple illustrative example follows.

Suppose that we observe network data represented by an adjacency matrix $\mathcal{A}$, which is generated by a degree-identifiable ExGM with canonical graphon W. We want to test the two hypothesis: for $W_Q(u,v) \triangleq \frac{1}{4}(u^2+v^2)$,

$$H_0 : W(u,v) = W_Q(u,v) \text{ versus } H_a : W(u,v) \neq W_Q(u,v).$$

By Theorem 1, we see that the USVT-$\mathcal{A}$ estimate $\hat{W}$ converges to the true canonical graphon in mean squared error to the true canonical graphon W when N is sufficiently large. Thus we can choose the test statistic to be

$$\begin{aligned} T &\triangleq \sqrt{\int_0^1 \int_0^1 \left| W_Q(u,v) - \hat{W}(u,v) \right|^2 du dv} \\ &\triangleq \left\| W_Q - \hat{W} \right\|, \end{aligned} \tag{2.7}$$

i.e., the L^2 distance between W_Q and $\hat{W}$. Although we cannot analytically know the sampling distribution of T under the null hypothesis H_0, we can approximate using simulations because of the fast implementation of our USVT-$\mathcal{A}$ method. Using 5000 Monte Carlo samples for $N = 3000$, we get the histogram of T as in Figure 1. We can see that the sampling distribution of T under H_0 is right skewed. The mean and standard deviation of

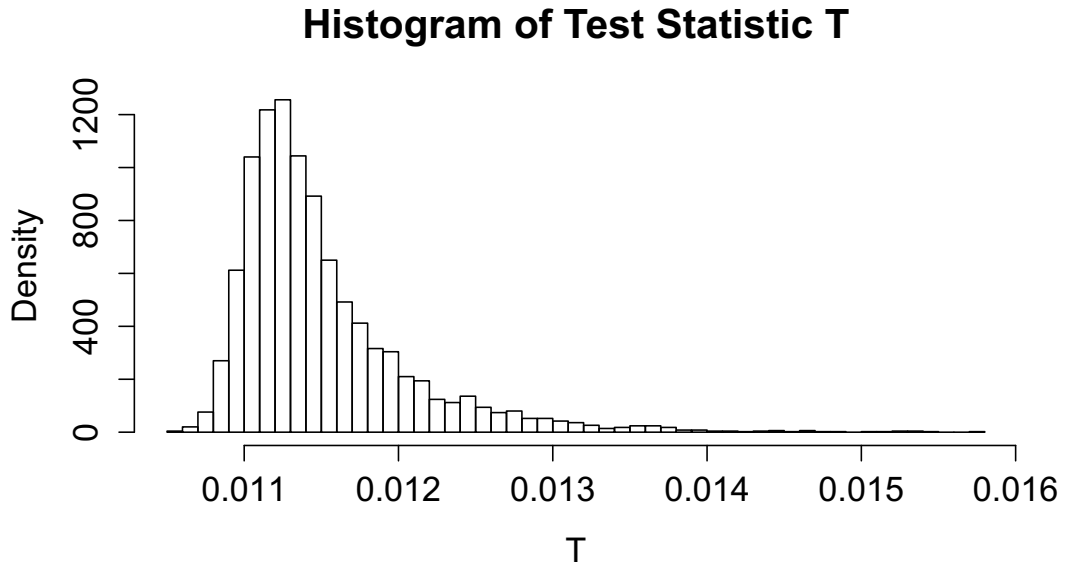


Figure 2.1: 5000 Monte Caro draws of T under H_0

the Monte Carlo samples are 0.0115 and 5.656×10^{-4}, and the 95% quantile is 0.0126, so the rejection region is $T \geq 0.0126$.

Given an observed adjacency matrix $\mathcal{A}$, we can calculate its corresponding $\hat{W}$ and T, and then reject H_0 if $T \geq 0.0126$.

We can then refer to the MSE of a data generating by logistic graphon, which is 0.019082, and then it gives us a p-value of much smaller than 0.001 because actually our 5000 simulations never generate a MSE greater than 0.019082. This then demonstrate a strong power of differentiating generating graphons under the scope of degree-identifiable ExGM.

2.5 Discussion

2.5.1 More about Identifiability

Furthermore, [19] provided one more equivalent formulation of equation (2.1.2) as follows. That is, there exists a twin-free $\tilde{W}$ defined on another probability space $\left(X, \mu_X\right)$ (i.e. there is no such a pair (x_1, x_2) in X such that $\tilde{W}(x_1, y) = \tilde{W}(x_2, y)$ for μ_X-a.s. $y \in X$) and measure preserving ϕ and ϕ' mappings from $[0,1]$ to X such that

$$W(u,v) = \tilde{W}\left(\phi(u), \phi(v)\right) \text{ and } W'(u,v) = \tilde{W}\left(\phi'(u), \phi'(v)\right),$$

for almost everywhere $(u,v) \in [0,1]^2$. Even though $\tilde{W}$ might live on another probability space X not necessarily the same as $[0,1]$, it seems plausible to consider a subclass of ExGM that could be generated from those twin-free graphons defined on $[0,1]$ and try to assign an unique representation W_{can} for each ExGM $\mathbb{P}$ in this subclass.

[19] actually proved that any graphon can lead to another twin-free graphon defined on the quotient space of $[0,1]$ with respect to the equivalent class of twins. This might shed a light in a future research direction that how one can pursue an ultimate canonical form of any ExGM.

Here we review some common ways to estimate the canonical graphon W underlying a degree-identifiable ExGM, and compare them to the proposed USVT-$\mathcal{A}$ estimator.

The first study of the properties of of an estimator for a graphon W has been carried out for a specific family of models; [20] considered blockmodels with a fixed number K of blocks [27]. They explicitly wrote the underlying graphon, and show why the maximum likelihood estimator for the blockmodel matrix is consistent while approaches based on modularity are not.

However, their estimator has some disadvantages: (i) the computational cost is high,

although in line with other estimators for blockmodels, and (ii) the need to pre-specify a fixed number of blocks K, which introduces a difficult model selection issue, in practice. It should be noted that although these papers formulate the estimation problem as P2, the estimation task is restricted to the parametric family of blockmodels, thus coming short of defining a general estimator $\hat{W}$ for the graphon W defining an ExGM.

To address some of the shortcomings in listed above, [30] and [16] consider a generalized blockmodel with a growing number of classes, $K = O\left(N^{1/2}\right)$. In this setting, inference does not need to have K pre-specified, while at the same time empirically leads to smaller model bias when compared with [20]. However, these papers still consider a parametric family of blockmodels, although less restrictive, and the estimation task is computationally expensive.

[31, 32] proposed a novel and fast way to calculate the graphon approximation by block models, in which they only require a Lipschitz continuity on the graphon W, but a clear disadvantage of them is the use of multiple graphical observations with external informations on the matchable nodes between different data graph, which is usually not applicable for data missing the nodes information and lack the original motivation in using ExGM. Also, they are in a P1 formulation.

[21] addresses the graphon estimation problem as a P2 formulation. They proposed a method of moment estimator that takes advantage of the counts of special subgraphs, referred to as *wheels*, in the observed network. They theoretically characterize the unknown graphon in terms of an abstract linear functional, based on the moments. While this approach is elegant, and leads to consistency in the absence of many assumptions on the graphon underlying an EcGM, the estimator is unfeasible in practice. This is because of the number of wheels could be huge for large networks and counting the frequency is computationally intractable. In addition, even given these counts, solving the canonical graphon from the characteristic linear functional described above is also challenging, because the

need to compute the eigenvalues and eigenvectors of the characteristic functional.

More recent work by [33] develops a nonparametric estimator for a graphon underlying an ExGM. They measure the error between the true graphon and the estimated graphon via the cut-metric [18]. The estimator is thus defined implicitly by an equation that is solvable only in theory. Their results do not allow explicitly numerical simulations to check the performance of the estimation. In addition, the asymptotic theory requires sophisticated assumptions, which may be untenable in practice.

In contrast, the proposed USVT-$\mathcal{A}$ estimator is computationally efficient, easy to implement, and come with the same consistency guarantees of existing methods, with little assumptions on the underlying graphon.

Singular Value Decomposition (USVT)

[22] proposed a very general method for attacking P1 estimation problem of graphon. Their method requires only one observation data, can be easily implemented with tremendous speed, and assume almost non-criterions on the underlying graphon except for its measurability in a purely nonparametric sense (so no model bias actually). When the observation is large enough, USVT method provides a very promising estimation results. However, it's still addressing the P1 formulation, in which no complete form for the W_{can} function can be written down as one seeks in a problem no. 2 formulation.

One of the major contributions of this paper proposed a way to transform a P1 formulation problem into a problem no. 2 formulation. See the next subsection for the discussion.

2.5.2 Concluding remarks

In this paper, we reformulated the existing literature on estimation problems for exchangeable graph models (ExGM), and dichotomized the existing approaches into two formulations: P1, addressing only on the probability matrix estimation; and P2, pursuing the fully

functional form estimate for the graphon underlying an ExGM.

We discussed the important issue of identifiability, which must be addressed before any attempts on addressing the P2 formulation of the estimation problem can take place. We characterized a subclass of exchangeable graph models, referred to as degree-identifiable ExGMs, which entails a uniquely-defined marginal degree function for the canonical graphon, and leads to a well-posed estimation problem. Within this subclass of models, we proposed a general 3-step procedure for constructing a flexible class of nonparametric estimates of the canonical graphon, which allows a large number of combinations of (i) probability matrix estimation methods, (ii) latent variable estimation methods, and (iii) smoothing methods.

We then focused on a pre-smoothing estimate, which we refer to as the USVT-$\mathcal{A}$ method. We theoretically proved its mean square error consistency, under the assumption of continuity of the canonical graphon degree function. Simulation results demonstrate the computational efficiency of the proposed USVT-$\mathcal{A}$ estimator, as well as its error properties, in practice. Our results also suggest that, if the canonical graphon W is believed to be smooth, then a smoothing algorithm like total variation minimization method [34] could be applied to get a further reduction of estimation errors [e.g., see 23]. However, simulation results also show that the reduction in RMSE obtained using total variation minimization seems to be relatively small. Other combinations of matrix estimators, latent variable estimators and smoothing methods should be considered as a promising avenue for future research.

3

Consistent estimation of dynamic and multi-layer block models

Significant progress has been made recently on theoretical analysis of estimators for the stochastic block model (SBM). In this paper, we consider the *multi-graph* SBM, which serves as a foundation for many application settings including dynamic and multi-layer networks. We explore the asymptotic properties of two estimators for the multi-graph SBM, namely spectral clustering and the maximum-likelihood estimate (MLE), as the number of layers of the multi-graph increases. We derive sufficient conditions for *consistency* of both estimators and propose a variational approximation to the MLE that is computationally feasible for large networks. We verify the sufficient conditions via simulation and demonstrate that they are practical. In addition, we apply the model to two real data sets: a dy-

namic social network and a multi-layer social network with several types of relations.

3.1 Introduction

Modeling relational data arising from networks including social, biological, and information networks has received much attention recently. Various probabilistic models for networks have been proposed, including stochastic block models and their mixed-membership variants [28, 35]. However, in many settings, we not only have a single network, but a collection of networks over a common set of nodes, which is often referred to as a *multi-graph*. Multi-graphs arise in several types of settings including dynamic networks with time-evolving edges, such as time-stamped social networks of interactions between people, and multi-layer networks, where edges are measured in multiple ways such as phone calls, text messages, e-mails, face-to-face contacts, etc.

A significant challenge with multi-graphs is to extract common information across the *layers* of the multi-graph in a concise representation, yet be flexible enough to allow differences across layers. Motivated by the above examples, we consider the *multi-graph stochastic block model* first proposed by [36], which divides nodes into classes that define blocks in the multi-graph. The key assumption is that nodes share the same block structure over the multiple layers, but the class connection probabilities may vary across layers. We believe this model is a flexible and principled way of analyzing multi-graphs and provides a strong foundation for many applications. The special case of a single layer, often referred to simply as the stochastic block model (SBM), has been studied extensively in recent years [20, 30, 16, 37, 38, 39, 40]. However, the more general multi-graph case has not been studied as much.

In this paper, we explore the asymptotic properties of several estimators for the multi-graph SBM by letting the number of network layers grow while keeping the number of

nodes *fixed*. We prove that a spectral clustering estimate of the class memberships is consistent for a special case of the model (Section 3.4.1). Next we derive sufficient conditions under which the maximum-likelihood estimate (MLE) of the class memberships is consistent in the general case (Section 3.4.2). Finally we propose a variational approximation to the MLE that is computationally tractable and is applicable to many multi-graph settings including dynamic and multi-layer networks (Section 3.4.2). We apply the spectral and variational approximation methods to several simulated and real data sets, including both a dynamic social network and a social network with multiple types of relations between people (Section 3.5).

Our main contribution is the consistency analysis for the MLE, which ensures the tractability of the model and paves the way for more sophisticated models and inference techniques. To the best of our knowledge, we provide the *first* theoretical results for the multi-graph SBM for a growing number of layers.

3.2 Related work

Probabilistic models for networks have been studied for several decades; many commonly used models are discussed in the survey by [35]. More recent work includes non-parametric network models using graphons [31, 33, 41]. Most previous models assume that a single network, rather than a multi-graph, is observed.

Two settings where multi-graphs arise include dynamic and multi-layer networks. Dynamic network models typically assume that a sequence of network snapshots is observed at discrete time steps. Previous work on dynamic network models has built upon models for a single network augmented with Markovian dynamics. [42, 43] built upon exponential random graph models. [44, 45, 46, 47, 48] built on stochastic block models. [49, 50, 51] used latent space models. [52, 53, 54] used latent feature models.

Multi-layer networks consider multiple types of connections simultaneously. For example, Facebook users interact by using "likes", comments, messages, and other means. Multi-layer networks go by many other names like multi-relational, multi-dimensional, multi-view, and multiplex networks. The analysis of multi-layer networks has a long history [36, 55, 56, 57, 58, 59]. However there has not been much work on probabilistic modeling of such networks, aside from the multi-view latent space model proposed by [60], which couples the latent spaces of the multiple layers.

A third related setting involves modeling populations of networks, where each observation consists of a network snapshot drawn from a probability mass function over a network-valued sample space. [61] proposed a nonparametric Bayesian model for this setting. This setting differs from the multi-graph setting that we consider in this paper because the network snapshots (layers) are drawn in an independent and identically distributed (iid) fashion, with no coupling between the snapshots.

The statistical properties of the inference algorithms in both dynamic and multi-layer network models have not typically been studied. Recently there has been a lot of progress on consistency analysis for single networks. Maximum-likelihood estimation, its variational approximation, and spectral clustering have all been proven to be consistent under the stochastic block model [20, 30, 16, 37, 62, 38, 39, 63, 8] as the number of nodes $N \to \infty$. Intuitively, for each new node added to the graph, we observe N realizations, hence larger N provides more information leading to consistent estimation of the model.

We extend the ideas used in single networks to multi-graphs. We note that the asymptotic regime is different in this case. For a single network, one typically lets $N \to \infty$, while for multi-graphs, we let $T \to \infty$ with N *fixed*. Intuitively it means we do not need to observe a very large network to get a correct understanding of the structure. Instead, we can gain the information through multiple samples, which may represent, for example, multiple observations over time or multiple relationships. In practice, it may be more realistic to

allow N to grow along with T, particularly for the dynamic network setting. Allowing N to grow provides *more* information; thus our analysis with fixed N serves as a conservative analysis for different settings.

3.3 Multi-graph stochastic block model

We present an overview of the *multi-graph stochastic block model* first proposed by [36]. A single relation is represented by an adjacency matrix $G^t = (G^t_{ij}),\ i,j = 1,\ldots,N$. We focus on symmetric binary relations with no self-edges. For a multi-graph, we observe an *adjacency array* $\vec{G} = \{G^1,\ G^2,\ \ldots,\ G^T\}$ sharing the same set of nodes. Subscripts denote the same node pairs for any t, while the superscript t indexes layers of the multi-graph. A layer may refer to time or type of relation depending on the application. If $\vec{G}$ is a random adjacency array for N nodes and T relations, then the probability distribution of $\vec{G}$ is called a *stochastic multi-graph*. Let the edge G^t_{ij} be a Bernoulli random variable with success probability Φ^t_{ij}. $\Phi^t = (\Phi^t_{ij}) \in [0,1]_{N\times N}$ is the probability matrix of graph G^t. Let $\vec{\Phi} = \{\Phi^1,\ \Phi^2\ \ldots\ \Phi^T\}$ be the *probability array*. We assume the independence of edges within and across layers conditioned on the probability array. That is, the adjacency array is generated according to

$$G^t_{ij}|\vec{\Phi} \overset{\text{ind}}{\sim} \text{Bern}(\Phi^t_{ij}).$$

The multi-graph stochastic block model is a special case of a stochastic multi-graph. In the multi-graph SBM, networks are generated in the following manner. First each node is assigned to a class with probability $\pi = \{\pi_1,\ldots,\pi_K\}$ where π_k is the probability for a node to be assigned to class k. Then, given that nodes i and j are in classes k and l, respectively, an edge between i and j in network layer t is generated with probability P^t_{kl}. In other words, nodes in the same classes in the same layer have the same connection probability governed by $\vec{P} = \{P^1,\ P^2,\ldots,P^T\} \in [0,1]_{K\times K}$, the *class connection probability array*. Let

$c_i \in \{1, \ldots, K\}$ denote the class label of node i. Then $\Phi^t_{ij} = P^t_{c_i c_j}$.

The nodes have class labels $\vec{c}$ *shared* by all of the layers of the multi-graph, but in each layer the class connection probabilities P^t_{kl} may be different. As we consider undirected networks, P^t is a symmetric matrix with $K(K+1)/2$ free parameters. One can see that the (single network) SBM is a special case of the multi-graph SBM with $T = 1$.

Though simple, this multi-graph model has not been formally studied in the a posteriori setting where class labels are estimated. It serves as a basis for many settings including dynamic networks and networks with multiple relations. More importantly, it can be theoretically analyzed and can provide insight on more complex models.

3.4 Consistent estimation for the multi-graph stochastic block model

[36] only discussed estimation of the multi-graph SBM with blocks specified a priori. The sample proportion of each layer t is the maximum-likelihood estimate (MLE) of the class probability matrix P^t. However, in most applications, the block structure is unknown. Hence our main goal is to accurately estimate the class memberships. We extend several inference techniques used for the single network SBM to the multi-layer case.

It is not immediately straightforward how we can utilize inference techniques designed for the single network SBM. One may imagine inferring $\vec{c}$ independently from each network and averaging across them, e.g. by majority voting. That is, each node is assigned the class label that occurs most often. We find in simulations that this ad-hoc method often does not work well.

We propose spectral clustering on the mean graph as a motivating method for a special case of the model. Then we discuss maximum-likelihood estimation, a natural way to combine the information contained in the different layers, for the general case. Maximum-likelihood estimation is intractable for large networks so we also consider a variational ap-

proximation to the MLE.

Our main focus is on the consistency properties of these methods. We consider a fixed number of nodes N but let the number of graph layers $T \to \infty$. In reality, although we do not have infinite layers, we often encounter situations with a large number of layers, such as dynamic networks over long periods of time.

3.4.1 Consistency of spectral clustering

Spectral clustering is a popular choice for estimating the block structure of the SBM because it scales to large networks and has shown to be consistent as $N \to \infty$ [64]. The method is based on singular value decomposition and K-means clustering on the singular vectors.

A natural way to extend spectral clustering from single networks to multi-graphs is to apply spectral clustering on the mean graph $\bar{G} = \frac{1}{T}\sum_{t=1}^{T} G^t$. This method is intuitively appealing as it matches with the assumption of a single set of class labels shared by all of the layers. We show that under some stationarity and ergodicity conditions, it indeed provides a consistent estimate of the class assignments. Specifically we consider the case where the class connection probabilities P^t vary across layers but have the same mean $\mathcal{M}$. The following theorem shows the consistency of spectral clustering on the mean graph $\bar{G}$ if the mean $\mathcal{M}$ is identifiable.

Theorem 4 *Assume $\vec{P}$ follows a stationary ergodic process such that $E(P^t_{kl}) = \mu_{kl}$ and $Var(P^t_{kl}) = \varepsilon^2_{kl}$ for all t. Assume $\mathcal{M} = [\mu_{kl}]$ is identifiable, i.e. $\mathcal{M}$ has no identical rows. Let $\bar{G} = \frac{1}{T}\sum_{t=1}^{T} G^t$. Spectral clustering of $\bar{G}$ gives accurate labels as $T \to \infty$. That is, let $U_{N\times K}$ be the first K right singular vectors in the singular value decomposition of $\bar{G}$. K-means clustering on the rows of $U_{N\times K}$ outputs class estimates $\hat{c}_1, \ldots, \hat{c}_N$. Up to permutation, $\hat{c} = c$, a.s. as $T \to \infty$.*

We provide a sketch of the proof; details can be found in Appendix B.1. Since we have independent errors in the probability matrix and also independent errors in the Bernoulli observations, averaging cancels the error so that $\bar{G} \to CMC'$. Here C is a rank K matrix incorporating the class assignment vectors. Using an inequality from [65], we bound the distance between the singular vectors of $\bar{G}$ and CMC'. Therefore, spectral clustering on $\bar{G}$ clusters the nodes into K different classes.

Remark 6 *To determine the number of classes is a difficult model selection problem even for a single network. We will not discuss this problem in detail. We assume K is fixed and known in this paper.*

Remark 7 *The diagonal of G^t is always* 0 *because no self-edges are allowed; however the diagonal of CMC' is not necessarily* 0. *This may not cause a problem as $N \to \infty$. But for finite N, it may cause error in estimating the singular vectors. If this is the case, we may utilize the singular value decomposition that minimizes the off-diagonal mean square error*

$$\arg\min_{U,S} \sum_{i<j} (\bar{G}_{ij} - U_i S U_j')^2,$$

which can be computed by iterative singular value decomposition [66].

The condition in Theorem 4 requiring $\vec{P}$ to be stationary with identifiable mean M is restrictive. Spectral clustering on the mean graph is not effective in many cases. Consider the case for which

$$P^t \in \left\{ \begin{pmatrix} 0.7 & 0.3 \\ 0.3 & 0.7 \end{pmatrix}, \begin{pmatrix} 0.3 & 0.7 \\ 0.7 & 0.3 \end{pmatrix} \right\}$$

where both outcomes are equally likely for all t. Then

$$M = \begin{pmatrix} 0.5 & 0.5 \\ 0.5 & 0.5 \end{pmatrix}$$

is not identifiable. Spectral clustering on the mean graph fails to correctly estimate the class assignments as $T \to \infty$. But there is information contained in this multi-graph. We can use maximum likelihood estimation to estimate the class assignments correctly in this case.

3.4.2 Consistency of maximum likelihood estimate

Now we focus on the general case where we do not place any structure on $\vec{P}$. A natural way to estimate the class assignment is to use the maximum-likelihood estimate (MLE). We show that for a large enough fixed N, the MLE will estimate the class memberships correctly as $T \to \infty$.

First we define some notation. For any class assignment z, let $n_k(z) = \#\{i : z_i = k\}$ be the number of nodes in class k. Let $m(z) = \min_k n_k(z)$ denote the minimum number of nodes in any class under labels z. Let

$$n_{kl}(z) = \begin{cases} n_k n_l, & k \neq l \\ n_k(n_k - 1)/2, & k = l. \end{cases}$$

be the number of pairs of nodes in each block. We drop the dependency on z whenever it is unambiguous. We also drop the superscript t when we talk about a single layer of the network.

Now define some notation related to the MLE. The complete log-likelihood for parame-

ters (z, P) is

$$l(z, P) = \sum_{i<j} \left(G_{ij} \log(P_{z_i z_j}) + (1 - G_{ij}) \log(1 - P_{z_i z_j}) \right).$$

Here P is a parameter not to be confused with the true class connection probability matrix. In particular, we are interested in the case $P_{kl} = \bar{P}_{kl}(z)$ where

$$\bar{P}_{kl}(z) = \frac{1}{n_{kl}(z)} \sum_{\substack{i:z_i=k \\ j:j\neq i, z_j=l}} P_{c_i c_j}.$$

Here $\bar{P}$ is the average of the true P under block assignment z. To ease notation, let

$$\sigma(p) = p \log(p) + (1 - p) \log(1 - p).$$

Denote the expectation of log-likelihood of $(z, \bar{P}(z))$ as

$$h(z) = E[l(z, \bar{P}(z))] = \sum_{k\leq l} n_{kl}(z)\, \sigma(\bar{P}(z)).$$

Now, as we do not observe the true P, the natural step is to estimate it with the empirical mean for any given z. So let

$$o_{kl}(z) = \sum_{\substack{z_i=k \\ z_j=l}} \begin{cases} G_{ij}, & k \neq l \\ G_{ij}/2, & k = l. \end{cases}$$

be the observed number of edges in block (k, l). Then the profile log-likelihood [20] is defined as

$$f(z) = \sum_{k\leq l} n_{kl}(z)\, \sigma\left(\frac{o_{kl}(z)}{n_{kl}(z)} \right).$$

Let the expectation of f be

$$g(z) = E(f(z)) = \sum_{k \le l} n_{kl}(z)\, E\left[\sigma\left(\frac{o_{kl}(z)}{n_{kl}(z)}\right)\right].$$

Now we are ready to state the consistency of the MLE for the multi-graph SBM. If all elements of P^t are bounded away from 0 and 1 and their column differences are at least some distance apart, then when we have a sufficient number of nodes in each block, the true label c uniquely maximizes the sum of profile log-likelihoods over the layers.

Theorem 5 *Let*

$$C_0 = \inf_{t,k,l}(P^t_{kl}, 1 - P^t_{kl})$$
$$\delta = \inf_{t,k,l} \max_m \left[\sigma(P^t_{km}) + \sigma(P^t_{lm}) - 2\sigma\left(\frac{P^t_{km} + P^t_{lm}}{2}\right)\right].$$

Assuming $C_0 > 0$ *and* $\delta > 0$, *if* $m(c) = \min_k n_k(c)$ *is sufficiently large, then*

$$\hat{c} = \arg\max_z \sum_t f^t(z) \to c, \text{ a.s. as } T \to \infty.$$

The idea is that $\sum_t f^t(z)$ is the sum of independent profile log-likelihoods. We need N to be sufficiently large so that the expectation of the profile log-likelihood at each layer is maximized at the true labels c. Then as $T \to \infty$, we have convergence to expectation for $\sum_t f^t(z)$. We formalize the ideas by establishing the following lemmas.

Lemma 1 (From [16]) *For any label assignment* z, *let* $r(z)$ *count the number of nodes whose true class assignments under* c *are not in the majority within their respective class assignment under* z. *Let*

$$\delta = \min_{k,l} \max_m \sigma(P_{km}) + \sigma(P_{lm}) - 2\sigma\left(\frac{P_{km} + P_{lm}}{2}\right).$$

Then the expectation of the log-likelihood h is maximized by $h(c)$, and

$$h(c) - h(z) \geq \frac{r(z)}{2} \delta \min_k n_k(c).$$

In particular, for all $z \neq c$,

$$h(c) - h(z) \geq \frac{1}{2} \delta \min_k n_k(c).$$

Lemma 1 shows the expectation of the log-likelihood is maximized at the true parameters, and the difference of the true parameters and any other candidate is at least some distance apart which depends on the column difference of the probability matrix. However, as we work with the profile log-likelihood, we establish Lemmas 2 and 3 to bound the difference between the expectation of the profile log-likelihood and the complete log-likelihood.

Lemma 2 *Let $x \sim \frac{1}{N} Bin(N, p)$. For $p \in (0, 1)$,*

$$E(\sigma(x)) \to \sigma(p) + \frac{1}{2N} + O\left(\frac{1}{N^2}\right), \text{ as } N \to \infty.$$

Lemma 3 *Assume $C_0 \leq P_{kl} \leq 1 - C_0$, $C_0 > 0$. For any $\delta_0 > 0$ and any z, if $\min_k n_k(z)$ is large enough, then the difference between the expectation of the profile log-likelihood $g(z)$ and the expectation of the complete log-likelihood $h(z)$ is bounded in the following manner:*

$$\left| g(z) - h(z) - \frac{K(K+1)}{4} \right| \leq \delta_0.$$

Lemma 2 utilizes a Taylor series expansion. For simplicity, we use big $O(\cdot)$ notation instead of specifying an actual bound. Readers can refer to Appendix B.2 for the bound and the constants in the bound. Lemma 3 uses Lemma 2 and shows that with a sufficiently large number of nodes, the difference of the expectation of the profile log-likelihood and com-

plete log-likelihood is $K(K+1)/4$ and a negligible term δ_o. Combining the lemmas and using the concentration inequality, we can show that Theorem 5 provides sufficient conditions for the consistency of the multi-graph SBM. The proofs of Lemmas 1–3 and Theorem 5 can be found also in Appendix B.2.

Remark 8 *The main difference between the $N \to \infty$ case considered in most previous work and the $T \to \infty$ case that we consider is that, for $N \to \infty$, a direct bound is put on f and l. For $T \to \infty$, we need only to bound the* expectation *of f and l. This is newly studied here. In other words, for some particular class connection probability matrix P, the number of nodes required in a single network to have an accurate estimate is much larger than what is needed in a multi-graph with a growing number of layers.*

Variational approximation

The MLE is computationally infeasible for large networks because the number of candidate class assignments grows exponentially with the number of nodes. To overcome the computational burden, variational approximation, which replaces the joint distribution with independent marginal distributions, can be used to approximate the MLE. [67] has a detailed discussion of variational approximation in the SBM. We adapt it to the multi-graph SBM, resulting in the following update equations:

$$b_{ik} \propto \pi_k \prod_{j \neq i} \prod_t \prod_l \left[(P^t_{kl})^{g^t_{ij}} (1 - P^t_{kl})^{1 - g^t_{ij}} \right]^{b_{jl}}$$

$$\pi_k \propto \sum_i b_{ik}$$

$$P^t_{kl} = \frac{\sum_{i \neq j} b_{ik} b_{jl} g^t_{ij}}{\sum_{i \neq j} b_{ik} b_{jl}},$$

where the b_{ik}'s denote the variational parameters. The derivation is straightforward; we provide details in Appendix B.4.

Variational approximation has been shown to be consistent in the SBM [37, 39]. We conjecture that the performance of variational approximation is also good in the multi-graph SBM. Unless otherwise specified we use variational approximation to replace the MLE in all experiments.

3.5 Experiments

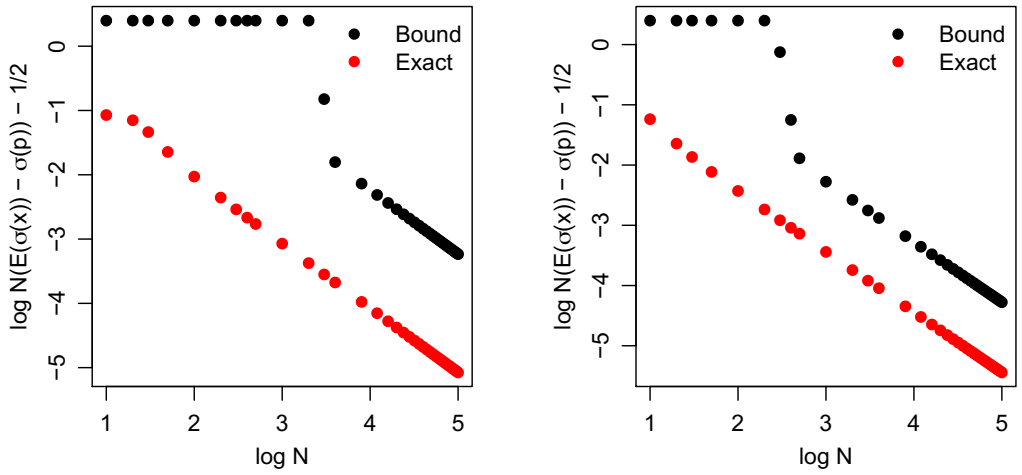
3.5.1 Numerical illustration

We begin with a toy example where we investigate empirically how many nodes are needed for the profile MLE to correctly recover the classes as $T \to \infty$. Due to the computational intractability of computing the exact profile MLE, we consider very a small network with $N = 16$ nodes and $K = 2$ classes where each class has 8 nodes. Consider two multi-graph SBMs with the following probability matrices:

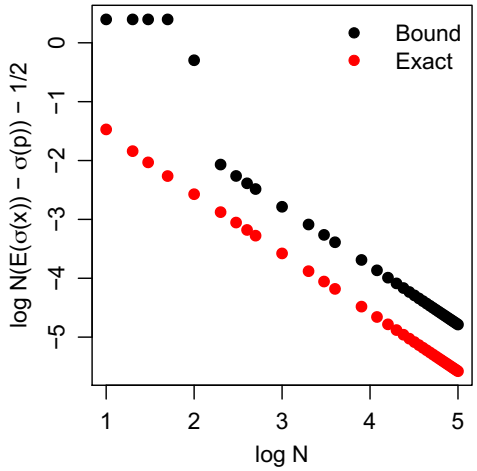
$$\text{Case 1: } P^* \equiv \begin{pmatrix} 0.55 & 0.45 \\ 0.45 & 0.55 \end{pmatrix}$$

$$\text{Case 2: } P^* \equiv \begin{pmatrix} 0.51 & 0.49 \\ 0.49 & 0.51 \end{pmatrix}$$

The δ (defined in Theorem 5) corresponding to the row difference of P^* is much smaller in case 2. Empirically the profile MLE succeeds to get the true labels in case 1 while it fails in case 2. Further analysis shows that in order to have consistency given the class connection probability matrix P^* in case 2, the total number of nodes should be at least 40. This toy example demonstrates that conditions on the probability matrices and network size are



(a) $p = 0.1$ **(b)** $p = 0.25$



(c) $p = 0.4$

Figure 3.1: Comparison of bound in Lemma 2 to exact values of $N(E(\sigma(x)) - \sigma(p)) - 1/2$ for varying N and p. The tightness of the bound affects the minimum number of nodes required to guarantee consistency in Theorem 5.

Table 3.1: Minimum number of nodes N required for consistency of the profile MLE with $K = 2$ classes under different values for parameters C_0 and δ from Theorem 5.

δ \ C_0	0.3	0.25	0.2	0.15	0.1	0.05
0.165	42	50	64	88	124	184
0.091	44	52	66	92	142	234
0.040	46	56	70	94	148	314
0.010	66	68	74	100	156	330

necessary for consistency. Theorem 5 provides *sufficient* conditions.

Next we investigate the tightness of the conditions in Theorem 5. The tightness of Lemma 1 was studied by [16]. We check the tightness of Lemma 2. For different p, we can calculate the exact value of $N(E(\sigma(x)) - \sigma(p)) - 1/2$ and compare it to the bound from Lemma 2. Figure 3.1 shows that the bound is loose for small N, but has almost the same asymptotic decay as the exact calculation. For small N, the remainder in Taylor expansion causes deviation. Also the bounds are looser for p closer to 0 or 1 but still informative in most cases.

For the special case of $K = 2$ classes, we can calculate all of the constants in the sufficient conditions in Theorem 5 for different values of C_0 and δ by enumerating cases. Details are provided in Appendix B.3. Table 3.1 shows the smallest number of nodes N that is sufficient for consistency of the profile MLE to hold for different values of C_0 and δ. Note that the minimum N is in the tens or hundreds, suggesting that the bounds in Theorem 5 are not overly loose and are indeed of practical significance.

3.5.2 Comparison with majority voting

As previously mentioned, majority voting is another way to utilize inference methods for a single network on multi-graphs. We consider two majority vote methods as baselines for comparison, one that utilizes spectral clustering on each layer, and one that applies a variational approximation to each layer. When using majority voting between different layers

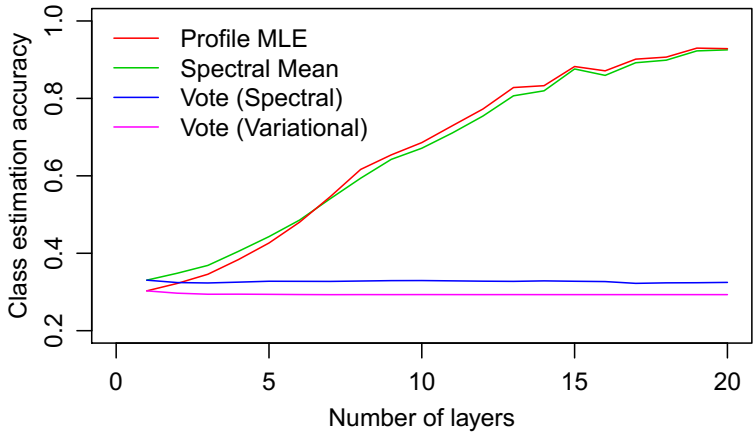


Figure 3.2: Simulation experiment comparing the proposed methods of profile MLE and spectral clustering on the mean graph with two majority vote baselines. The proposed methods increase in accuracy as the number of layers increases, but the two heuristic methods based on majority vote do not.

of the network, the estimated class labels for each layer must first be aligned or matched. We utilize the Hungarian algorithm [68] to compute the maximum agreement matching between the estimated labels at layer t with the majority vote up to layer $t-1$.

We conduct simulations to compare our proposed methods of spectral clustering on the mean graph and profile maximum-likelihood estimation with the majority vote baselines. We consider a well-studied scenario where we have 128 nodes initialized randomly into 4 classes [69]. For each layer, the within-class connection probability is 0.0968, and the between-class connection probability is 0.0521. Under such connection probabilities, the classes are below the detectability limit [70] for a single layer, so the class estimation accuracy from a single layer is very low. We increase the number of layers and observe how the accuracy changes.

Figure 3.2 shows the accuracy of the two proposed methods compared to the two majority voting methods averaged over 100 replications. Both the profile MLE and spectral clustering on the mean graph have the anticipated increasing accuracy over time. But the accuracies of the two heuristic majority vote methods do not improve. Though one may ex-

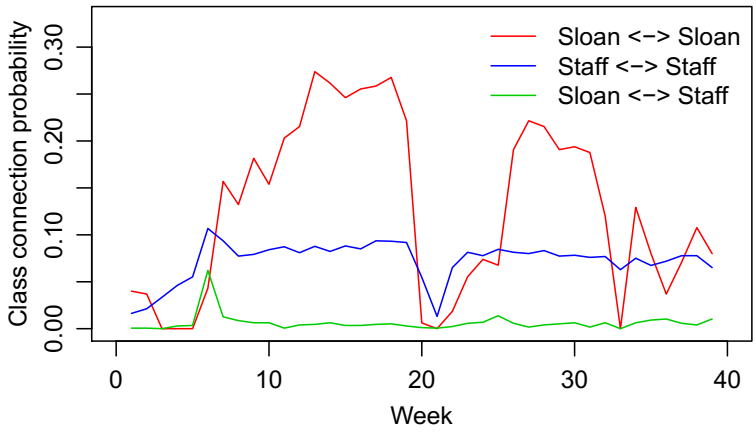


Figure 3.3: Estimates of class connection probabilities in the Reality Mining data set. The probabilities vary significantly over time, particularly for edges between Sloan students.

pect the errors in majority vote to be canceled out over time, these results show that, without careful averaging of errors, we cannot gain from the multiple layers. We find that this is due to choosing connection probabilities below the detectability limit; if we make the estimation problem easier by increasing the within-class probability above the detectability limit, then the majority vote methods do improve with increasing layers, albeit much slower than the methods we propose in this paper.

3.5.3 MIT Reality Mining data

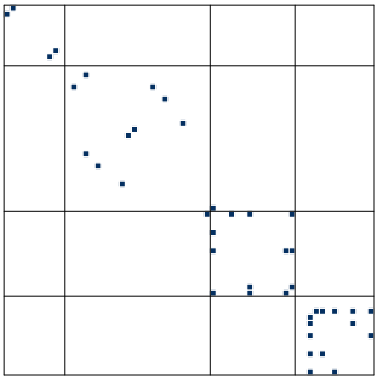
Next we apply our model on the MIT Reality Mining data set [71]. This data set comprises 93 students and staff at MIT in the 2004-2005 school year during which time their cell phone activities were recorded. We construct dynamic networks based on physical proximity, which was measured using scans for nearby Bluetooth devices at 5-minute intervals. We exclude data near the beginning and end of the experiment where participation was low. We discretize time into 1-week intervals, similar to [72, 73], resulting in 39 time steps between August 2004 and May 2005.

We treat the affiliation of participants as ground-truth class labels and test our proposed

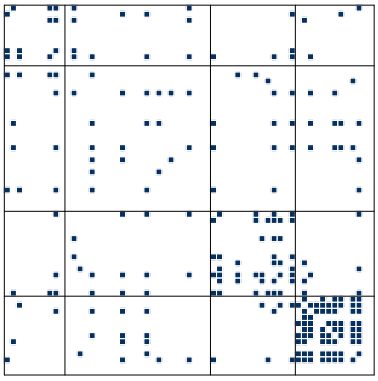
Table 3.2: Class estimation accuracy in the Reality Mining data set given data up to week listed in the first column. Best performer in each row is listed in bold. Both the proposed spectral clustering on the mean graph and profile maximum-likelihood estimation approaches improve over time, but majority vote does not.

Week	Maj. vote	Spectral Mean	Profile MLE
10	0.76	0.62	0.57
15	0.82	0.94	0.95
20	0.83	0.95	0.98
25	0.78	0.95	0.99
30	0.80	0.97	0.99
35	0.80	0.97	0.99
End	0.77	0.97	0.99

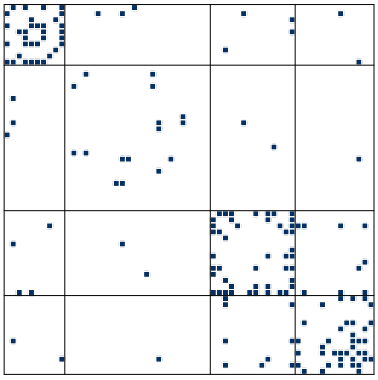
methods. Two communities are found: one of 26 Sloan business school students, and one of 67 staff working in the same building. Since degree heterogeneity may cause problems in detecting communities using the SBM [74], we reduce its impact by connecting each participant to the 5 other participants who spent the most time in physical proximity during each week. Figure 3.3 shows the empirical block connection probabilities within and between the two classes, estimated by the profile MLE. The class connection probabilities vary significantly over time, which validates the importance of the varying class connection probability assumption in our model. Notice that the two communities become well-separated around week 8. The class estimation accuracies for the different methods are shown in Table 3.2. Since the community structure only becomes clear at around week 8, the spectral and profile MLE methods are initially worse than majority voting but quickly improve and are superior over the remainder of the data trace. By combining information across time, the proposed methods successfully reveal the community structure while majority voting continues to improperly estimate the classes of about 20% of the people.



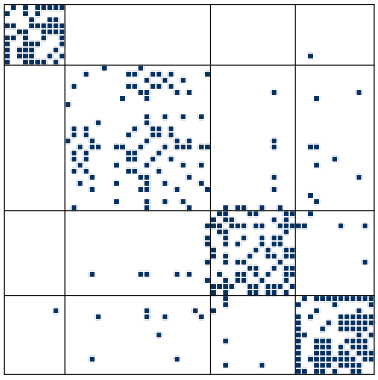
(a) Co-authorship



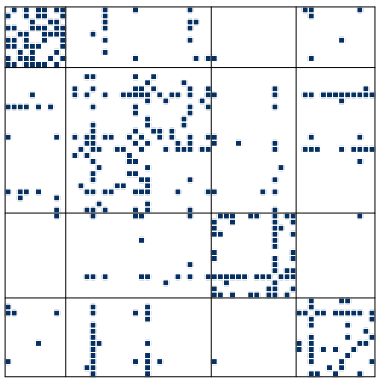
(b) Facebook



(c) Leisure



(d) Lunch



(e) Work

K	ICL
2	4087
3	3914
4	3830
5	3841
6	3878

(f) ICL (lower denotes better fit)

Figure 3.4: The estimated community structures in the AU-CS multi-layer networks overlaid onto the adjacency matrices of different relations. The dots denote connections (edges), and the grids correspond to SBM blocks.

3.5.4 AU-CS multi-layer network data

We look at another example from a multi-layer network comprising five kinds of self-reported on-line and off-line relationships between the employees of a research department: Facebook, leisure, work, co-authorship, and lunch [75]. We assume the class structure to be invariant across the different types of relations and apply our model. For model selection, we extend the Integrated Classification Likelihood (ICL) criterion proposed by [67] for the single network SBM to multi-graphs to select the number of blocks K. Specifically we maximize

$$-2Q(\vec{G}) + (K-1)\log N + \frac{TK(K+1)}{2}\log\frac{N(N-1)}{2},$$

where $Q(\vec{G})$ is the variational approximation to the complete log-likelihood. We initialize the variational approximation with different randomizations as well as the spectral clustering solution. The term is maximized at $K = 4$.

Figure 3.4 shows the estimated 4 classes overlaid onto the adjacency matrix of each relation. Although we have no ground truth for this data set, we detect well-separated communities in all relations aside from co-authorship, which is an extremely sparse layer. Notice once again the difference in empirical connection probabilities over the multiple layers of the multi-graph.

For this data set, we do not have ground truth labels to evaluate the class estimation accuracy. We note, however, that the ICL obtained by our variational approximation algorithm is much better than the ICLs obtained by fitting an SBM on the mean graph and by majority vote, both of which are over 4000.

3.6 Discussion

In this paper, we investigated the multi-graph stochastic block model applied to dynamic and multi-layer networks with invariant class structure. Both spectral clustering on the

mean graph and maximum-likelihood estimation are proved to be consistent for a fixed number of nodes when we have an increasing number of network layers, provided certain sufficient conditions are satisfied.

There are several interesting avenues for extensions of our analysis. First we can add a layer of probabilistic modeling on the probability matrices if we have additional information. Since dynamic networks usually vary smoothly over time, we can put a state-space model on the adjacency array [47]. We can also use a hierarchical model on the probability matrices to couple them for analyzing multi-layer networks. Since our sufficient conditions do not consider such additional structure, an interesting area of future work would be to derive sufficient conditions that utilize the structure on the probability matrices, which would likely produce tighter bounds. It would also be interesting to draw connections to recent work on consistent estimation for populations of networks [61], for which no coupling between samples (layers) exists.

4

Sequential Adaptive Nonlinear Modeling of Time Series

In this chapter, we propose a method for adaptive nonlinear sequential modeling of time series data. Data is modeled as a nonlinear function of past values corrupted by noise, and the underlying non-linear function is assumed to be approximately expandable in a spline basis. We cast the modeling of data as finding a good fit representation in the linear span of multi-dimensional spline basis, and use a variant of l_1-penalty regularization in order to reduce the dimensionality of representation. Using adaptive filtering techniques, we design our online algorithm to automatically tune the underlying parameters based on the minimization of the regularized sequential prediction error. We demonstrate the generality and flexibility of the proposed approach on both synthetic and real-world datasets. Moreover,

we analytically investigate the performance of our algorithm by obtaining both bounds on prediction errors and consistency in variable selection.

4.1 Introduction

Sequentially observed multi-dimensional time series are emerging in various applications. In most of these applications, modeling nonlinear functional inter-dependency between present and past data is crucial for both representation and prediction. In developing a reliable and flexible model that can be widely used in practical scenarios, a practitioner is often faced with these challenges: 1) sequential inference, which means that the inference and prediction at each time step should be made by using only the data before that time, and be updated upon each newly arrived data; in view of that, analysis techniques such as trend-cycle decomposition that lacks such sequential nature may not be applicable; 2) nonlinearity, in which the functional relation is not necessarily linear, and classical linear approaches such as autoregressive-moving-average model [76, 77, 78] may not suffice; 3) adaptivity, which means that the data-generating model is varying over time, and a Kalman filter-type update is desired to track new environments; 4) high dimensionality, which naturally arises when a practitioner seeks to look for more candidate models as sample size grows. For example, environmental science combines high dimensional weather signals for real time prediction [79]. In epidemics, huge amount of online search data is used to form fast prediction of influenza epidemics [80]. In finance, algorithmic traders demand adaptive models to accommodate a fast changing stock market. In robot autonomy, there is the challenge of learning the high dimensional movement systems [81]. These tasks usually take high dimensional input signals which may contain a large number of irrelevant signals. In all these applications, methods to remove redundant signals and learn the nonlinear model with low computational complexity are well sought after. This motivates our work in this paper,

where we propose an approach to sequential nonlinear adaptive modeling of potentially high dimensional time series.

Inference of nonlinear models has been a notoriously difficult problem, especially for large dimensional data[82, 81, 83]. In low dimensional settings, there have been remarkable parametric and nonparametric nonlinear time series models that have been applied successfully to data from various domains. Examples include threshold models[84], generalized autoregressive conditional hetero-scedasticity models[85], multivariate adaptive regression splines (MARS)[82], generalized additive models[86], functional coefficient regression models[87], etc. However, some of these methods may suffer from prohibitive computational complexity. Variable selection using some of these approaches is yet another challenge as they may not guarantee the selection of significant predictors (variables that contribute to the true data generating process) given limited data size. In contrast, there exist high dimensional nonlinear time series models that are mostly inspired by high dimensional statistical methods. There are typically two kinds of approaches. In one approach, a small subset of significant variables is first selected and then nonlinear time series models are applied to selected variables. For example, independence screening techniques such as [88, 89, 90] or the MARS may be used to do variable selection. In another approach, dimension reduction method such as least absolute shrinkage and selection operator (LASSO) [91] are directly applied to nonlinear modeling. Sparse additive models have been developed in recent works of Ravikumar et al. [92] and Huang et al. [83]. In the work of Bazerque et al. [93], splines additive models together with group-sparsity penalty was proposed and applied to spectrum cartography. These offline approaches seem promising and may benefit from additional reductions in computational complexity.

In this work, inspired by the second approach, we develop a new method referred to as Sequential Learning Algorithm for Nonlinear Time Series (SLANTS). A challenging problem in sequential inference is that the data generating process varies with time, which is

common in many practical applications [79, 80, 81]. We propose a method that can help address sequential inference of potentially time-varying models. Moreover, the proposed method provides computational benefits as we avoid repeating batch estimation upon sequential arrival of data. Specifically, we use the spline basis to dynamically approximate the nonlinear functions. The algorithm can efficiently give unequal weights to data points by design, as typical in adaptive filtering. We also develop an online version of group LASSO for dimensionality reduction (i.e. simultaneous estimation and variable selection). To this end, the group LASSO regularization is re-formulated into a recursive estimation problem that produces an estimator close to the maximum likelihood estimator from batch data. We theoretically analyze the performance of SLANTS. Under reasonable assumptions, we also provide an estimation error bound, and a backward stepwise procedure that guarantees consistency in variable selection.

The outline of this chapter is given next. In Section 4.2, we formulate the problem mathematically and present our inference algorithm. In Section 4.3, we present our theoretical results regarding prediction error and model consistency. In Section 4.4, we provide numerical results using both synthetic and real data examples. The results demonstrate excellent performance of our method. We make our conclusions in Section 4.5.

4.2 Sequential modeling of nonlinear time series

In this section, we first present our mathematical model and cast our problem as l_1-regularized linear regression . We then propose an expectation–maximization (EM) type algorithm to sequentially estimate the underlying coefficients. Finally we disclose methods for tuning the underlying parameters. Combining our proposed EM estimation method with automatic parameter tuning, we tailor our algorithm to sequential time series applications.

4.2.1 Formulation of SLANTS

Consider a multi-dimensional time series given by

$$\boldsymbol{X}_t = [X_{1,t}, \ldots, X_{D,t}]^\intercal \in \mathbb{R}^D, t = 1, 2, \ldots$$

Our main objective in this paper is to predict the value of $\boldsymbol{X}_T$ at time T given the past observations $\boldsymbol{X}_{T-1}, \ldots, \boldsymbol{X}_1$. For simplicity, we present our results for the prediction of scalar random variable $X_{1,T+1}$. We start with the general formulation

$$X_{1,T} = f(\boldsymbol{X}_{T-1}, \ldots, \boldsymbol{X}_{T-L}) + \varepsilon_T, \tag{4.1}$$

where $f(\cdot, \cdots, \cdot)$ is smooth (or at least piece-wise smooth), ε_t are independent and identically distributed (i.i.d.) zero mean random variables and the lag order L is a finite but unknown nonnegative integer.

We rewrite the model in (4.1) as

$$\begin{aligned} X_{1,T} = {} & f(X_{1,T-1}, \ldots, X_{1,T-L}, \ldots, X_{D,T-1}, \ldots, X_{D,T-L}) \\ & + \varepsilon_T. \end{aligned}$$

With a slight abuse of notation, we rewrite the above model (4.1) as

$$Y_T = f(X_{1,T}, \ldots, X_{\tilde{D},T}) + \varepsilon_T, \tag{4.2}$$

with observations $Y_T = X_{1,T}$ and $[X_{1,T}, \ldots, X_{\tilde{D},T}] = [X_{1,T-1}, \ldots, X_{1,T-L}, \ldots, X_{D,T-1}, \ldots, X_{D,T-L}]$, where $\tilde{D} = DL$. To estimate $f(\cdot, \cdots, \cdot)$, we consider the following least squares formula-

tion

$$\min_{f} \sum_{t=1}^{T} w_{T,t}(Y_t - f(X_{1,t}, \ldots, X_{\tilde{D},t}))^2 \tag{4.3}$$

where $\{w_{T,t} \in [0,1]\}$ are weights used to emphasize varying influences of the past data. The weights may also be used to accommodate different variance levels across dimensions. The appropriate choice of $\{w_{T,t} \in [0,1]\}$ will be later discussed in Section 4.2.3.

In order to estimate the nonlinear function $f(\cdot, \cdots, \cdot)$, we further assume a nonlinear additive model, i.e.

$$f(X_{1,t}, \ldots, X_{\tilde{D},t}) = \mu + \sum_{i=1}^{\tilde{D}} f_i(X_i), \quad E\{f_i(X_i)\} = 0, \tag{4.4}$$

where f_i are scalar functions, and expectation is with respect to the stationary distribution of X_i. The second condition is required for identifiability. To estimate f_i, we use B-splines (extensions of polynomial regression techniques [94]). In our presentation, we consider the additive model mainly for brevity. Our methods can be extended to models where there exist interactions among $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_{\tilde{D}}$ using multidimensional splines in a straight-forward manner.

We assume that there are v spline basis of degree ℓ for each f_i. Incorporating the B-spline basis into regression, we write

$$\begin{aligned} f_i(x) &= \sum_{j=1}^{v} c_{i,j} b_{i,j}(x), \\ b_{i,j}(x) &= B(x \mid s_{i,1}, \ldots, s_{i,v-\ell+1}) \end{aligned} \tag{4.5}$$

where $s_{i,1}, \ldots, s_{i,v-\ell+1}$ are the knots and $c_{i,j}$ are the coefficients associated with the B-spline

basis. Replacing these into (4.3), the problem of interest is now the minimization of

$$\hat{e}_T = \sum_{t=1}^{T} w_{T,t} \left\{ Y_t - \mu - \sum_{i=1}^{\tilde{D}} \sum_{j=1}^{v} c_{i,j} b_{i,j}(X_{i,t}) \right\}^2 \tag{4.6}$$

over $c_{i,j},\ i = 1, \ldots, \tilde{D}$, $j = 1, \ldots, v$, under the constraint

$$\sum_{t=1}^{T} \sum_{j=1}^{v} c_{i,j} b_{i,j}(x_i) = 0, \text{ for } i = 1, \ldots, L. \tag{4.7}$$

which is the sample analog of the constraint in (4.4). Equivalently, we obtain an unconstrained optimization problem by centering the basis functions. Let $b_{i,j}(x_{i,t})$ be replaced by $b_{i,j}(x_{i,t}) - T^{-1} \sum_{t=1}^{T} b_{i,j}(x_{i,t})$. By proper rearrangement, (4.6) can be rewritten into a linear regression form

$$\hat{e}_T = \sum_{t=1}^{T} w_{T,t} (Y_t - \boldsymbol{z}_t^{\mathrm{T}} \beta_T)^2 \tag{4.8}$$

where β_T is a $(1+\tilde{D}v) \times 1$ column vector to be estimated and $\boldsymbol{z}_t$ is $(1+\tilde{D}v) \times 1$ column vector $\boldsymbol{z}_t = [1, b_{1,1}(x_{1,t}), \ldots, b_{1,v}(x_{1,t}), \ldots, b_{\tilde{D},1}(x_{\tilde{D},t}), \ldots, b_{\tilde{D},v}(x_{\tilde{D},t})]$. Let Z_T be the design matrix of stacking the row vectors $\boldsymbol{z}_t^{\mathrm{T}}$, $t = 1, \ldots, T$. Note that we have used β_T instead of a fixed β to emphasize that β_T may vary with time. We have used bold style for vectors to distinguish them from matrices. Let W_T be the diagonal matrix whose elements are $w_{T,t}$, $t = 1, \ldots, T$. Then the optimal β_T in (4.8) can be recognized as the MLE of the following linear Gaussian model

$$\boldsymbol{Y}_T = Z_T \beta_T + \boldsymbol{\varepsilon} \tag{4.9}$$

where $\varepsilon \in \mathcal{N}(0, W_T^{-1})$. Here, we have used $\mathcal{N}(\mu, V)$ to denote Gaussian distribution with mean μ and covariance matrix V.

To obtain a sharp model from large L, we further assume that the expansion of $f(\cdot, \cdots, \cdot)$ is sparse, i.e., only a few additive components f_i are active. Selecting a sparse model is critical as models of over large dimensions lead to inflated variance, thus compromising the predictive power. To this end, we give independent Laplace priors for each sub-vector of β_T corresponding to each f_i. Our objective now reduces to obtaining the maximum a posteriori estimator (MAP)

$$\begin{aligned} &\log p(\boldsymbol{Y}_T \mid \beta_T, Z_T) - \lambda_T \sum_{i=1}^{\tilde{D}} \|\beta_{T,i}\|_2 \\ &= -\frac{1}{2} \sum_{t=1}^{T} w_{T,t} (Y_t - \boldsymbol{z}_t^T \beta_T)^2 - \lambda_T \sum_{i=1}^{\tilde{D}} \|\beta_{T,i}\|_2 + c \end{aligned} \tag{4.10}$$

where c is a constant that depends only on W_T. The above prior corresponds to the so called group LASSO [95]. The bold $\beta_{T,i}$ is to emphasize that it is not a scalar element of β_T but a sub-vector of it. It will be interesting to consider adaptive group LASSO [96], i.e., to use $\lambda_{T,i}$ instead of a unified λ_T and this is currently being investigated. We refer to [83] for a study of adaptive group LASSO for batch estimation.

4.2.2 Implementation of SLANTS

In order to solve the optimization problem given by (4.10), we build on an EM-based solution originally proposed for wavelet image restoration [97]. This was further applied to online adaptive filtering for sparse linear models [98] and nonlinear models approximated by Volterra series [99]. The basic idea is to decompose the optimization (4.10) into two parts that are easier to solve and iterate between them. One part involves linear updates, and the other involves group LASSO in the form of orthogonal covariance which leads to

closed-form solution.

For now, we assume that the knot sequence $t_{i,1}, \ldots, t_{i,v}$ for each i and v is fixed. Suppose that all the tuning parameters are well-defined. We introduce an auxiliary variable τ_T that we refer to as the *innovation parameter*. This helps us to decompose the problem so that underlying coefficients can be iteratively updated. It also allows the sufficient statistics to be rapidly updated in a sequential manner. The model in (4.9) now can be rewritten as

$$\boldsymbol{Y}_T = Z_T \vartheta_T + W_T^{-\frac{1}{2}} \varepsilon_1, \ \vartheta_T = \beta_T + \tau_T \varepsilon_2,$$

where

$$\varepsilon_1 \in \mathcal{N}(0, I - \tau_T^2 W_T^{\frac{1}{2}} Z_T Z_T^{\intercal} W_T^{\frac{1}{2}}), \quad \varepsilon_2 \in \mathcal{N}(0, I) \tag{4.11}$$

We treat ϑ_T as the missing data, so that an EM algorithm can be derived. By basic calculations similar to that of [97], we obtain the kth step of EM algorithm

E step:

$$Q(\beta \mid \hat{\beta}_T^{(k)}) = -\frac{1}{2\tau_T^2} \|\beta - \boldsymbol{r}^{(k)}\|_2^2 - \lambda_T \sum_{i=1}^{\tilde{D}} \|\beta_i\|_2 \tag{4.12}$$

where

$$\boldsymbol{r}^{(k)} = (I - \tau_T^2 \mathcal{A}_T) \hat{\beta}_T^{(k)} + \tau_T^2 B_T, \tag{4.13}$$

$$\mathcal{A}_T = Z_T^{\intercal} W_T Z_T, \ \boldsymbol{B}_T = Z_T^{\intercal} W_T \boldsymbol{Y}_T. \tag{4.14}$$

The derivation of Equation (4.12) is included in the appendix.

M step: $\hat{\beta}_T^{(k+1)}$ is the maximum of $Q(\beta \mid \hat{\beta}_T^{(k)})$ given by

$$\hat{\beta}_{T,i}^{(k+1)} = \left[1 - \frac{\lambda_T \tau_T^2}{\|\boldsymbol{r}_i^{(k)}\|_2}\right]_+ \boldsymbol{r}_i^{(k)}, \quad i = 1, \ldots, \tilde{D}. \tag{4.15}$$

Suppose that we have obtained the estimator $\hat{\beta}_T$ at time step T. Consider the arrival of the $(T+1)$th point $(y_{T+1}, \boldsymbol{z}_{T+1})$, respectively corresponding to the response and covariates of time step $T+1$. We first compute $\boldsymbol{r}_{T+1}^{(\circ)}$, the initial value of $\boldsymbol{r}$ to be input the EM at time step $T+1$:

$$\boldsymbol{r}_{T+1}^{(\circ)} = (I - \tau_T^2 \mathcal{A}_{T+1})\hat{\beta}_T + \tau_T^2 \boldsymbol{B}_{T+1}, \tag{4.16}$$

where

$$\begin{aligned} \mathcal{A}_{T+1} &= (1 - \gamma_{T+1})\mathcal{A}_T + \gamma_{T+1} \boldsymbol{z}_{T+1} \boldsymbol{z}_{T+1}^{\mathrm{T}}, \\ \boldsymbol{B}_{T+1} &= (1 - \gamma_{T+1})\boldsymbol{B}_T + \gamma_{T+1} y_{T+1} \boldsymbol{z}_{T+1}. \end{aligned} \tag{4.17}$$

Then we run the above EM for $K > 0$ iterations to obtain an updated $\hat{\beta}_{T+1}$.

Remark 9 *In the above equation, $\{\gamma_t\}$ is a nonnegative sequence which we refer to as the step sizes. We shall elaborate on its relation with $\{W_t\}$ in Subsection 4.2.3.*

SLANTS can be efficiently implemented. The recursive computation of $\mathcal{A}_T$ (resp. B_T) reduces the complexity from $O(\tilde{D}^3)$ to $O(\tilde{D}^2)$ (resp. from $O(\tilde{D}^2)$ to $O(\tilde{D})$). Moreover, straightforward computations indicate that the complexity of SLANTS at each time t is $O(\tilde{D}^2)$, which does not depend on T. Coordinate descent [100] is perhaps the most widely used algorithm for batch LASSO. Adapting coordinate descent to sequential setting has the same complexity for updating sufficient statistics. But straightforward use of batch LASSO has complexity $O(\tilde{D}^2 T)$.

Theorem 6 *At each iteration, the mapping from $\hat{\beta}_T^{(k)}$ to $\hat{\beta}_T^{(k+1)}$ is a contraction mapping for any τ_T, whenever the absolute values of all eigenvalues of $I - \tau_T^2 A_{T+1}$ are less than one. In addition, there exists a unique global maximum point of (4.10) denoted by $\hat{\beta}_T$, and the error $\|\hat{\beta}_T^{(k+1)} - \hat{\beta}_T\|_2$ decays exponentially in k.*

Remark 10 *The theorem states that EM can converge exponentially fast to the MAP of (4.10). From its assumption, it can be directly calculated that (4.10) as a function of β_T is strictly concave. We note that the assumption is not mild, so the application of Theorem 1 is limited. But the proposed algorithm does converge exponentially fast in our various synthetic and real data experiments. The proof of Theorem 6 is given in the appendix C.*

4.2.3 The choice of tuning parameters: from a prequential perspective

To evaluate the predictive power of an inferential model estimated from all the currently available data, ideally we would apply it to independent and identically generated datasets. However, it is not realistic to apply this cross-validation idea to real-world time series data, since real data is not permutable and has a "once in a lifetime" nature. As an alternative, we adopt a prequential perspective [101] that the goodness of a sequential predictive model shall be assessed by its forecasting ability.

Specifically, we evaluate the model in terms of the one-step prediction errors upon each newly arrived data point and subsequently tune the necessary control parameters, including regularization parameter λ_t and innovation parameter τ_t (see details below). Automatic tuning of the control parameters are almost a necessity in many real-world applications in which any theoretical guidance (e.g., our Theorem 2) may be insufficient or unrealistic. Throughout our algorithmic design, we have adhered to the prequential principle and implemented the following strategies.

The choice of $w_{T,t}$: In view of equation (4.17), $w_{T,t}$ is determined by $w_{1,1} = \gamma_1$, and

$$w_{t,t} = \gamma_t,\ w_{t,j} = w_{t-1,j}(1-\gamma_t), \quad j = 1, \ldots, t-1,$$

for $t > 1$.

It includes two special cases that have been commonly used in the literature. The first case is $\gamma_t = 1/t$. It is easy to verify that $w_{T,t} = 1/T, t = 1, \ldots, T$ for any T. This leads to the usual least squares. The second case is $\gamma_t = c$ where c is a positive constant. It gives $w_{T,t} = c(1-c)^{T-t}, t = 1, \ldots, T$. From (4.3), the estimator of f remains unchanged by rescaling $w_{T,t}$ by $1/c$, i.e. $w_{T,t} = (1-c)^{T-t}$ which is a series of powers of $1-c$. The value $1-c$ has been called the "forgetting factor" in the signal processing literature and used to achieve adaptive filtering [98].

The choice of τ_T: Because the optimization problem

$$\log p(\boldsymbol{Y}_T \mid \beta_T) - \lambda_T \sum_{i=1}^{L} \|\beta_{T,i}\|_2 \tag{4.18}$$

is convex, as long as τ_T is proper, the EM algorithm converges to the optimum regardless of what τ_T is. But τ_T affects the speed of convergence of EM as $\lambda_T \tau_T^2$ determines how fast β_T shrinks. Intuitively the larger τ_T is, the faster is the convergence. Therefore we prefer τ_T to be large and proper. A necessary condition for τ_T to be proper is to ensure that the covariance matrix of ε_1 in

$$\varepsilon_1 \in \mathcal{N}(0, I - \tau_T^2 W^{\frac{1}{2}} Z_T Z_T^{\mathrm{T}} W^{\frac{1}{2}}), \quad \varepsilon_2 \in \mathcal{N}(0, I) \tag{4.19}$$

is positive definite. Therefore, there is an upper bound $\bar{\tau}_T$ for τ_T, and $\bar{\tau}_T$ converges to a positive constant $\bar{\tau}$ under some mild assumptions (e.g. the stochastic process X_t is sta-

tionary). Extensive experiments have shown that $\bar{\tau}_T/2$ produces satisfying results in terms of model fitting. However, it is not computationally efficient to calculate $\bar{\tau}_T$ at each T in SLANTS. Nevertheless without computing $\bar{\tau}_T$, we can determine if $\tau_T < \bar{\tau}_T$ by checking the EM convergence. If τ_T exceeds $\bar{\tau}_T$, the EM would diverge and coefficients go to infinity exponentially fast. This can be proved via a similar argument to that of proof of Theorem 1. This motivates a lazy update of τ_T with shrinkage only if EM starts to diverge.

The choice of λ_T: On the choice of regularization parameter λ_T, different methods have been proposed in the literature. The common way is to estimate the batch data for a range of different λ_T's, and select the one with minimum cross-validation error. To reduce the underlying massive computation required for such an approach, in the context of Bayesian LASSO [102], [103] proposed an sequential Monte Carlo (SMC) based strategy to efficiently implement cross-validation. The main proposal is to treat the posterior distributions educed by an ordered sequence of λ_T as $\pi_t, t = 0, 1, \ldots$, the target distributions in SMC, and thus avoid the massive computation of applying Markov chain Monte Carlo (MCMC) for each λ independently. Another method is to estimate the hyper-parameter λ_T via empirical Bayes method [102]. In our context, however, it is not clear whether the Bayesian setting with MCMC strategy can be efficient, as the dimension Lv can be very large. An effective implementation technique is to run three channels of our sequential modeling, corresponding to $\lambda_T^- = \lambda_T/\delta, \lambda_T, \lambda_T^+ = \lambda_T * \delta$, where $\delta > 1$ is a small step size. The one with minimum average prediction error over the latest window of data was chosen as the new λ_T. For example, if λ_T^- gives better performance, let the three channels be $\lambda_T^-/\delta, \lambda_T^-, \lambda_T^- * \delta$. If there is an underlying optimal λ^* which does not depend on T, we would like our channels to converge to the optimal λ^* by gradually shrinking the stepsize δ. Specifically in case that the forgetting factor $\gamma_t = 1/t$, we let $\delta_T = 1 + \frac{1}{T}(\delta - 1)$ so that the step size $\delta_T \to 1$ at the same speed as weight of new data.

The choice of knots: The main difficulty in applying spline approximation is in deter-

mining the number of the knots to use and where they should be placed. Jupp [104] has shown that the data can be fit better with splines if the knots are free variables. de Boor suggests the spacing between knots is decreased in proportion to the curvature (second derivative) of the data. It has been shown that for a wide class of stationary process, the number of knots should be of the order of $O(T^{\zeta})$ for available sample size T and some positive constant ζ to achieve a satisfying rate of convergence of the estimated nonlinear function to the underlying truth (if it exists) [105]. Nevertheless, under some assumptions, we will show in Theorem 7 that the prediction error can be upper bounded by an arbitrarily small number (which depends on the specified number of knots). It is therefore possible to identify the correct nonzero additive components in the sequential setting. On the other hand, using a fixed number of knots is computationally desirable because sharp selection of significant spline basis/support in a potentially varying environment is computationally intensive. It has been observed in our synthetic data experiments that the variable selection results are not very sensitive to the number of knots as long as this number is moderately large (e.g. around $v = 10$).

4.3 Theoretical results

Consider the harmonic step size $\gamma_t = 1/t$. For now assume that the sequential update at each time t produces $\hat{\beta}_t$ that is the same as the penalized least squares estimator given batch data. We are interested in two questions. First, how to extend the current algorithm in order to take into account an ever-increasing number of dimensions? Second, is it possible to select the "correct" nonzero components as sample size increases?

The first question is important in practice as any prescribed finite number of dimensions/time series may not contain the data-generating process, and it is natural to consider more candidates whenever more samples are obtained. It is directly related to the widely

studied high-dimensional regression for batch data. In the second question, we are not only interested in optimizing the prediction error but also to obtain a consistent selection of the true nonzero components. Moreover, in order to maintain low complexity of the algorithm, we aim to achieve the above goal by using a fixed number of spline basis. We thus consider the following setup. Recall the predictive model (4.1) and its alternative form (4.2). We assume that L is fixed while D is increasing with sample size T at certain rate.

Following the setup of [106], we suppose that each X_d takes values from a compact interval $[a, b]$. Let $[a, b]$ be partitioned into J equal-sized intervals $\{I_j\}_{j=1}^{J}$, and let $\mathcal{F}$ denote the space of polynomial splines of degree $\ell \geq 1$ consisting of functions $g(\cdot)$ satisfying 1) the restriction of $g(\cdot)$ to each interval is a polynomial of degree ℓ, and 2) $g(\cdot) \in C^{\ell-1}[a, b]$ ($\ell - 1$ times continuously differentiable). Typically, splines are called linear, quadratic or cubic splines accordingly as $\ell = 1, 2$, or 3. There exists a normalized B-spline basis $\{b_j\}_{j=1}^{v}$ for $\mathcal{F}$, where $v = J + \ell$, and any $f_i(x) \in \mathcal{F}$ can be written in the form of (4.5). Let $k \leq \ell$ be a nonnegative integer, $\beta \in (0, 1]$ that $p = k + \beta > 0.5$, and $\mathcal{M} > 0$. Suppose each considered (non)linear function f has kth derivative, $f^{(k)}$, and satisfies the Holder condition with exponent β: $|f^{(k)}(x) - f^{(k)}(x')| < \mathcal{M}|x - x'|$ for $x, x' \in [a, b]$. Define the norm $\|f\|_2 = \sqrt{\int_a^b f(x)^2 dx}$. Let $f^* \in \mathcal{F}$ be the best L_2 spline approximation of f. Standard results on splines imply that $\|f_d - f_d^*\|_\infty = O(v^{-p})$ for each d. The spline approximation is usually an estimation under a mis-specified model class (unless the data-generating function is low-degree polynomials), and large v narrows the distance to the true model. We will show that for large enough v, it is possible to achieve the aforementioned two goals. To make the problem concrete, we need the following assumptions on the data-generating procedure.

Assumption 1 *The number of additive components is finite and will be included into the candidate set in finite time steps. In other words, there exists a "significant" variable set* $S_0 = \{i_1, \ldots, i_{D_0}\}$ *such that 1)* $f_d(x) \neq 0$ *for each* $d \in S_0$*, 2)* $f_d(x) \equiv 0$ *for* $d \notin S_0$*, and 3) both* D_0

and i_{D_0} are finite integers that do not depend on sample size T.

We propose two steps for a practitioner targeting two goals given below.

Step 1. (unbiasedness) This step aims to discover the significant variable set with probability close to one as more data is collected. The approach is to minimize the objective function in (4.10), and it can be efficiently implemented using the proposed sequential algorithm in Section 4.2.2 with negligible error (Theorem 6). In the case of equal weights $w_{T,t} = 1/T$, it can be rewritten as

$$\|Y_T - Z_T\beta_T\|_2^2 + \tilde{\lambda}_T \sum_{i=1}^{\tilde{D}} \|\beta_{T,i}\|_2 \tag{4.20}$$

where $\tilde{\lambda}_T = 2T\lambda_T$. Due to Assumption 1, the significant variable set S_0 is included in the candidate set $\{1, \ldots, \tilde{D}\}$ for sufficiently large T. Our selected variables are those whose group coefficients are nonzero, i.e. $S_1 = \{d : 1 \leq d \leq \tilde{D}, \hat{\beta}_{T,d} \neq \mathbf{0}\}$. We are going to prove that all the significant variables will be selected by minimizing (4.20) with appropriately chosen $\tilde{\lambda}_T$, i.e., $S_0 \subseteq S_1$.

Step 2. (minimal variance) The second step is optional and it is applied only when a practitioner's goal is to avoid selecting any redundant variables outside S_0. Suppose that we obtain a candidate set of $\tilde{D}$ variables S_1 (satisfying $S_0 \subseteq S_1$ from the previous step). Since a thorough search over all subsets of variables is computationally demanding, we use a backward stepwise procedure. We start with the set of selected variables S_1, delete one variable at a time by minimizing the MSE of a spline model with $v_T = T$ number of equally spaced knots. We note that v_T in the optional Step 2 can be different from the v in SLANTS. Specifically, suppose that at step k ($k = 1, 2, \ldots$), the survived candidate models

are indexed by $\mathcal{S}^{(k)}$. We solve the least-squares problem for each $\bar{d} \in \mathcal{S}^{(k)}$

$$\hat{e}_{\bar{d}}^{(k)} = \min_{,c_{d,j}} \sum_{t=1}^{T} \left(Y_t - \mu - \sum_{d \in \mathcal{S}} \sum_{j=1}^{v_T} c_{d,j} b_{d,j}(X_{d,t}) \right)^2 \tag{4.21}$$

where $\mathcal{S} = \mathcal{S}^{(k-1)} - \{\bar{d}\}$, and select $\bar{d} = \bar{d}_k^*$ that minimize the $\hat{e}_{\bar{d}}^{(k)}$ with minimum denoted by $\hat{e}^{(k)}$. Here $\mathcal{A} - B$ denotes the set of elements that are in a set $\mathcal{A}$ but not in a set B. We let $\mathcal{S}^{(k)} = \mathcal{S}^{(k-1)} - \{\bar{d}_k^*\}$. By default, we let $\mathcal{S}^{(0)} = S_1$ and use $\hat{e}^{(0)}$ to denote the minimum of (4.21) with $\mathcal{S} = S_1$. If $\hat{e}^{(k-1)} - \hat{e}^{(k)} < (v_T \log T)/T$, i.e., the gain of goodness of fit is less than the incremented Bayesian information criterion (BIC) penalty [107], then we stop the procedure and output $S_2 = \mathcal{S}^{(k-1)}$; otherwise we proceed to the $(k+1)$th iteration. We prove that the finally selected subset S_2 satisfies $\lim_{T\to\infty} P(S_2 = S_0) = 1$.

Before we proceed to the theoretical result, we introduce some necessary assumptions and their interpretations.

Assumption 2 *There is a positive constant c_0 such that* $\min_{d \in S_0} \|f_d\|_2 \geq c_0$.

Assumption 3 *The noises ε_t are sub-Gaussian distributed, i.e., $E(e^{w\varepsilon_t}) \leq e^{w^2 \sigma^2/2}$ for a constant $\sigma > 0$ and any $w \in \mathbb{R}$.*

Assumption 4 *Suppose that S_1 is a finite subset of $\{1, \ldots, \tilde{D}\}$. In addition, the "design matrix" Z_{S_1} satisfies $Z_{S_1}^\tau Z_{S_1}/T \geq \varkappa$ for a positive constant $\varkappa$ that depend only on v (the number of splines).*

We use $o_p(1)$ and $O_p(1)$ to denote a sequence of random variables that converges in probability to zero, and that is stochastically bounded, respectively. We use $O(1)$ to denote a bounded deterministic sequence.

Theorem 7 *Suppose that Assumptions 1-4 hold. Then for any given v it holds that*

$$\begin{aligned}\|\beta_{\tilde{S}_1} - \hat{\beta}_{\tilde{S}_1}\|_2^2 \leq & 8c_2 v^{-2p}/\varkappa + O_p(T^{-1}\log\tilde{D}) + \\ & O_p(T^{-1}) + O(T^{-2}\tilde{\lambda}^2) \end{aligned} \tag{4.22}$$

for some positive constant c_2. If we further assume that $\log\tilde{D} = o(T)$, $\tilde{\lambda} = o(T)$, then there exists a constant $c_1 > 0$ such that for all $v > c_1 c_0^{-1/p}\max\{1, c_0^{-\frac{1}{p(2p+1)}}\}$, $\lim_{T\to\infty} P(S_0 \subseteq S_1) = 1$.

Remark 11 *Theorem 7 gives an error bound between the estimated spline coefficients with the oracle, where the first term is dominating. As a result, if v is sufficiently large, then it is guaranteed that S_0 will be selected with probability close to one. We note that the constant c_1 depends only on the true nonlinear function and the selected spline basis function. In proving Theorem 7, Assumption 2-3 serve as standard conditions to ensure that a significant variable is distinguishable, and that any tail probability could be well bounded. Assumption 4 is needed to guarantee that if the estimated coefficients $\hat{\beta}$ produces low prediction errors, then it is also close to the true (oracle) coefficients. This assumption is usually guaranteed by requiring $\tilde{\lambda} > c\sqrt{T\log D}$. See for example [83, 108].*

To prove the consistency in step 2, we also need the following assumption (which further requires that the joint process is strictly stationary and strongly mixing).

Assumption 5 $\sup_x\{E(|Y_t|^r|\boldsymbol{X}_t = x)\} < \infty$ *for some* $r > 2$.

The α-mixing coefficient is defined as $\alpha_S(j) = \sup\{P(E_y \cap E_x) - P(E_y)P(E_x) : E_y \in \sigma(\{(Y_{\tilde{t}}, X_{d,\tilde{t}}, d \in S) : \tilde{t} \leq n\}), E_x \in \sigma(\{(Y_{\tilde{t}}, X_{d,\tilde{t}}, d \in S) : \tilde{t} \geq n + j\})\}$, where $\sigma(\cdot)$ denotes the σ-field generated by the random variables inside the parenthesis.

Assumption 6 *The process* $\{(X_{d,t}, d \in S_1)\}$ *is strictly stationary, and the joint process* $\{(Y_t, X_{d,t}, d \in S_1)\}$ *is* α*-mixing with coefficient*

$$\alpha_{S_1}(j) \leq \min\{O(j^{-2.5 \quad /(1- \quad)}), O(j^{-2r/(r-2)})\},$$

where ζ *has been defined in Step 2.*

Theorem 8 *Suppose that Assumptions 1-6 hold, then the* S_2 *produced by the above step 2 satisfies* $\lim_{T\to\infty} P(S_2 = S_0) = 1$.

4.4 Numerical results

In this section, we present experimental results to demonstrate the theoretical results and the advantages of SLANTS on both synthetic and real-world datasets. The synthetic experiments include cases where the data-generating model is fixed over time, is varying over time, or involves large dimensionality.

4.4.1 Synthetic data experiment: modeling nonlinear relation in stationary environment

The purpose of this experiment is to show the performance of SLANTS in stationary environment where the data-generating model is fixed over time. We generated synthetic data using the following nonlinear model

$$\begin{aligned} X_{1,t} &= \varepsilon_{1,t} \\ X_{2,t} &= 0.5X_{1,t-1}^2 - 0.8X_{1,t-7} + 0.2\varepsilon_{2,t},\ t = 1, \ldots, 500 \end{aligned}$$

where $\varepsilon_{1,t}$ and $\varepsilon_{2,t}$ are i.i.d. standard Gaussian. The goal is to model/forecast the series $X_{2,t}$. We choose $L = 8$, and place $v = 10$ quadratic splines in each dimension. The knots are

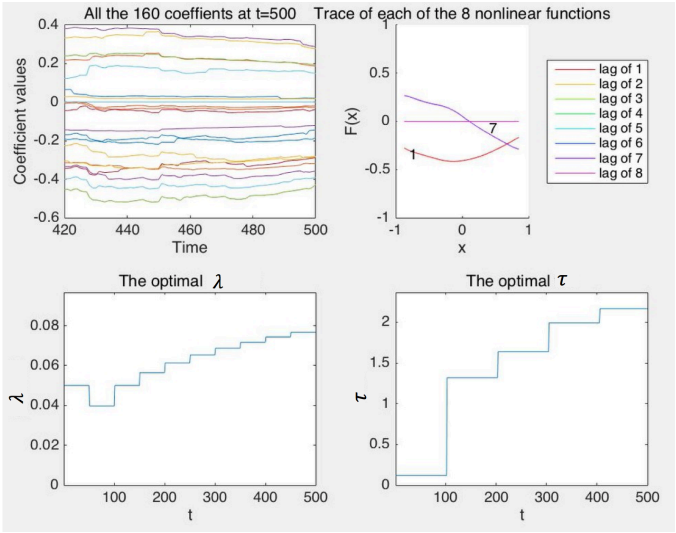


Figure 4.1: Four subplots show the estimated coefficients of splines, nonlinear functions, and trace plots of automatically-tuned regularization parameter λ_t and innovation parameter τ_t.

equally spaced between the 0.01 and 0.99 quantiles of observed data. The initial L values of $X_{2,t}$ are set to zeros. We choose the step size $\gamma_t = 1/t$ to ensure convergence.

Simulation results are summarized in Fig. 4.1. The left-top plot shows the convergence of all the $2 \times 8 \times 10 = 160$ spline coefficients. The right-top plot shows how the eight nonlinear components f_d, $d = 1, \ldots, 8$ evolve, where the number 1-8 indicate each additive component (splines). The values of each function are centralized to zero for identifiability. The remaining two plots show the optimal choice of control parameters λ_t and τ_t that have been automatically tuned over time. In the experiment, the active components f_1 and f_7 are correctly selected and well estimated. It is remarkable that the convergence is mostly achieved after only a few incoming points (less than the number of coefficients 160).

4.4.2 Synthetic data experiment: modeling nonlinear relation in adaptive environment

The purpose of this experiment is to show the performance of SLANTS in terms of prediction and nonlinearity identification when the underlying date generating model varies over time.

We have generated a synthetic data using the following nonlinear model where there is a change at time $t = 500$,

$$
\begin{aligned}
X_{1,t} &= \varepsilon_{1,t} \\
X_{2,t} &= 0.5X_{1,t-1}^2 - 0.8X_{1,t-7} + 0.2\varepsilon_{2,t},\ t = 1, \ldots, 500 \\
X_{1,t} &= u_{1,t} \\
X_{2,t} &= -2X_{1,t-1}^2 + \exp(X_{1,t-7}) + 0.2\varepsilon_{2,t},\ t = 501, \ldots, 1000
\end{aligned}
$$

where $\varepsilon_{1,t}$ and $\varepsilon_{2,t}$ are i.i.d. standard Gaussian. $u_{1,t}$ are i.i.d. uniform on $[-1, 1]$. The goal is to model the series $X_{2,t}$. Compared with the previous experiment, the only difference is that the forgetting factor is set to $\gamma = 0.99$ in order to track potential changes in the underlying true model. Fig. 4.2 shows that SLANTS successfully tracked a change after the change point $t = 500$. The top plot in Fig. 4.2 shows the inference results right before the change. It successfully recovers the quadratic pattern of lag 1 and linear effect of lag 7. The bottom plot in Fig. 4.2 shows the inference results at $t = 1000$. It successfully finds the exponential curve of lag 7 and reversed sign of the quadratic curve of lag 1. From the bottom left subplot we can see how the autotuning regularization parameter decreases since the change point $t = 500$.

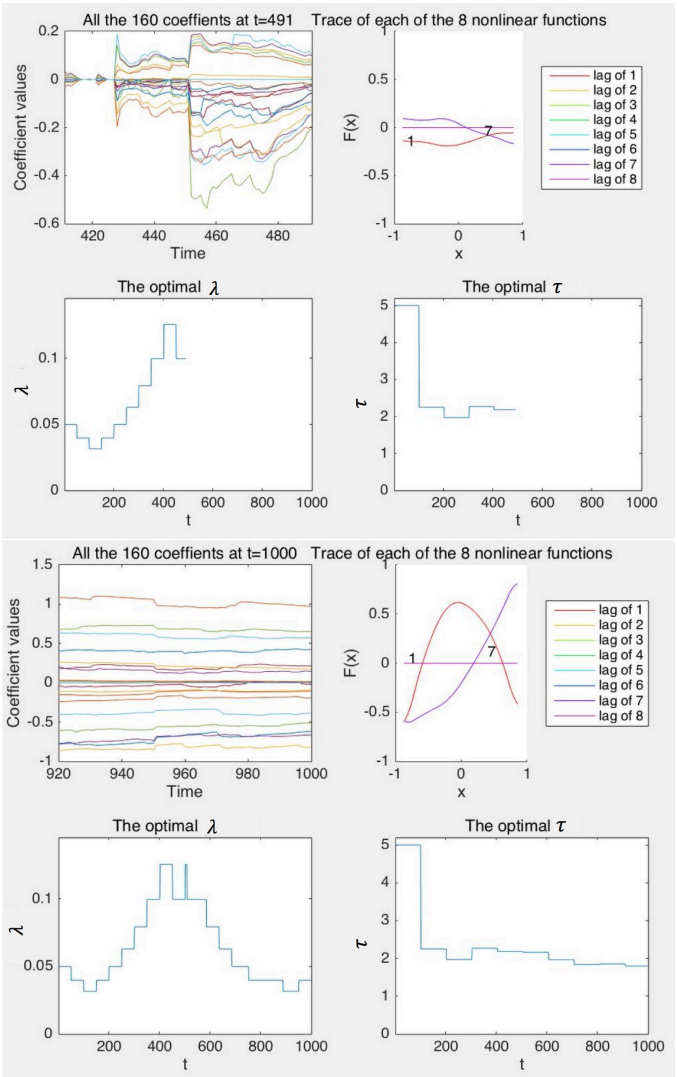


Figure 4.2: Two plots stacked vertically, each consisting of four subplots that show the estimated coefficients of splines, nonlinear functions, and trace plots of automatically-tuned regularization parameter λ_t and innovation parameter τ_t at time $t = 491$ and $t = 1000$ respectively.

4.4.3 Synthetic data experiment: causal discovery for multi-dimensional time series

The purpose of this experiment is to show the performance of SLANTS in identifying nonlinear functional relation (thus Granger-type of causality) among multi-dimensional time series. We have generated a 9-dimensional time series using the following nonlinear network model,

$$
\begin{aligned}
X_{1,t} &= \varepsilon_{1,t} \\
X_{2,t} &= 0.6X_{3,t-1} + \varepsilon_{2,t} \\
X_{3,t} &= 0.3X_{4,t-2}^2 + \varepsilon_{3,t} \\
X_{4,t} &= 0.7X_{5,t-1} - 0.2X_{5,t-2} + \varepsilon_{4,t} \\
X_{5,t} &= -0.2X_{2,t-1}^2 + \varepsilon_{5,t} \\
X_{6,t} &= 0.5X_{6,t-2} + 1 + \varepsilon_{6,t} \\
X_{7,t} &= 2\exp(-X_{7,t-2}^2) + \varepsilon_{7,t} \\
X_{8,t} &= 6X_{7,t-1} - 5X_{9,t-2} + \varepsilon_{8,t} \\
X_{9,t} &= -X_{6,t-1} + 0.9X_{7,t-2} + \varepsilon_{9,t}
\end{aligned}
$$

where $\varepsilon_{1,t}$ and $\varepsilon_{2,t}$ are i.i.d. standard Gaussian. The initial L values are set to zero. The goal is to model each dimension and draw sequential causality graph based on the estimation. We choose $L = 2$, $v = 10$ and $\gamma_t = 1/t$. For illustration purpose, we only show the estimation for $X_{9,t}$. The left-top plot shows the 9 dimensional raw data that are sequentially obtained. The right-top plot shows the convergence of the $DLv = 9 \times 2 \times 10 = 180$ coefficients in modeling $X_{9,t}$. The right-bottom plot shows how the nonlinear components $f : X_{6,t-1} \mapsto X_{9,t}$ and $f : X_{7,t-2} \mapsto X_{9,t}$ evolve. Similar as before, the values of each

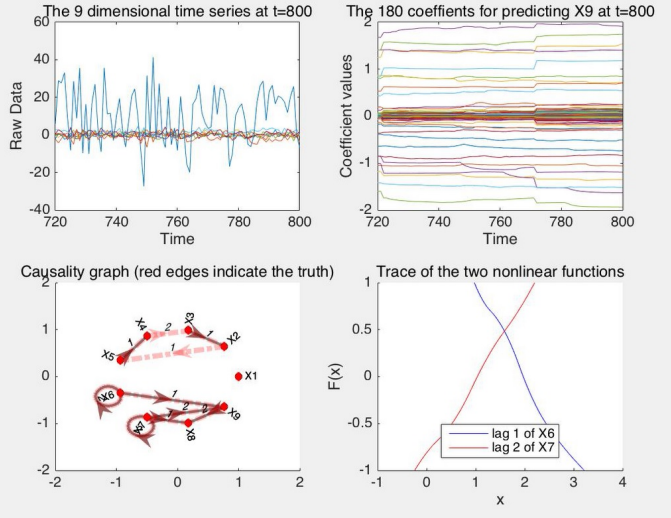


Figure 4.3: Four subplots show the time series data, convergence of the coefficients, causality graph, and trace plot of the nonlinear functions. A demo video is available in the supplement.

function are centralized to zero for identifiability. The left-bottom plot shows the causality graph, which is the digraph with black directed edges and edge labels indicating functional relations. For example, in modeling $X_{9,t}$, if the function component corresponding to $X_{6,t-1}$ is nonzero, then we draw a directed edge from 6 to 9 with label 1; if the function components corresponding to both $X_{6,t-1}$ and $X_{6,t-2}$ are nonzero, then we draw a directed edge from 6 to 9 with label 12. The true causality graph (determined by the above data generating process) is drawn as well, in red thick edges. From the simulation, the discovered causality graph quickly gets close to the truth.

4.4.4 Synthetic data experiment: computational cost

The purpose of this experiment is to show that SLANTS is computationally efficient by comparing it with standard batch group LASSO algorithm. We use the same data generating process in the first synthetic data experiment, and let the size of data be $T = 100, 200, \ldots, 1000$.

We compare SLANTS with the standard R package "grplasso" [109] and "gglasso"[110]

which implement widely used group LASSO algorithms. The package "gglasso" implements the efficient active-set algorithm proposed in [111]. For the two packages, at each time t, solution paths on a fixed grid of 100 penalties are calculated. To provide fair comparisons, we run SLANTS in two ways. The first is the proposed algorithm with adaptive tuned penalties. In the table, it is denoted as SLANTS(a). The second is SLANTS without adaptive tuning but also run on a fixed grid of 100 equivalent penalties as in "grplasso" and "gglasso", denoted as SLANTS(b). In computing solution paths, we adopted the techniques suggested in [111]. The results are shown in Table 4.1.

Table 4.1 shows the time in seconds for SLANTS(a), SLANTS(b), gglasso, and grplasso to run through a dataset sequentially with different size T. Each run is repeated 30 times and the standard error of running time is shown in parenthesis. From Table 4.1, the computational cost of SLANTS grows linearly with T while gglasso and grplasso grow much faster. Moreover, the prediction error is very similar for SLANTS(b), gglasso and grplasso on the grid of penalties. This is understandable as they calculate the solution to the same optimization problem. SLANTS(a) approaches the optimal prediction error as the penalty parameter is stabilized. But SLANTS(a) is faster than SLANTS(b) as it only calculates solutions to three penalties at each time. In summary, both SLANTS(a) and SLANTS(b) are computationally faster than existing batch algorithms with comparable prediction performance.

The computational cost of SLANTS is slightly larger than that of grplasso when $T < 100$. This is because SLANTS is written purely in R, while the core part of gglasso and grplasso is implemented in Fortran (which is usually a magnitude faster than R). However, the growth of computational cost of SLANTS is much slower than that of grplasso, and thus SLANTS is faster for large T.

Table 4.1: The table shows the computational cost in seconds with standard error in parenthesis for SLANTS(a), SLANTS(b), gglasso, and grplasso, with increasing T.

T	SLANTS(a)	SLANTS(b)	gglasso	grplasso
100	4.8(0.1)	35.5(2.2)	32.6(2.5)	9.9(3.6)
200	11.1(0.3)	82.5(3.8)	110.8(7.9)	98.3(9.3)
300	15.4(0.7)	131.4(5.6)	204.3(9.2)	238.6(16.7)
400	21.4(0.7)	180.3(7.2)	296.2(10.7)	392.3(21.4)
500	26.0(0.9)	228.8(9.0)	386.8(12.1)	563.2(26.3)
600	31.3(1.1)	277.0(10.8)	477.5(13.4)	753.3(30.6)
700	37.1(1.2)	324.8(12.7)	569.4(15.0)	961.3(34.6)
800	42.1(1.4)	372.3(14.5)	663.0(19.1)	1189.0(38.5)
900	46.3(1.6)	419.4(16.3)	758.6(20.4)	1435.7(43.3)
1000	53.3(1.8)	466.3(18.1)	856.5(21.3)	1702.5(46.8)

4.4.5 Real data experiment: Boston weather data from 1980 to 1986

In this experiment, we study the daily Boston weather data from 1980 Jan to 1986 Dec. with $T = 2557$ points in total. The data is a six-dimensional time series, with each dimension corresponding respectively to temperature (K), relative humidity (%), east-west wind (m/s), north-south wind (m/s), sea level pressure (Pa), and precipitation (mm/day). In other words, the raw data is in the form of $X_{d,t}, d = 1, \ldots, 6, t = 1, \ldots, T$. We plot the raw data corresponding to year 1980 (i.e. $X_{d,t}, d = 1, \ldots, 6, t = 1, \ldots, 366$) in Fig. 4.4.

We compare the predictive performance of SLANTS with that of a linear model. For brevity, suppose that we are going to predict the east-west wind. We chose the autoregressive model of order 3 (denoted by AR(3)) as the representative linear model. The order was chosen by applying *Bridge criterion* [112] to the batch data of T observations. We started processing the data from $t_0 = 10$, and for each $t = t_0 + 1, \ldots, T$ the one-step ahead prediction error $\hat{e}_t$ was made by applying AR(3) and SLANTS to the currently available $t - 1$ observations. The cumulated average prediction error at time step t is computed to be $\sum_{i=t_0+1}^{t} \hat{e}_i/(t - t_0)$, where $\hat{e}_i$ is the squared difference between the true observation and

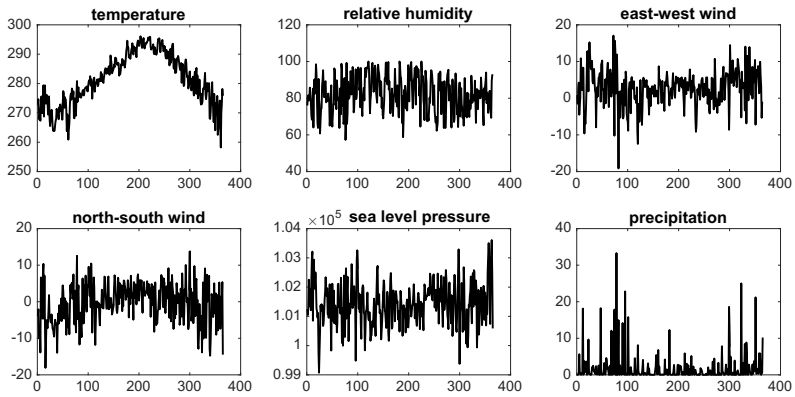


Figure 4.4: A graph showing the raw data of (a) temperature (K), (b) relative humidity (%), (c) east-west wind (m/s), (d) north-south wind (m/s), (e) sea level pressure (Pa), and (f) precipitation (mm/day).

our prediction at time step i. The results are shown in Fig. 4.5(a). At the last time step, the significant (nonzero) functional components are the third, fourth, and sixth dimension, corresponding to EW wind, NS wind, precipitation, have been plotted in Fig. 4.5 (b), (c), (d), respectively. From the plot, the marginal effect of $X_{4,t}$ on $X_{3,t+1}$ is clearly nonlinear. It seems that the correlation is low for $X_{4,t} < 0$ and high for $X_{4,t} > 0$. In fact, if we let $\mathcal{T} = \{t : X_{4,t} > 0\}$, the correlation of $\{X_{4,t} : t \in \mathcal{T}\}$ with $\{X_{3,t+1} : t \in \mathcal{T}\}$ is 0.25 (with p value 1.4×10^{-8}) while $\{X_{4,t} : t \notin \mathcal{T}\}$ with $\{X_{3,t+1} : t \notin \mathcal{T}\}$ is -0.05 (with p value 0.24)

4.4.6 Real data experiment: the weekly unemployment data from 1996 to 2015

In this experiment, we study the US weekly unemployment initial claims from Jan 1996 to Dec 2015. The data is a one-dimensional time series with $T = 1043$ points in total. we plot the raw data in Fig. 4.6.

Though the data exhibits strong cyclic pattern, it may be difficult to perform cycle-trend decomposition in a sequential setting. We explore the power of SLANTS to do lag selection to compensate the lack of such tools.

We compare three models. The first model, AR(5), is linear autoregression with lag order

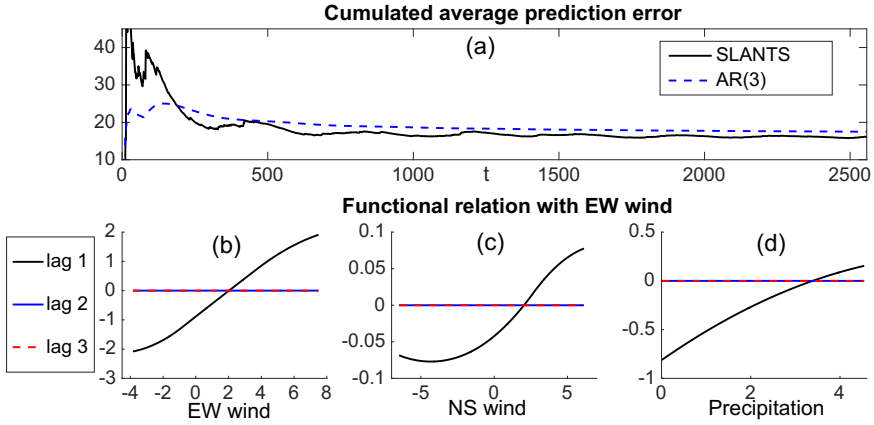


Figure 4.5: A graph showing (a) the cumulated average one-step ahead prediction error of east-west wind (m/s) produced by two approaches, and east-west wind decomposed into nonlinear functions of lagged values of (b) east-west wind, (c) north-south wind (m/s), and (c) precipitation (mm/day). The functions were output from SLANTS at the last time step $t = T$.

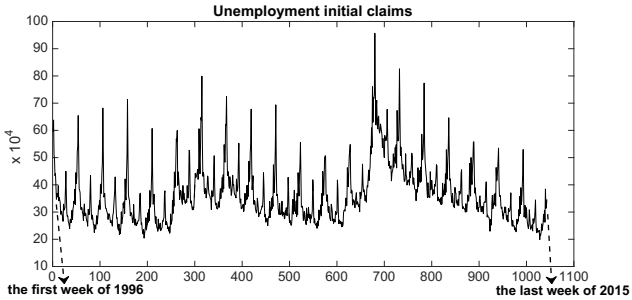


Figure 4.6: A graph showing the raw data of the number of unemployment initial claims.

5. The lag order was chosen by applying Bridge criterion [112] to the batch data. The second and third are SLANTS(1) with linear spline and SLANTS(2) with quadratic splines. SLANTS(1) have 1 spline per dimension, which is exactly LASSO with auto-tuned penalty parameter in SLANTS. SLANTS(2) have 8 splines per dimension. We allow SLANTS to select from a maximum lag of 55, which is roughly the size of annual cycle of 52 weeks.

Fig. 4.7 shows the cumulative average one-step ahead prediction error at each time step by the above three approaches. Here we plot the fits to the last 800 data points due to the unstable estimates of AR and SLANTS at the beginning. The results show that SLANTS

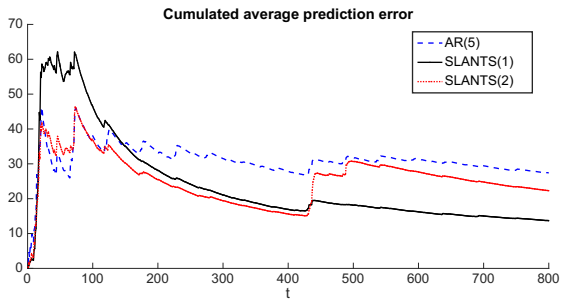


Figure 4.7: A graph showing the cumulated average one-step ahead prediction error at each time step produced by three approaches: linear autoregressive model, SLANTS with linear splines, and SLANTS with quadratic splines.

is more flexible and reliable than linear autoregressive model in practical applications. Both SLANTS(1) and SLANTS(2) selected lag 1,2,52,54 as significant predictors. It is interesting to observe that SLANTS(2) is preferred to SLANTS(1) before time step 436 (around the time when the 2008 financial crisis happened) while the simpler model SLANTS(1) is preferred after that time step. The fitted quadratic splines from SLANTS(2) are almost linear, which means the data has little nonlinearity. So SLANTS(1) performs best overall.

4.5 Concluding remarks

To address several challenges in time series prediction that arises from environmental science, economics, and finance, we proposed a new method to model nonlinear and high dimensional time series data in a sequential and adaptive manner. The performance of our method was demonstrated by both synthetic and real data experiments. We also provided rigorous theoretical analysis of the rate of convergence, estimation error, and consistency in variable selection of our method.

Future work may include modeling and joint prediction of $\boldsymbol{X}_{1,T}, \ldots, \boldsymbol{X}_{D,T}$. Currently, the prediction is separated into D individual problems. The performance may be further enhanced by considering potential correlations of innovations in each series. Adap-

tive placement of knots is another direction for future work. The knot sequence should adequately cover the range of data. In this chapter, we assumed that the range of data is known. In some practical applications, however, the range may vary over time. In such case, it would be helpful to add a rejuvenation step that routinely updates the empirical domain of the data (and thus the knot placement).

A

Supplementary material for Chapter 2

A.1 Proof of Theorem 3

We first provide a Lemma which plays an essential rule in the proofs of both the Glivenko-Cantelli Theorem and our Theorem 3.

Lemma 4 *Suppose F_n and F are (nonrandom) distribution functions on $\mathbb{R}$ such that*

$$F_n(x) \to F(x) \text{ for all } x \in \mathbb{R}.$$

If F is continuous, then we have

$$F_n(x-) \to F(x) \text{ for all } x \in \mathbb{R}, \tag{A.1}$$

and, moreover,

$$\sup_{x\in\mathbb{R}} |F_n(x) - F(x)| \to 0. \tag{A.2}$$

Proof 1 *We first show equation (A.1). Suppose x is a continuity point of F. Since* $F_n(x-) \leq F_n(x)$*,*

$$\limsup_{n\to\infty} F_n(x-) \leq \limsup_{n\to\infty} F_n(x) = F(x).$$

For any $y < x$*, we have*

$$F_n(y) \leq F_n(x-),$$

which implies that

$$F(y) = \liminf_{n\to\infty} F_n(y) \leq \liminf_{n\to\infty} F_n(x-).$$

Since

$$\lim_{y\to x-} F(y) = F(x),$$

the desired result follows.

We now show equation (A.2). Let $\varepsilon > 0$ *be given and consider a partition of the real line into finitely many pieces of the form* $-\infty = t_0 < t_1 < \cdots < t_k = \infty$ *such that, for* $0 \leq j \leq k-1$*,*

$$F\left(t_{j+1}\right) - F\left(t_j\right) \leq \frac{\varepsilon}{2}.$$

For any $x \in \mathbb{R}$*, there exists j such that* $t_j \leq x < t_{j+1}$*. For such j,*

$$\begin{aligned} F_n\left(t_j\right) &\leq F_n(x) \leq F_n\left(t_{j+1}-\right), \\ F\left(t_j\right) &\leq F(x) \leq F\left(t_{j+1}\right), \end{aligned}$$

which implies that

$$F_n\left(t_j\right) - F\left(t_{j+1}\right) \leq F_n\left(x\right) - F(x) \leq F_n\left(t_{j+1}-\right) - F\left(t_j\right).$$

Furthermore,

$$\begin{aligned} & F_n\left(t_j\right) - F\left(t_j\right) + F\left(t_j\right) - F\left(t_{j+1}\right) \\ \leq \;& F_n\left(x\right) - F(x), \\ & F_n\left(t_{j+1}-\right) - F\left(t_{j+1}\right) + F\left(t_{j+1}\right) - F\left(t_j\right) \\ \geq \;& F_n\left(x\right) - F(x). \end{aligned}$$

By the construction of the partition, we have that

$$\begin{aligned} F_n\left(t_j\right) - F\left(t_j\right) - \frac{\varepsilon}{2} &\leq F_n\left(x\right) - F(x), \\ F_n\left(t_{j+1}-\right) - F\left(t_{j+1}\right) + \frac{\varepsilon}{2} &\geq F_n\left(x\right) - F(x). \end{aligned}$$

For each j, let $N_j = N_j(\varepsilon)$ be such that, for $n > N_j$,

$$F_n\left(t_j\right) - F\left(t_j\right) > -\frac{\varepsilon}{2}.$$

Also, by equation (A.1), let $M_j = M_j(\varepsilon)$ be such that, for $n > M_j$,

$$F_n\left(t_j-\right) - F\left(t_j\right) < \frac{\varepsilon}{2}.$$

Let $N = \max_{1 \leq j \leq k} N_j \vee M_j$. For $n > N$ and any $x \in \mathbb{R}$, we then have that

$$\left|F_n\left(x\right) - F(x)\right| < \varepsilon.$$

The desired result follows.

Proof 2 *[of Theorem 3] Let W be the canonical graphon of a degree-identifiable ExGM. The marginal integral $g(u) \triangleq \int_0^1 W(u,v)\,dv$ must be strictly increasing, of which the range is not necessarily the whole $[0,1]$ interval. However, we will still use the notation g^{-1} to denote the corresponding CDF of the degree proportion random variable $g(U)$. As mentioned in Remark 2, g^{-1} will be a continuous function on $[0,1]$.*

First we show that $\left|\hat{U}_i - U_i\right| \to 0$ in probability. Recall that

$$\hat{U}_i = \hat{F}(D_i) \text{ and } U_i = g^{-1}(g(U_i)),$$

so

$$\left|\hat{U}_i - U_i\right| \leq \left|\hat{F}(D_i) - g^{-1}(D_i)\right| + \left|g^{-1}(D_i) - g^{-1}(g(U_i))\right|.$$

In the proof of Theorem 5 in [21], they prove that actually

$$\mathbb{E}\left(D_i - g(U_i)\right)^2 \to 0,$$

so, in particular, $D_i \to g(U_i)$ in probability and hence Continuous Mapping Theorem suggests

$$g^{-1}(D_i) \to g^{-1}(g(U_i)) \text{ in probability.}$$

Furthermore, Theorem 5 in [21] also suggests that $M_2\left(\hat{F}, g^{-1}\right) \to 0$ in probability. Here M_2 means the Mallows 2-distance between two distributions F_1 and F_2, which is defined by

$$M_2(F_1,F_2) \triangleq \min_F \left\{ \sqrt{\mathbb{E}(X-Y)^2} \;\middle|\; \begin{array}{c} (X,Y) \backsim F, \\ X \backsim F_1, Y \backsim F_2 \end{array} \right\}.$$

Especially, $M_2(F_n, F) \to 0$ if and only only if $F_n \Rightarrow F$ in distribution, i.e.,

$$F_n(x) \to F(x) \text{ for all } x \text{ being the continuous point of } F,$$

and

$$\int x^2 dF_n(x) \to \int x^2 dF(x).$$

Hence, for every subsequence $N_1, \ldots, N_m$, there exists a further subsequence $N_{m_1}, \ldots, N_{m_k}$ such that $M_2\left(\hat{F}_{N_{m_k}}, g^{-1}\right) \to 0$ a.s. In particular, this means, for a.s. ω,

$$\hat{F}_{N_{m_k}}(x) \to g^{-1}(x) \quad \forall x \in [0,1].$$

Because g^{-1} is continuous, Lemma 4 extends this result to

$$\sup_{x \in [0,1]} \left| \hat{F}_{N_{m_k}}(x) - g^{-1}(x) \right| \to 0 \text{ for a.s. } \omega.$$

Thus we have

$$\sup_{x \in [0,1]} \left| \hat{F}(x) - g^{-1}(x) \right| \to 0 \text{ in probability,} \tag{A.3}$$

which implies that

$$\left| \hat{F}(D_i) - g^{-1}(D_i) \right| \to 0 \text{ in probability.}$$

The desired result follows.

Next we show that $\left|\hat{U}_i - \tilde{U}_i\right| \to 0$ *in probability. We observe that*

$$\begin{aligned}
& \left|\tilde{U}_i - \hat{U}_i\right| \\
\leq\ & \left|\hat{F}(D_i) - \hat{F}(D_i-)\right| \\
\leq\ & \left|\hat{F}(D_i) - g^{-1}(D_i)\right| + \left|g^{-1}(D_i) - \hat{F}(D_i-)\right| \\
\leq\ & \sup_{x\in[0,1]}\left|\hat{F}(x) - g^{-1}(x)\right| + \lim_{x\to D_i-}\left|g^{-1}(x) - \hat{F}(x)\right| \\
\leq\ & 2\sup_{x\in[0,1]}\left|\hat{F}(x) - g^{-1}(x)\right|,
\end{aligned}$$

so equation (A.3) guarantees that $\left|\hat{U}_i - \tilde{U}_i\right| \to 0$ *in probability. This then finishes the proof of Theorem 3.*

A.2 Proof of Theorem 1

Proof 3 *The USVT-A estimate for the canonical graphon is explicitly written as*

$$\hat{W}(u,v) \triangleq \sum_{i,j=1}^{N} \hat{P}_{ij}\mathbb{1}_{S_{ij}}(u,v),$$

where $S_{ij} \triangleq \left(\tilde{U}_i - 1/N, \tilde{U}_i\right] \times \left(\tilde{U}_j - 1/N, \tilde{U}_j\right]$ *is the* (i,j)*-th square with spacing* $1/N$ *and* $\hat{P}_{ij}$ *is the USVT probability matrix estimation defined in Chapter 2 section 2.2.*

To begin with, we define

$$\left\|\hat{W} - W\right\|^2 \triangleq \mathbb{E}\left(\int_0^1\int_0^1 \left(\hat{W}(u,v) - W(u,v)\right)^2 dudv\right),$$

which can be decomposed into four pieces by inserting the following three objects

$$\begin{aligned} W_1(u,v) &\triangleq \sum_{i,j=1}^{N} W\left(\tilde{U}_i, \tilde{U}_j\right) 1_{S_{ij}}(u,v), \\ W_2(u,v) &\triangleq \sum_{i,j=1}^{N} W\left(U_i, U_j\right) 1_{S_{ij}}(u,v), \\ W_3(u,v) &\triangleq \sum_{i,j=1}^{N} P_{ij} 1_{S_{ij}}(u,v) \end{aligned}$$

then

$$\begin{aligned} \left\|\hat{W} - W\right\| \leq & \|W - W_1\| + \|W_1 - W_2\| \\ & + \|W_2 - W_3\| + \left\|W_3 - \hat{W}\right\|. \end{aligned}$$

Now we will look at these terms individually.

For the first term, since W_1 is just a stepwise W evaluated at the upright corner for each small square S_{ij}, $\|W - W_1\| \to 0$ as W is continuous on $[0,1]^2$ and hence uniformly continuous.[*]

For the third term, it's easy to see that, by the definition of P_{ij},

$$\begin{aligned} \|W_2 - W_3\|^2 &= \mathbb{E}\left(\frac{1}{N^2} \sum_{1 \leq i,j \leq N} \left(W\left(U_i, U_j\right) - P_{ij}\right)^2\right) \\ &= \mathbb{E}\left(\frac{1}{N^2} \sum_{i=1}^{N} W(U_i, U_i)^2\right) \\ &= \frac{1}{N}\mathbb{E}\left(W(U_i, U_i)^2\right) \to 0. \end{aligned}$$

[*]Actually, what we need here is just a uniform continuity modulus δ such that $|W(u,v) - W(u',v')| < \varepsilon$ whenever $|u - u'| < \delta$ and $|v - v'| < \delta$ in $[0,1]^2$. Thus, the continuity condition of W can be actually weaken by piecewise continuity on $[0,1]^2$ with only finitely many number of pieces.

For the final term, we refer to Theorem 2, where by definition the left hand side of equation (2.3) is the same as $\left\|W_3 - \hat{W}\right\|^2$.

Now we discuss $\|W_1 - W_2\|$, *which depends only on the empirical degree sorting. Denote the event* $\left|U_i - \tilde{U}_i\right| > \delta$ *by* E_i, *where* δ *is chosen by the uniform continuity of* W *such that*

$$\begin{aligned} |u_1 - u_2| &\leq \delta \quad \text{and } |v_1 - v_2| \leq \delta \\ &\Rightarrow \quad |W(u_1, v_1) - W(u_2, v_2)| < \varepsilon. \end{aligned}$$

Then we have

$$\begin{aligned} & \mathbb{E}\left(\left|W\left(\tilde{U}_i, \tilde{U}_j\right) - W\left(U_i, U_j\right)\right|^2\right) \\ = \; & \mathbb{E}\left(\left|W\left(\tilde{U}_i, \tilde{U}_j\right) - W\left(U_i, U_j\right)\right|^2; E_i \cup E_j\right) \\ & + \mathbb{E}\left(\left|W\left(\tilde{U}_i, \tilde{U}_j\right) - W\left(U_i, U_j\right)\right|^2; E_i^c \cap E_j^c\right) \\ \leq \; & 4\mathbb{P}\left(E_i \cup E_j\right) + \varepsilon^2 \\ \leq \; & 4\left(\mathbb{P}\left(E_i\right) + \mathbb{P}\left(E_j\right)\right) + \varepsilon^2, \end{aligned}$$

so

$$\begin{aligned}
& \|W_1 - W_2\|^2 \\
= & \sum_{i,j=1}^{N} \frac{1}{N^2} \mathbb{E}\left(\left|W\left(\tilde{U}_i, \tilde{U}_j\right) - W\left(U_i, U_j\right)\right|^2\right) \\
\leq & \sum_{i,j=1}^{N} \frac{1}{N^2}\left(4\left(\mathbb{P}\left(E_i\right) + \mathbb{P}\left(E_j\right)\right) + \varepsilon^2\right) \\
= & \frac{8}{N}\sum_{i=1}^{N} \mathbb{P}\left(\left|U_i - \tilde{U}_i\right| > \delta\right) + \varepsilon^2 \\
= & 8\mathbb{P}\left(\left|U_i - \tilde{U}_i\right| > \delta\right) + \varepsilon^2,
\end{aligned}$$

where the last equality clearly follows from the exchangeability and the definition of $\tilde{U}_i$*. By Theorem 3, we have* $\mathbb{P}\left(\left|U_i - \tilde{U}_i\right| > \delta\right) \to 0$*, so we have* $\limsup_{N\to\infty} \|W_1 - W_2\| \leq \varepsilon^2$*. Letting* $\varepsilon \to 0$ *implies* $\limsup_{N\to\infty} \|W_1 - W_2\| = 0$*, which finishes the proof.*

B

Supplementary material for Chapter 3

B.1 Proof of Theorem 4

Proof 4 *Denote the latent class label for each node as a vector* $\vec{C}_i = (C_{i1}, \ldots, C_{iK})$ *where*

$$C_{ij} = \begin{cases} 0, & c_i \neq j \\ 1, & c_i = j \end{cases}.$$

Define the $N \times K$ *matrix*

$$C = \begin{pmatrix} \vec{C}_1 \\ \vdots \\ \vec{C}_N \end{pmatrix}.$$

Notice that

$$E(\bar{G}) = E(\bar{\Phi}) = CE(\bar{P})C' = CMC',$$

where the last equality follows from ergodicity of the process $\{P^t\}$*. Intuitively* $\bar{G}$ *would converge to* CMC'*. Since the matrix of eigenvectors of* CMC' *only has K distinct rows, the eigenvectors of* $\bar{G}$ *would converge to those of* CMC'*, and eventually the rows of the eigenvector matrix would be well-separated for nodes in different classes.*

More formally, we first bound the difference of $\bar{G}$ *and* CMC'*. Let* $\|\mathcal{A}\|_F = (\sum_{i,j} a_{ij}^2)^{1/2}$ *denote the Frobenius norm of a matrix* $\mathcal{A}$*. We have*

$$E(\|\bar{G} - CMC'\|_F^2) = \sum_{i,j} \mathrm{Var}(\bar{G}_{ij}) = \sum_{i,j} \left[E(\mathrm{Var}(\bar{G}_{ij}|\bar{\Phi}_{ij})) + \mathrm{Var}(E(\bar{G}_{ij}|\bar{\Phi}_{ij}))\right].$$

The first term can be bounded by

$$\mathrm{Var}(\bar{G}_{ij}|\bar{\Phi}_{ij}) = \frac{\bar{\Phi}_{ij}(1 - \bar{\Phi}_{ij})}{T} \leq \frac{1}{4T}$$

because $0 \leq \bar{\Phi}_{ij} \leq 1$*. For the second term,*

$$\mathrm{Var}(E(\bar{G}_{ij}|\bar{\Phi}_{ij})) = \mathrm{Var}(\bar{\Phi}_{ij}) = \mathrm{Var}(\bar{P}_{c_i c_j}) = \frac{\varepsilon_{c_i c_j}^2}{T}.$$

Therefore,

$$E(\|\bar{G} - CMC'\|_F^2) \leq \frac{N^2(1 + 4\varepsilon^2)}{4T}$$

where $\varepsilon = \max_{c_i, c_j} \varepsilon_{c_i c_j}$*. By the Markov inequality, for any* $\delta > 0$*,*

$$P(\|\bar{G} - CMC'\|_F^2 > \delta) \leq \frac{N^2(1 + 4\varepsilon^2)}{4T\delta} \to 0 \text{ as } T \to \infty$$

As a result, the spectral norm $\|\bar{G} - CMC'\| \leq \|\bar{G} - CMC'\|_F$ *goes to 0 too. Based on lemma A.2 by [65], if M has K distinct eigenvalues, then the eigenvectors of* $\bar{G}$ *are close to the corresponding eigenvectors of* CMC'*. That is, let* u_i *be the eigenvector corresponding to the ith largest eigenvalues of* $\bar{G}$*. Let* ϑ_i *be the counterpart for* CMC'*. If* $\|\bar{G} - CMC'\| < \varepsilon$*, then* $\|u_i u_i^T - \vartheta_i \vartheta_i^T\| < \delta\varepsilon$*. This implies that* $1 - (u_i^T \vartheta_i)^2 < \delta\vartheta$*. That is,* u_i *is close to* ϑ_i *or* $-\vartheta_i$*. But* CMC' *has only K distinct rows. So the results show a spectral clustering on* $\bar{G}$ *will eventually lead to perfect recovery of the class labels.*

B.2 Proof of Theorem 5

We begin with the proofs of Lemmas 1–3.

Proof 5 (Proof of Lemma 1) *This is from Lemmas A1 and A2 by [16]. The main arguments are as follows. h, the expectation of the log-likelihood, is always maximized at the true parameters. For any partition of P, any refinement of the partition increases h. For any label assignment z, we can find a refinement that has at least* $r(z)/2$ *pairs of nodes that connect to at least* $\min_k n_k(c)$ *of nodes that differ at least* δ *from the truth.*

Proof 6 (Proof of Lemma 2) *Because of symmetry, we only consider* $p \in (0, \frac{1}{2}]$*. Let* $C_0 = p/2$*. Then* $C_0 < p < 1 - C_0$*. Let region* $C = [C_0, 1 - C_0]$*. By the Chernoff bound,* $P(|x - p| > \varepsilon) \leq 2\exp(-2N\varepsilon^2)$*. Therefore,* $P(x \notin C) \leq 2\exp(-Np^2/2)$*. Let* $E_C(x) = \sum_{x \in C} xp(x)$*. The subscript C denotes any operation restricted on region C. De-*

fine the following functions and constants:

$$\sigma(p) = p\log(p) + (1-p)\log(1-p); \quad \mathcal{M}_0 = \max_{p\in C} |\sigma(p)| = -\sigma(0.5) \le 0.7$$

$$\sigma'(p) = \log(p) - \log(1-p); \quad \mathcal{M}_1 = \max_{p\in C} |\sigma'(p)| = \log(1-C_0) - \log(C_0)$$

$$\sigma''(p) = \frac{1}{p} + \frac{1}{1-p}; \quad \mathcal{M}_2 = \max_{p\in C} |\sigma''(p)| = \frac{1}{C_0} + \frac{1}{1-C_0}$$

$$\sigma'''(p) = -\frac{1}{p^2} + \frac{1}{(1-p)^2}; \quad \mathcal{M}_3 = \max_{p\in C} |\sigma'''(p)| = \frac{1}{C_0^2} - \frac{1}{(1-C_0)^2}$$

$$\sigma^{(4)}(p) = \frac{1}{2p^3} + \frac{1}{2(1-p)^3}; \quad \mathcal{M}_4 = \max_{p\in C} |\sigma^{(4)}(p)| = \frac{1}{2C_0^3} + \frac{1}{2(1-C_0)^3}$$

We can get the following bounds:

$$|E_{\bar{C}}\sigma(x)| \le \mathcal{M}_0 P(x \notin C) \le 2\exp\left(-\frac{Np^2}{2}\right)$$

$$E(x-p)^3 = \frac{p(1-p)(1-2p)}{N^2} \le \frac{1}{4N^2}$$

$$E(x-p)^4 = \frac{p(1-p)^3 + p^3(1-p)}{N^3} + \frac{3(N-1)p^2(1-p)^2}{N^3} \le \frac{1}{2N^3} + \frac{1}{4N^2}$$

By Taylor expansion on region C,

$$\sigma(x) = \sigma(p) + \sigma'(p)(x-p) + \frac{\sigma''(p)}{2}(x-p)^2 + \frac{\sigma'''(p)}{6}(x-p)^3 + R(x)$$

$$|R(x)| \le \max_{x\in C} |\sigma^{(4)}(x)(x-p)^4/24|.$$

Thus

$$
\begin{aligned}
&N[E(\sigma(x)) - \sigma(p)] - \frac{1}{2} \\
&= NE\left[\sigma(x) - \sigma(p) - \sigma'(p)(x-p) - \frac{\sigma''(p)}{2}(x-p)^2\right] \\
&\leq NE_C\left[\sigma(x) - \sigma(p) - \sigma'(p)(x-p) - \frac{\sigma''(p)}{2}(x-p)^2\right] + N(2M_0 + 2M_1)P(x \notin C) \\
&\leq NE_C\left[\frac{\sigma'''(p)}{6}(x-p)^3\right] + \max_{x\in C}|\sigma^{(4)}(x)(x-p)^4/24| + N(2M_0 + 2M_1)P(x \notin C) \\
&\leq \frac{M_3}{24N} + \frac{M_4}{24}\left(\frac{1}{2N^2} + \frac{1}{4N}\right) + 2N\left(1 + M_1 + \frac{M_3}{6}\right)\exp\left(-\frac{Np^2}{2}\right) \\
&\to 0 \text{ as } N \to \infty,
\end{aligned}
$$

which completes the proof.

Proof 7 (Proof of Lemma 3) *For any z, as $\bar{P}$ averages P, $C_0 \leq \bar{P}_{kl}(z) \leq 1 - C_0$. We have*

$$g(z) - h(z) = \sum_{k\leq l} n_{kl}(z)[E(x_{kl}(z)) - \sigma(\bar{P}_{kl})]$$

where

$$x_{kl}(z) = \frac{o_{kl}(z)}{n_{kl}(z)} \sim \frac{1}{n_{kl}} Bin(n_{kl}, \bar{P}_{kl}).$$

By Lemma 2,

$$n_{kl}(z)[E(x_{kl}(z)) - \sigma(\bar{P}_{kl})] \to \frac{1}{2} + O\left(\frac{1}{n_{kl}(z)}\right).$$

Therefore

$$g(z) - h(z) = \frac{K(K+1)}{4} + O\left(\frac{K^2}{m(z)}\right).$$

Proof 8 (Proof of Theorem 5) *We want to show that there exists δ_0 such that*

$$Ef(c) - Ef(z) \geq \delta_0 \tag{B.1}$$

for all $z \neq c$. Then by Bernstein's inequality, we have

$$\frac{1}{T}\left|\sum_t [f^t(z) - Ef^t(z)]\right| \to 0 \text{ as } T \to \infty.$$

Therefore

$$\frac{1}{T}\sum_t f^t(c) - \frac{1}{T}\sum_t f^t(z) \to \frac{1}{T}\sum_t (Ef(c) - Ef(z)) \geq \delta_0$$

for all $z \neq c$. Then we get the conclusion that c is the unique maximizer of $\sum_t f^t(z)$. To show (B.1)*, we know*

$$\begin{aligned} Ef(c) - Ef(z) &= g(c) - g(z) \\ &= (h(c) - h(z)) + (g(c) - h(c)) - (g(z) - h(z)) \\ &\geq \delta m(c) r(z) + (g(c) - h(c)) - (g(z) - h(z)). \end{aligned}$$

by Lemma 1.

Let n_0 denote the threshold that, for all $N \geq n_0$, $|g(z) - h(z) - K(K+1)/4| \leq \delta_1$ if $m(z) \geq n_0$. Then for $m(c) \geq n_0$,

$$g(c) - h(c) \geq \frac{K(K+1)}{4} - \delta_1.$$

For any z, the total number of nodes are $N \geq Km(c)$. The total number of nodes that do not satisfy $n_k(z) \geq n_0$ is at most $n_0(K-1)$. And for the rest of the nodes, we still have the bounded feature as in Lemma 2. Hence

$$g(z) - h(z) \leq \delta_2 n_0 (K-1) + \frac{K(K+1)}{4} + \delta_1.$$

Therefore

$$Ef(c) - Ef(z) \geq \delta m(c) r(z) - 2\delta_1 - \delta_2 n_0 (K-1).$$

This is an increasing function of $m(c)$. So we can find large enough $m(c)$ that $\delta m(c) r(z) - 2\delta_1 - \delta_2 n_0 (K-1) \geq \delta_0$ as required.

B.3 Minimum number of nodes for consistency with 2 classes

Theorem 5 guarantees consistency of the MLE provided the conditions on C_0 and δ are satisfied, and the minimum number of nodes in any class is large enough. For the special case of $K = 2$ classes, we can calculate the minimum number of nodes N required to guarantee consistency. We need N sufficiently large so that the expectation of the log-likelihood under the true class labels c at each layer is larger than the expectation of the log-likelihood under any other class assignment z. With 2 classes, for any given value of N, we can simply enumerate over the number of misclassified nodes to determine if this is indeed true. It suffices only to check two boundary cases for the other class assignment z:

- 2 nodes are in one class, and the remaining $N-2$ nodes are in the other class.
- Only a single node is misclassfied in z, i.e. $c_i = z_i$ for all except a single value of $i \in \{1, \ldots, N\}$.

If the expectation of the log-likelihood under c is indeed larger than the expectation of the log-likelihood under any other class assignment z for both of these cases, then consistency as $T \to \infty$ is guaranteed for this value of N. If it is not, then one can simply iterate over values of N until it is large enough to guarantee sufficiency.

B.4 Details of variational approximation

Let vector $\vec{z}_i = (z_{i1}, \ldots, z_{iK})$ denote the class assignment vector for each node i.

$$z_{ik} = \begin{cases} 1 & \text{if } i \text{ in class } k \\ 0 & \text{otherwise.} \end{cases}$$

So $\vec{z}_i$ has all zeros except a single one indicating its class. This notation is easier for writing down the likelihood. We denote the initial class assignment probability by $\vec{\pi} = (\pi_1, \ldots, \pi_K)$. This is the multinomial parameter of $\vec{z}_i$. The likelihood is

$$l = \prod_{i,k} \pi_k^{z_{ik}} \prod_{i<j,k\leq l,t} [(p_{kl}^t)^{g_{ij}^t}(1-p_{kl}^t)^{1-g_{ij}^t}]^{z_{ik}z_{jl}}.$$

It is difficult to maximize because $\vec{z}$ cannot be integrated out. Instead we use variational approximation to decompose the likelihood into independent marginal distributions and apply an expectation-maximization technique to search for the maximum [67]. Let $\vec{z}_i$ follow independent multinomial distributions, i.e.

$$\vec{z}_i \overset{\text{ind}}{\sim} \text{Multi}(b_{i1}, \ldots, b_{iK}), \quad E[z_{ik}] = b_{ik}.$$

In the variational E-step, the approximate marginal distribution $q(z_i)$ is

$$\ln q(z_i) = \sum_k z_{ik} \left(\ln \pi_k + \sum_{j\neq i,l,t} E[z_{jl}] \left(g_{ij}^t \ln P_{kl}^t + (1-g_{ij}^t) \ln(1-P_{kl}^t) \right) \right) + \text{Const.} \quad \text{(B.2)}$$

We update b_{ik} according to (B.2). That is,

$$b_{ik} \propto \pi_k \prod_{j \neq i} \prod_t \prod_l \left[(P^t_{kl})^{g^t_{ij}} (1 - P^t_{kl})^{1 - g^t_{ij}} \right]^{b_{jl}}$$

In the M-step, we maximize $\vec{\pi}$ and P^t by

$$\pi_k \propto \sum_i E[z_{ik}] = \sum_i b_{ik}$$
$$P^t_{kl} = \frac{\sum_{i \neq j} E[z_{ik}] E[z_{jl}] g^t_{ij}}{\sum_{i \neq j} E[z_{ik}] E[z_{jl}]} = \frac{\sum_{i \neq j} b_{ik} b_{jl} g^t_{ij}}{\sum_{i \neq j} b_{ik} b_{jl}}.$$

We iterate between the two steps until convergence.

C

Supplementary material for Chapter 4

We prove Theorems 1-3 in Chapter 4 in the appendix. For any real-valued column vector $x = [x_1, \ldots, x_m]$, we let $\|x\|_2 = (\sum_{i=1}^m x_i^2)^{1/2}$, $\|x\|_A = x^T A x$ denote respectively the ℓ_2 norm and matrix norm (with respect to A, a positive semidefinite matrix).

Proof of Theorem 6

At time T and iteration k, we define the functions $h(\cdot)$ and $g(\cdot)$ that respectively map $\hat{\beta}_T^{(k)}$ to $\boldsymbol{r}_T^{(k)}$ and from $\boldsymbol{r}_T^{(k)}$ to $\hat{\beta}_T^{(k+1)}$, namely $\hat{\beta}_T^{(k)} \overset{h}{\mapsto} \boldsymbol{r}_T^{(k)}$, $\boldsymbol{r}_T^{(k)} \overset{g}{\mapsto} \hat{\beta}_T^{(k+1)}$. Suppose that the largest eigenvalue of $I - \tau^2 A_{T+1}$ in absolute value is ξ ($\xi < 1$). We shall prove that

$$\|g(h(\chi_1)) - g(h(\chi_2))\|_2 \leq \xi \|\chi_1 - \chi_2\|_2. \tag{C.1}$$

It suffices to prove that $\|h(\alpha_1)-h(\alpha_2)\|_2 \leq \xi\|\alpha_1-\alpha_2\|_2$ and $\|g(\chi_1)-g(\chi_2)\|_2 \leq \|\chi_1-\chi_2\|_2$ for any vectors $\alpha_1, \alpha_2, \chi_1, \chi_2$. The first inequality follows directly from the definition of $\boldsymbol{r}^{(k)}$ in the E step, and $h(\alpha_1) - h(\alpha_2) = (I - \tau^2 A_T)(\alpha_1 - \alpha_2)$. To prove the second inequality, we prove

$$\|g(\chi_{1,i}) - g(\chi_{2,i})\|_2 \leq \|\chi_{1,i} - \chi_{2,i}\|_2, \tag{C.2}$$

where $\chi_{k,i}$ $(i = 1, \ldots, L)$ are subvectors (groups) of corresponding to $\hat{\beta}_{T,i}^{(k)}$ for either $k = 1$ or $k = 2$. For brevity we define $\tilde{\tau} = \lambda_T \tau_T^2$. We prove (C.2) by considering three possible cases: 1) $\|\chi_{1,i}\|_2, \|\chi_{2,i}\|_2 \geq \tilde{\tau}$; 2) one of $\|\chi_{1,i}\|_2$ and $\|\chi_{2,i}\|_2$ is less than $\tilde{\tau}$ while the other is no less than $\tilde{\tau}$; 3) $\|\chi_{1,i}\|_2, \|\chi_{2,i}\|_2 < \tilde{\tau}$. For case 1), $g(\chi_{1,i}) = g(\chi_{2,i}) = \mathbf{0}$ and (C.2) trivially holds. For case 2), assume without loss of generality that $\|\chi_{2,i}\|_2 < \tilde{\tau}$. Then

$$\begin{aligned}\|g(\chi_{1,i}) - g(\chi_{2,i})\|_2 = \|g(\chi_{1,i})\|_2 &= \|\chi_{1,i}\|_2 - \tilde{\tau} \\ &\leq \|\chi_{1,i}\|_2 - \|\chi_{2,i}\|_2 \leq \|\chi_{1,i} - \chi_{2,i}\|_2.\end{aligned}$$

For case 3), we note that $g(\chi_{k,i})$ is in the same direction of $\chi_{k,i}$ for $k = 1, 2$. We define the angle between $\chi_{1,i}$ and $\chi_{2,i}$ to be ϑ, and let $a = \|\chi_{1,i}\|$, $b = \|\chi_{2,i}\|$. By the Law of Cosines, to prove $\|g(\chi_1) - g(\chi_2)\|_2^2 \leq \|\chi_1 - \chi_2\|_2^2$ it suffices to prove that

$$\begin{aligned}&(a - \tilde{\tau})^2 + (b - \tilde{\tau})^2 - 2(a - \tilde{\tau})(b - \tilde{\tau})\cos(\vartheta) \\ &\leq a^2 + b^2 - 2ab\cos(\vartheta).\end{aligned} \tag{C.3}$$

By elementary calculations, Inequality (C.3) is equivalent to $2\{1 - \cos(\vartheta)\}\{(a + b)\tilde{\tau} - \tilde{\tau}^2)\} \geq 0$, which is straightforward.

Finally, Inequality (C.1) and Banach Fixed Point Theorem imply that there exists a *unique*

fixed point $\hat{\beta}_T$ and,

$$\|\hat{\beta}_T^{(k)} - \hat{\beta}_T\|_2 \leq \frac{\xi^k}{1-\xi}\|\hat{\beta}_T^{(1)} - \hat{\beta}_T^{(0)}\|_2$$

which decays exponentially in k for any given initial value $\hat{\beta}_T^{(0)}$.

Moreover, the fixed point $\hat{\beta}_T$ is MAP, because each EM iteration increases the value in (4.10) *implicitly* by increasing the value in $Q(\beta \mid \hat{\beta}_T^{(k)})$ (see the justification of EM algorithm [113, 114]).

Proof of Theorem 7

The proof follows standard techniques in high-dimensional regression settings [83, 108]. We only sketch the proof below. For brevity, $\hat{\beta}_T$ and $\hat{\beta}_{T,d}$ are denoted as $\hat{\beta}$ and $\hat{\beta}_d$, respectively.

Let $\tilde{S}_1 = S_0 \cup S_1$ be the set union of truly nonzero set of coefficients and the selected nonzero coefficients. By the definition of $\tilde{S}_1$, we have

$$\begin{aligned} &\|Y - Z_{\tilde{S}_1}\hat{\beta}_{\tilde{S}_1}\|_2^2 + \tilde{\lambda}\sum_{d\in\tilde{S}_1}\|\hat{\beta}_d\|_2 \\ &\leq \|Y - Z_{\tilde{S}_1}\beta_{\tilde{S}_1}\|_2^2 + \tilde{\lambda}\sum_{d\in\tilde{S}_1}\|\beta_d\|_2. \end{aligned} \tag{C.4}$$

Define $\varrho = Y - Z\beta$, and $\psi = Z_{\tilde{S}_1}(\hat{\beta}_{\tilde{S}_1} - \beta_{\tilde{S}_1})$. We obtain

$$\begin{aligned}
\|\psi\|_2^2 &\le 2\psi^T\varrho + \tilde{\lambda}\sum_{d\in\tilde{S}_1}(\|\beta_d\|_2 - \|\hat{\beta}_d\|_2) \\
&\le 2\psi^T\varrho + \tilde{\lambda}\sum_{d\in S_0}(\|\beta_d\|_2 - \|\hat{\beta}_d\|_2) \\
&\le 2\psi^T\varrho + \tilde{\lambda}\sqrt{|S_0|}\|\beta_{\tilde{S}_1} - \hat{\beta}_{\tilde{S}_1}\|_2 \\
&\le 2\psi^T\varrho + \tilde{\lambda}\sqrt{|S_1|}\|\beta_{\tilde{S}_1} - \hat{\beta}_{\tilde{S}_1}\|_2 \\
&\le 2\|\psi\|_2\|\varrho\|_2 + \tilde{\lambda}\sqrt{|S_1|}\|\beta_{\tilde{S}_1} - \hat{\beta}_{\tilde{S}_1}\|_2
\end{aligned}$$

where the first inequality is rewritten from (C.4), the second and fourth follow from $S_0 \subseteq \tilde{S}_1$, the third and fifth follow from Cauchy inequality. From the above equality and $2\|\psi\|_2\|\varrho\|_2 \le \|\psi\|_2^2/2 + 2\|\varrho\|_2^2$, we obtain

$$\|\psi\|_2^2 \le 4\|\varrho\|_2^2 + 2\tilde{\lambda}\sqrt{|S_1|}\|\beta_{\tilde{S}_1} - \hat{\beta}_{\tilde{S}_1}\|_2. \tag{C.5}$$

On the other hand, Assumption 4 gives $\|\psi\|_2^2 \ge \kappa T\|\beta_{\tilde{S}_1} - \hat{\beta}_{\tilde{S}_1}\|_2^2$. Therefore,

$$\begin{aligned}
\kappa T\|\beta_{\tilde{S}_1} - \hat{\beta}_{\tilde{S}_1}\|_2^2 &\le 4\|\varrho\|_2^2 + 2\tilde{\lambda}\sqrt{|S_1|}\|\beta_{\tilde{S}_1} - \hat{\beta}_{\tilde{S}_1}\|_2 \\
&\le 4\|\varrho\|_2^2 + \frac{2\tilde{\lambda}^2|S_1|}{\kappa T} + \frac{\kappa T}{2}\|\beta_{\tilde{S}_1} - \hat{\beta}_{\tilde{S}_1}\|_2^2
\end{aligned}$$

which implies that

$$\|\beta_{\tilde{S}_1} - \hat{\beta}_{\tilde{S}_1}\|_2^2 \le 8\|\varrho\|_2^2/(\kappa T) + 4\tilde{\lambda}^2|S_1|/(\kappa T)^2. \tag{C.6}$$

In order to bound $\|\beta_{\tilde{S}_1} - \hat{\beta}_{\tilde{S}_1}\|_2$, it remains to bound $\|\varrho\|_2$. Since ϱ_t can be written as

$$\varepsilon_t + \sum_{d \in \tilde{S}_1} \{f_d(X_{d,t}) - f_d^*(X_{d,t})\} + (\mu - \bar{Y}),$$

where $(\mu - \bar{Y}) = O_p(T^{-1})$ and $\|f_d - f_d^*\|_\infty = O(v^{-p} + v^{1/2}T^{-1/2})$ [83, Lemma 1], we obtain $\|\varrho\|_2^2 \leq 2\|\varepsilon\|_{P_X}^2 + c_2 T v^{-2p} + O_p(1)$ for sufficiently large T, where c_2 is a constant that does not depend on v, and P_X is the projection matrix of $Z_{\tilde{S}_1}$. On the other side,

$$\|\varepsilon\|_{P_X}^2 \leq \|Z_{\tilde{S}_1}^T \varepsilon\|_2^2/(\varkappa T).$$

Therefore,

$$\begin{aligned}\|\beta_{\tilde{S}_1} - \hat{\beta}_{\tilde{S}_1}\|_2^2 \leq 8c_2 v^{-2p}/\varkappa + O(T^{-2}\|Z_{\tilde{S}_1}^T \varepsilon\|_2^2) \\ + O_p(T^{-1}) + O(T^{-2}\tilde{\lambda}^2).\end{aligned}$$

To finish the proof of (4.22), it remains to prove that $\|Z_{\tilde{S}_1}^T \varepsilon\|_2^2 = O_p(T \log \tilde{D})$. Note that the elements of ε are not i.i.d. conditioning on $Z_{\tilde{S}_1}$ due to time series dependency, which is different from the usual regression setting. However, for any of the $|S_1|v$ column of $Z_{\tilde{S}_1}$, say $\boldsymbol{z}_{d,j}$, the inner product $\boldsymbol{z}_{d,j}^T \boldsymbol{\varepsilon} = \sum_{t=1}^T z_{d,j,t}\varepsilon_t$ is the sum of a martingale difference sequence (MDS) with sub-exponential condition. Applying the Bernstein-type bound for a MDS, we obtain for all $w > 0$ that

$$P\left(\left|\sum_{t=1}^T z_{d,j,t}\varepsilon_t\right| > w\right) \leq 2\exp\left\{-w^2/(2\sum_{t=1}^T \eta_t)\right\}, \text{ where}$$
$$\eta_t \stackrel{\Delta}{=} var z_{d,j,t}\varepsilon_t \leq z_{d,j,t}^2\sigma^2 \leq \sup_{x \in [a,b]} \{b_{d,j}(x)\}^2\sigma^2.$$

Thus, $\sum_{t=1}^T z_{d,j,t}\varepsilon_t$ is a sub-Gaussian random variable for each d, j. By applying similar tech-

niques used in the maximal inequality for Gaussian random variables [115],

$$\max_{d\in\tilde{S}_1,1\le j\le v} E(T^{-1/2}\boldsymbol{z}_{d,j}^{\mathrm{T}}\boldsymbol{\varepsilon}) \le O(T^{-1/2}(\log\tilde{D})^{1/2}).$$

Therefore,

$$\begin{aligned}\|Z_{\tilde{S}_1}^{\mathrm{T}}\boldsymbol{\varepsilon}\|_2^2 &\le |S_1|vT\max_{d\in\tilde{S}_1,1\le j\le v}\{E(T^{-1/2}\boldsymbol{z}_{d,j}^{\mathrm{T}}\boldsymbol{\varepsilon})\}^2\\ &\le O_p(T\log\tilde{D}).\end{aligned}$$

To prove $\lim_{T\to\infty} P(S_0 \subseteq S_1) = 1$, we define the event E_0 as "There exists $d \in S_0$ such that $\hat{\beta}_d = 0$ and $\beta_d \neq 0$". Under event E_0, let d satisfy the above requirement. Since $\|f_d - f_d^*\|_\infty = O(v^{-p} + v^{1/2}T^{-1/2})$, there exists a constant c_1' such that for all $v \ge c_1' c_0^{-1/p}$ and sufficiently large T, $\|f_d^*\|_2 \ge c_0/2$. By a result from [116], $\|\beta_d\|_2^2/v \ge c_2'\|f_d^*\|_2^2$ holds for some constant c_2'. Then, under E_0 it follows that $\|\beta - \hat{\beta}\|_2^2 \ge \|\beta_d\|_2^2 \ge c_2' v c_0^2/4 \ge 16c_2 v^{-2p}/\varkappa$ for all $v \ge c_1'' c_0^{-2/(2p+1)}$, where c_1'' is some positive constant. This contradicts the bound given in (4.22) for large T.

Proof of Theorem 8

Recall that the backward selection procedure produces a nested sequence of subsets $S_2 = \mathcal{S}^{(K)} \subseteq \cdots \subseteq \mathcal{S}^{(1)} \subseteq \mathcal{S}^{(0)} = S_1$ with corresponding MSE $\hat{e}^{(k)}$ ($k = 0, \ldots, K$), where $0 \le K \le |S_1| - |S_2|$. In addition, $\mathcal{S}^{(k)} = \mathcal{S}^{(k-1)} - \{\bar{d}_k^*\}$ for some $\bar{d}_k^* \in \mathcal{S}^{(k-1)}$. It suffices to prove that as T goes to infinity, with probability going to one i) $S_0 \subseteq \mathcal{S}^{(k)}$ for each $k = 0, \ldots, K$, and ii) $|S_2| = |S_0|$.

Following a similar proof by [105, Proof of Theorem 1], it can be proved that for any k, conditioned on $S_0 \subseteq \mathcal{S}^{(k-1)}$, we have $\hat{e}^{(k-1)} - \hat{e}^{(k)} = O_p(v_T/T)$ if $S_0 \subseteq \mathcal{S}^{(k-1)}$, and $\hat{e}^{(k-1)} - \hat{e}^{(k)} = c + o_p(1)$ for some constant $c > 0$ if $S_0 \not\subseteq \mathcal{S}^{(k-1)}$. Note that the penalty

increment $(v_T \log T)/T$ is larger than $O_p(v_T/T)$ and smaller than $c + o_p(1)$ for large T. By successive application of this fact finitely many times, we can prove that $S_0 \subseteq \mathcal{S}^{(k)}$ for each $k = 0, \ldots, K$, and that $|S_2| = |S_0|$ with probability close to one.

Derivation of Equation (4.12) in SLANTS

We need to compute

$$Q(\beta \mid \hat{\beta}_T^{(k)}) = E_{\vartheta_T | (\hat{\beta}_T^{(k)}, \boldsymbol{Y}_T)} \log p(\boldsymbol{Y}_T, \vartheta_T \mid \beta_T) - \lambda_T \sum_{i=1}^{\tilde{D}} \|\beta_i\|_2$$

up to a constant (which does not depend on β). The complete log-likelihood is

$$\begin{aligned} \log p(\boldsymbol{Y}_T, \vartheta_T \mid \beta) &= C_0 - \frac{\|\vartheta_T - \beta\|^2}{2\tau_T^2} \\ &= C_1 - \frac{\beta^T \beta - 2\beta^T \vartheta_T}{2\tau_T^2}, \end{aligned}$$

where C_1 and C_2 are constants that do not involve β. So it remains to calculate $E_{\vartheta_T | (\hat{\beta}_T^{(k)}, \boldsymbol{Y}_T)} \vartheta_T$. Note that $\boldsymbol{Y}_T | \vartheta_T \sim N(Z_T \vartheta_T, W_T^{-1} - \tau_T^2 Z_T Z_T^T)$, $\vartheta_T | \hat{\beta}_T^{(k)} \sim N(\hat{\beta}_T^{(k)}, \tau_T^2 I)$. Thus, $\vartheta_T \mid (\hat{\beta}_T^{(k)}, \boldsymbol{Y}_T)$ is Gaussian with mean

$$E_{\vartheta_T | (\hat{\beta}_T^{(k)}, \boldsymbol{Y}_T)} \vartheta_T = \boldsymbol{r}^{(k)}.$$

It follows that

$$Q(\beta \mid \hat{\beta}_T^{(k)}) = -\frac{1}{2\tau_T^2} \|\beta - \boldsymbol{r}^{(k)}\|_2^2 - \lambda_T \sum_{i=1}^{\tilde{D}} \|\beta_i\|_2.$$

References

[1] Panos Toulis and Edward Kao. Estimation of causal peer influence effects. In *International Conference on Machine Learning*, pages 1489–1497, 2013.

[2] Guillaume W Basse and Edoardo M Airoldi. Optimal design of experiments in the presence of network-correlated outcomes. *ArXiv e-prints*, 2015.

[3] Alexander D'Amour and Edoardo Airoldi. Misspecification, sparsity, and superpopulation inference for sparse social networks. *Ongoing work for publication and inclusion in dissertation*, 2016.

[4] Daniel L Sussman and Edoardo M Airoldi. Elements of estimation theory for causal effects in the presence of network interference. *arXiv preprint arXiv:1702.03578*, 2017.

[5] Panos Toulis, Alexander Volfovsky, and Edoardo M Airoldi. Propensity score methodology in the presence of network entanglement between treatments. *arXiv preprint arXiv:1801.07310*, 2018.

[6] Jean Pouget-Abadie, David C Parkes, Vahab Mirrokni, and Edoardo M Airoldi. Optimizing cluster-based randomized experiments under a monotonicity assumption. *arXiv preprint arXiv:1803.02876*, 2018.

[7] Niloy Biswas and Edoardo M Airoldi. Estimating peer-influence effects under homophily: Randomized treatments and insights. In *International Workshop on Complex Networks*, pages 323–347. Springer, 2018.

[8] Justin J. Yang, Qiuyi Han, and Edoardo M. Airoldi. Non-parametric estimation and testing of exchangeable graph models. In *Proceedings of the 17th International Conference on Artificial Intelligence and Statistics*, pages 1060–1067, 2014.

[9] Qiuyi Han, Kevin Xu, and Edoardo M. Airoldi. Consistent estimation of dynamic and multi-layer block models. In *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)*, pages 1511–1520, 2015.

[10] Qiuyi Han, Jie Ding, Edoardo M. Airoldi, and Vahid Tarokh. Slants: Sequential adaptive nonlinear modeling of time series. *IEEE Journal of Selected Topics in Signal Processing*, 65:4994–5005, 2017.

[11] D. N. Hoover. Relations on probability spaces and arrays of random variables. Preprint, *Institute for Advanced Study*, Princeton, NJ, 1979.

[12] D. J. Aldous. Representations for partially exchangeable arrays of random variables. *Journal of Multivariate Analysis*, 11:581–598, December 1981.

[13] O. Kallenberg. On the representation theorem for exchangeable arrays. *Journal of Multivariate Analysis*, 30(1):137–154, 1989.

[14] O. Kallenberg. *Probabilistic Symmetries and Invariance Principles*. Springer, 2005.

[15] K. T. Miller, T. S. Griffiths, and M. I. Jordan. Nonparametric latent feature models for link prediction. In *Advances in Neural Information Processing Systems (NIPS)*, 2009.

[16] David S. Choi, Patrick J. Wolfe, and Edoardo M. Airoldi. Stochastic blockmodels with a growing number of classes. *Biometrika*, 99:273–284, 2012.

[17] P. Diaconis and S. Janson. Graph limits and exchangeable random graphs. *Rendiconti di Matematica edelle sue Applicazioni, Series VII*, 28:33–61, 2008.

[18] C. Borgs, J. T. Chayes, L. Lovász, V. T. Sós, and K. Vesztergombi. Convergent sequences of dense graph I: Subgraph frequencies, metric properties and testing. *Advances in Mathematics*, 219:1801–1851, 2008.

[19] C. Borgs, J. T. Chayes, and L. Lovász. Moments of two-variable functions and the uniqueness of graph limits. *Geometric and Functional Analysis*, 19:1597–1619, 2010.

[20] Peter J. Bickel and Aiyou Chen. A nonparametric view of network models and Newman-Girvan and other modularities. *Proceedings of the National Academy of Sciences*, 106(50):21068–21073, 2009.

[21] P. J. Bickel, A. Chen, and E. Levina. The method of moments and degree distributions for network models. *Annals of Statistics*, 39(5):2280–2301, 2011.

[22] Sourav Chatterjee et al. Matrix estimation by universal singular value thresholding. *The Annals of Statistics*, 43(1):177–214, 2015.

[23] S. H. Chan and E. M. Airoldi. A consistent histogram estimator for exchangeable graph models. *Journal of Machine Learning Research, W&CP*, 32:208–216, 2014.

[24] B. P. Olding and P. J. Wolfe. Inference for graphs and networks: Extending classical tools to modern data. ArXiv:0906.4980, 2009. Unpulished manuscript.

[25] P. Erdös and A. Renyi. On random graphs. *Publicationes Mathematicae*, 6:290–297, 1959.

[26] E. N. Gilbert. Random graphs. *Annals of Mathematical Statistics*, 30(4):1141–1144, 1959.

[27] K. Nowicki and T. Snijders. Estimation and prediction of stochastic block structures. *Journal of American Statistical Association*, 96:1077–1087, 2001.

[28] Edoardo M. Airoldi, David M. Blei, Stephen E. Fienberg, and Eric P. Xing. Mixed membership stochastic blockmodels. *Journal of Machine Learning Research*, 9:1981–2014, 2008.

[29] J. K. Blitzstein and P. Diaconis. A sequential importance sampling algorithm for generating random graphs with prescribed degrees. Technical report, Stanford University, 2006.

[30] Karl Rohe, Sourav Chatterjee, and Bin Yu. Spectral clustering and the high-dimensional stochastic blockmodel. *The Annals of Statistics*, 39(4):1878–1915, 2011.

[31] Edoardo M. Airoldi, Thiago B. Costa, and Stanley H. Chan. Stochastic blockmodel approximation of a graphon: Theory and consistent estimation. In *Advances in Neural Information Processing Systems 26*, pages 692–700, 2013.

[32] S. H. Chan, T. B. Costa, and E. M. Airoldi. Estimation of exchangeable random graph models by stochastic blockmodel approximation. In *Proc. IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, pages 293–296, 2013.

[33] Patrick J. Wolfe and Sofia C. Olhede. Nonparametric graphon estimation. *arXiv preprint arXiv:1309.5936*, 2013.

[34] S. H. Chan, R. Khoshabeh, K. B. Gibson, P. E. Gill, and T. Q. Nguyen. An augmented lagrangian method for total variation video restoration. *IEEE Transactions on Image Processing*, 20(11):3097–3111, 2011.

[35] Anna Goldenberg, Alice X. Zheng, Stephen E. Fienberg, and Edoardo M. Airoldi. A survey of statistical network models. *Foundations and Trends® in Machine Learning*, 2(2):129–233, 2010.

[36] Paul W. Holland, Kathryn Blackmond Laskey, and Samuel Leinhardt. Stochastic blockmodels: First steps. *Social networks*, 5(2):109–137, 1983.

[37] Alain Celisse, Jean-Jacques Daudin, and Laurent Pierre. Consistency of maximum-likelihood and variational estimators in the stochastic block model. *Electronic Journal of Statistics*, 6:1847–1899, 2012.

[38] Jiashun Jin. Fast network community detection by SCORE. *arXiv preprint arXiv:1211.5803*, 2012.

[39] Peter Bickel, David Choi, Xiangyu Chang, and Hai Zhang. Asymptotic normality of maximum likelihood and its variational approximation for stochastic blockmodels. *The Annals of Statistics*, 41(4):1922–1943, 2013.

[40] Arash A. Amini, Aiyou Chen, Peter J. Bickel, and Elizaveta Levina. Pseudo-likelihood methods for community detection in large sparse networks. *The Annals of Statistics*, 41(4):2097–2122, 2013.

[41] Chao Gao, Yu Lu, and Harrison H. Zhou. Rate-optimal graphon estimation. *arXiv preprint arXiv:1410.5837*, 2014.

[42] Amr Ahmed and Eric P. Xing. Recovering time-varying networks of dependencies in social and biological studies. *Proceedings of the National Academy of Sciences*, 106(29):11878–11883, 2009.

[43] Steve Hanneke, Wenjie Fu, and Eric P. Xing. Discrete temporal models of social networks. *Electronic Journal of Statistics*, 4:585–605, 2010.

[44] Katsuhiko Ishiguro, Tomoharu Iwata, Naonori Ueda, and Joshua B. Tenenbaum. Dynamic infinite relational model for time-varying relational data analysis. In *Advances in Neural Information Processing Systems 23*, pages 919–927, 2010.

[45] Tianbao Yang, Yun Chi, Shenghuo Zhu, Yihong Gong, and Rong Jin. Detecting communities and their evolutions in dynamic social networks—a Bayesian approach. *Machine Learning*, 82(2):157–189, 2011.

[46] Qirong Ho, Le Song, and Eric P. Xing. Evolving cluster mixed-membership blockmodel for time-evolving networks. In *Proceedings of the 14th International Conference on Artificial Intelligence and Statistics*, pages 342–350, 2011.

[47] Kevin S. Xu and Alfred O. Hero. Dynamic stochastic blockmodels for time-evolving social networks. *IEEE Journal of Selected Topics in Signal Processing*, 8(4):552–562, 2014.

[48] Kevin S. Xu. Stochastic block transition models for dynamic networks. In *Proceedings of the 18th International Conference on Artificial Intelligence and Statistics*, pages 1079–1087, 2015.

[49] Purnamrita Sarkar and Andrew W. Moore. Dynamic social network analysis using latent space models. *ACM SIGKDD Explorations Newsletter*, 7(2):31–40, 2005.

[50] Purnamrita Sarkar, Sajid M. Siddiqi, and Geoffrey J. Gordon. A latent space approach to dynamic embedding of co-occurrence data. In *Proceedings of the 11th*

International Conference on Artificial Intelligence and Statistics, pages 420–427, 2007.

[51] Daniele Durante and David Dunson. Bayesian logistic gaussian process models for dynamic networks. In *Proceedings of the 17th International Conference on Artificial Intelligence and Statistics*, pages 194–201, 2014.

[52] James R. Foulds, Christopher DuBois, Arthur U. Asuncion, Carter T. Butts, and Padhraic Smyth. A dynamic relational infinite feature model for longitudinal social networks. In *Proceedings of the 14th International Conference on Artificial Intelligence and Statistics*, pages 287–295, 2011.

[53] Creighton Heaukulani and Zoubin Ghahramani. Dynamic probabilistic models for latent feature propagation in social networks. In *Proceedings of the 30th International Conference on Machine Learning*, pages 275–283, 2013.

[54] Myunghwan Kim and Jure Leskovec. Nonparametric multi-group membership model for dynamic networks. In *Advances in Neural Information Processing Systems 26*, pages 1385–1393, 2013.

[55] Stephen E. Fienberg, Michael M. Meyer, and Stanley S. Wasserman. Statistical analysis of multiple sociometric relations. *Journal of the American Statistical Association*, 80(389):51–67, 1985.

[56] Michael Szell, Renaud Lambiotte, and Stefan Thurner. Multirelational organization of large-scale social networks in an online world. *Proceedings of the National Academy of Sciences*, 107(31):13636–13641, 2010.

[57] Peter J. Mucha, Thomas Richardson, Kevin Macon, Mason A. Porter, and Jukka-Pekka Onnela. Community structure in time-dependent, multiscale, and multiplex networks. *Science*, 328(5980):876–878, 2010.

[58] Matteo Magnani and Luca Rossi. The ML-model for multi-layer social networks. In *Proceedings of the International Conference on Advances in Social Networks Analysis and Mining*, pages 5–12, 2011.

[59] Brandon Oselio, Alex Kulesza, and Alfred O. Hero. Multi-layer graph analytics for dynamic social networks. *IEEE Journal of Selected Topics in Signal Processing*, 8, 2014.

[60] Michael Salter-Townshend and Tyler H McCormick. Latent space models for multiview network data. Technical Report 622, University of Washington, 2013.

[61] Daniele Durante, David B. Dunson, and Joshua T. Vogelstein. Nonparametric bayes modeling of populations of networks. *arXiv preprint arXiv:1406.7851*, 2014.

[62] Yunpeng Zhao, Elizaveta Levina, and Ji Zhu. Consistency of community detection in networks under degree-corrected stochastic block models. *The Annals of Statistics*, 40(4):2266–2292, 2012.

[63] Jing Lei and Alessandro Rinaldo. Consistency of spectral clustering in stochastic block models. *The Annals of Statistics*, 43(1):215–237, 2014.

[64] Daniel L. Sussman, Minh Tang, Donniell E. Fishkind, and Carey E. Priebe. A consistent adjacency spectral embedding for stochastic blockmodel graphs. *Journal of the American Statistical Association*, 107(499):1119–1128, 2012.

[65] Roberto Imbuzeiro Oliveira. Concentration of the adjacency matrix and of the Laplacian in random graphs with independent edges. *arXiv preprint arXiv:0911.0600*, 2009.

[66] Edward R. Scheinerman and Kimberly Tucker. Modeling graphs using dot product representations. *Computational Statistics*, 25(1):1–16, 2010.

[67] Jean-Jacques Daudin, Franck Picard, and Stéphane Robin. A mixture model for random graphs. *Statistics and Computing*, 18(2):173–183, 2008.

[68] Harold W. Kuhn. The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2(1-2):83–97, 1955.

[69] Mark Newman and Michelle Girvan. Finding and evaluating community structure in networks. *Physical Review E*, 69(2):026113, 2004.

[70] Aurelien Decelle, Florent Krzakala, Cristopher Moore, and Lenka Zdeborová. Asymptotic analysis of the stochastic block model for modular networks and its algorithmic applications. *Physical Review E*, 84(6):066106, 2011.

[71] Nathan Eagle and Alex Pentland. Reality mining: sensing complex social systems. *Personal and Ubiquitous Computing*, 10(4):255–268, 2006.

[72] Ali Yener Mutlu and Selin Aviyente. Dynamic network summarization using convex optimization. In *Proceedings of the IEEE Workshop on Statistical Signal Processing*, pages 117–120, 2012.

[73] Kevin S. Xu, Mark Kliger, and Alfred O. Hero. Adaptive evolutionary clustering. *Data Mining and Knowledge Discovery*, 28(2):304–336, 2014.

[74] Brian Karrer and Mark Newman. Stochastic blockmodels and community structure in networks. *Physical Review E*, 83(1):016107, 2011.

[75] AU-CS MultiLayer network. http://sigsna.net/impact/datasets/, 2014.

[76] Theodore Wilbur Anderson. *The statistical analysis of time series*. John Wiley & Sons, 1971.

[77] George EP Box, Gwilym M Jenkins, and Gregory C Reinsel. *Time series analysis: forecasting and control*, volume 734,. John Wiley & Sons, 2011.

[78] Peter J Brockwell and Richard A Davis. *Time series: theory and methods*. Springer Science & Business Media, 2013.

[79] SHI Xingjian, Zhourong Chen, Hao Wang, Dit-Yan Yeung, Wai-kin Wong, and Wang-chun Woo. Convolutional lstm network: A machine learning approach for precipitation nowcasting. In *Adv. Neural. Inf. Process. Syst.*, pages 802–810, 2015.

[80] Shihao Yang, Mauricio Santillana, and SC Kou. Accurate estimation of influenza epidemics using google search data via argo. *Proc. Natl. Acad. Sci. U.S.A.*, 112(47):14473–14478, 2015.

[81] Sethu Vijayakumar, Aaron D'souza, and Stefan Schaal. Incremental online learning in high dimensions. *Neural computation*, 17(12):2602–2634, 2005.

[82] Jerome H. Friedman. Multivariate adaptive regression splines. *Ann. Statist.*, 19(1):1–67, 03 1991.

[83] Jian Huang, Joel L Horowitz, and Fengrong Wei. Variable selection in nonparametric additive models. *Ann. Stat.*, 38(4):2282, 2010.

[84] Howell Tong. *Threshold models in non-linear time series analysis*, volume 21. Springer Science & Business Media, 2012.

[85] Christian Gouriéroux. *ARCH models and financial applications*. Springer Science & Business Media, 2012.

[86] Trevor J Hastie and Robert J Tibshirani. *Generalized additive models*, volume 43. CRC Press, 1990.

[87] Zongwu Cai, Jianqing Fan, and Qiwei Yao. Functional-coefficient regression models for nonlinear time series. *J. Amer. Statist. Assoc.*, 95(451):941–956, 2000.

[88] Kun Zhang and Aapo Hyvärinen. On the identifiability of the post-nonlinear causal model. In *UAI*, pages 647–655. AUAI Press, 2009.

[89] Kun Zhang, Jonas Peters, Dominik Janzing, and Bernhard Schölkopf. Kernel-based conditional independence test and application in causal discovery. *arXiv preprint arXiv:1202.3775*, 2012.

[90] Jianqing Fan, Yang Feng, and Rui Song. Nonparametric independence screening in sparse ultra-high-dimensional additive models. *J. Amer. Statist. Assoc.*, 106(494):544–557, 2012.

[91] Robert Tibshirani. Regression shrinkage and selection via the LASSO. *J. R. Stat. Soc. Ser. B*, pages 267–288, 1996.

[92] Pradeep Ravikumar, John Lafferty, Han Liu, and Larry Wasserman. Sparse additive models. *J. Roy. Statist. Soc. Ser. B*, 71(5):1009–1030, 2009.

[93] Juan Andrés Bazerque, Gonzalo Mateos, and Georgios B Giannakis. Group-lasso on splines for spectrum cartography. *IEEE Trans. Signal Process.*, 59(10):4648–4663, 2011.

[94] Grace Wahba. *Spline models for observational data*, volume 59. Siam, 1990.

[95] Ming Yuan and Yi Lin. Model selection and estimation in regression with grouped variables. *J. Roy. Statist. Soc. Ser. B*, 68(1):49–67, 2006.

[96] Hui Zou. The adaptive lasso and its oracle properties. *J. Amer. Statist. Assoc.*, 101(476):1418–1429, 2006.

[97] Mário AT Figueiredo and Robert D Nowak. An em algorithm for wavelet-based image restoration. *IEEE Trans. Image Process.*, 12(8):906–916, 2003.

[98] Behtash Babadi, Nicholas Kalouptsidis, and Vahid Tarokh. Sparls: The sparse rls algorithm. *IEEE Trans. Signal Process.*, 58(8):4013–4025, 2010.

[99] Gerasimos Mileounis, Behtash Babadi, Nicholas Kalouptsidis, and Vahid Tarokh. An adaptive greedy algorithm with application to nonlinear communications. *IEEE Trans. Signal Process.*, 58(6):2998–3007, 2010.

[100] Hastie T. Friedman, J. and R. Tibshirani. Regularization paths for generalized linear models via coordinate descent. *J. Stat. Softw.*, 33(1):1–22, 2008.

[101] A Philip Dawid. Present position and potential developments: Some personal views: Statistical theory: The prequential approach. *J. Roy. Statist. Soc. Ser. A*, pages 278–292, 1984.

[102] Trevor Park and George Casella. The bayesian lasso. *J. Amer. Statist. Assoc.*, 103(482):681–686, 2008.

[103] Luke Bornn, Arnaud Doucet, and Raphael Gottardo. An efficient computational approach for prior sensitivity analysis and cross-validation. *Can J. Stat.*, 38(1):47–64, 2010.

[104] David LB Jupp. Approximation to data by splines with free knots. *SIAM. J. Numer. Anal.*, 15(2):328–343, 1978.

[105] Jianhua Z Huang and Lijian Yang. Identification of non-linear additive autoregressive models. *J. Roy. Statist. Soc. Ser. B*, 66(2):463–477, 2004.

[106] Charles J Stone. Additive regression and other nonparametric models. *Ann. Stat.*, pages 689–705, 1985.

[107] Gideon Schwarz. Estimating the dimension of a model. *Ann. Statist.*, 6(2):461–464, 1978.

[108] Trevor Hastie, Robert Tibshirani, and Martin Wainwright. *Statistical learning with sparsity: the LASSO and generalizations*. CRC Press, 2015.

[109] Lukas Meier, Sara Van De Geer, and Peter Bühlmann. The group lasso for logistic regression. *J. Roy. Statist. Soc. Ser. B*, 70(1):53–71, 2008.

[110] Yi Yang and Hui Zou. A fast unified algorithm for solving group-lasso penalize learning problems. *Statistics and Computing*, 25(6):1129–1141, 2015.

[111] Volker Roth and Bernd Fischer. The group-lasso for generalized linear models: uniqueness of solutions and efficient algorithms. In *ICML*, pages 848–855. ACM, 2008.

[112] Jie Ding, Vahid Tarokh, and Yuhong Yang. Bridging AIC and BIC: a new criterion for autoregression. *IEEE Trans. Inf. Theory*, 2017.

[113] Arthur P Dempster, Nan M Laird, and Donald B Rubin. Maximum likelihood from incomplete data via the EM algorithm. *J. Roy. Statist. Soc. Ser. B*, pages 1–38, 1977.

[114] CF Jeff Wu. On the convergence properties of the em algorithm. *Ann. Stat.*, pages 95–103, 1983.

[115] Aad W Van Der Vaart and Jon A Wellner. *Weak Convergence*. Springer, 1996.

[116] Carl De Boor. *A practical guide to splines*, volume 27. Springer-Verlag New York, 1978.