Statistica Sinica **30** (2020), 1437-1462
doi:https://doi.org/10.5705/ss.202017.0533

GROUPED NETWORK VECTOR AUTOREGRESSION

Xuening Zhu and Rui Pan

Fudan University and Central University of Finance and Economics

Abstract: Time series analyses are often used to model a continuous response for all individuals at equally spaced time points. With the rapid advance of social network sites, network data are becoming increasingly available. The network vector autoregression (NAR) model incorporates the network information among individuals. The response of each individual can be explained by its lagged value, the average of its neighbors, and a set of node-specific covariates. However, all individuals are assumed to be homogeneous because they share the same autoregression coefficients. To express individual heterogeneity, we develop a grouped NAR (GNAR) model. Individuals in a network can be classified into different groups characterized by sets of parameters. The strict stationarity of the GNAR model is established. Two estimation procedures are developed, as well as the asymptotic properties of the proposed model. Numerical studies are conducted to evaluate the finite-sample performance of our proposed methodology. Lastly, two real-data examples are presented, based on studies on user posting behavior on the Sina Weibo platform and on air pollution patterns (especially $PM_{2.5}$) in mainland China, respectively.

Key words and phrases: EM algorithm, network data, ordinary least square estimator, vector autoregression.

1. Introduction

An important result of the rapid development of the Internet has been the rise of social networks, such as Facebook, Twitter, Sina Weibo, and many others. Accordingly, network data are becoming increasingly available. On the one hand, users (i.e., nodes) in a social network are related (e.g., friendship) rather than being independent of each other. On the other hand, many covariates can be collected for each user, including personal information, consumption behavior, and textual records. As a result, network data play an important role in various disciplines. They can be used to provide site user portraits (Lewis et al. (2008)), characterize social capital flow patterns (Bohn et al. (2014)), and analyze consumer behavior (Hofstra, Corten and Buskens (2015)).

Mathematically, we use an adjacency matrix $A = (a_{ij}) \in \mathbb{R}^{N\times N}$ to represent the network structure, where N is the total number of nodes. If the ith node

follows the jth node, we set $a_{ij} = 1$; otherwise, $a_{ij} = 0$. For convenience, we always let $a_{ii} = 0$. In addition, we assume that a continuous response $Y_{it} \in \mathbb{R}^1$ can be observed for each node over time t. On a social network platform, Y_{it} could be the number of characters posted by node i at time t, reflecting nodal activeness. Furthermore, we study the dynamic pattern of $\mathbb{Y}_t = (Y_{1t}, \ldots, Y_{Nt})^\top \in \mathbb{R}^N$. To this end, prior studies often use vector autoregression (VAR) models and their corresponding dimension-reduction methods, especially the factor models (Pan and Yao (2008); Lam and Yao (2012)). Recently, Zhu et al. (2017) proposed a network vector autoregression (NAR) model, which takes the network structure into account when modeling the dynamics of $\mathbb{Y}_t$.

The NAR model assumes that the response Y_{it} is influenced by four factors: (a) its lagged value $Y_{i(t-1)}$; (b) its socially connected neighbors $n_i^{-1} \sum_j a_{ij} Y_{j(t-1)}$ with $n_i = \sum_j a_{ij}$; (c) a set of node-specific covariates $V_i \in \mathbb{R}^p$; and (d) an independent noise ε_{it}. Thus, the model is specified as follows:

$$Y_{it} = \beta_0 + \beta_1 n_i^{-1} \sum_j a_{ij} Y_{j(t-1)} + \beta_2 Y_{i(t-1)} + V_i^\top \gamma + \varepsilon_{it}, \tag{1.1}$$

where β_0, β_1, β_2, and γ are referred to as the baseline effect, network effect, momentum effect, and nodal effect, respectively.

Although model (1.1) can be used to study the dynamic pattern of $\mathbb{Y}_t$ when network information is available, it treats all nodes as being homogenous. For instance, according to the NAR model, the node-irrelevant network effect β_1 implies that all nodes are influenced by their neighbors to the same extent. This is obviously unrealistic in practice. For example, consider Sina Weibo, one of the most popular social network platforms in China. Some nodes on the platform are super stars or political leaders, and have millions of fans. These nodes are referred to as opinion leaders, and are less influenced by others (Wasserman and Faust (1994)). As a result, the network effect (i.e., β_1) for opinion leaders should be small. In contrast, their followers are more likely to be affected, leading to a relatively large network effect for these ordinary nodes.

From the above discussion, we conclude that the baseline effect, network effect, momentum effect, and nodal effect might vary among groups of nodes. As discussed later, our real-data shows that nodes in a network can be classified into K groups characterized by different sets of parameters (e.g., β_{1k}, for $k = 1, \ldots, K$). Figure 1 shows that for the Sina Weibo data set, nodes are classified into three groups, each with different coefficient estimates. Specifically, the estimated network effect is much smaller for group 3 than it is for group 2 (i.e.,

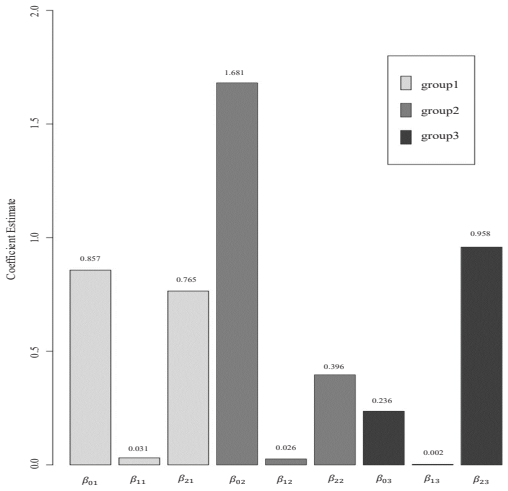

Figure 1. Coefficient estimates for three groups. Distinct characteristics are evident for different groups of nodes.

$\widehat{\beta}_{12} = 0.026$ vs. $\widehat{\beta}_{13} = 0.002$). On the other hand, group 3 has a larger estimated momentum effect than that of group 2 (i.e., $\widehat{\beta}_{22} = 0.396$ vs. $\widehat{\beta}_{23} = 0.958$). This indicates that nodes in group 2 tend to be affected by their connected neighbors, whereas those in group 3 are more likely to be self-influenced.

In order to capture this interesting phenomenon, we propose a grouped network vector autoregression (GNAR) model. The GNAR model basically assumes that nodes in a network can be classified into groups characterized by different sets of parameters. The proposed model is related to the literature on the clustering of time series data, where the most popular technique is model-based clustering, established using finite-mixture models (Fröhwirth-Schnatter and Kaufmann (2008); Juárez and Steel (2010); Wang et al. (2013)). According to this approach, each time series is assumed to belong to one group, and each group is characterized by a different data-generating mechanism. The method is widely applied to gene expression classification (Luan and Li (2003); Heard, Holmes and Stephens (2006)), financial data modeling (Frühwirth-Schnatter and Kaufmann (2006); Bauwens and Rombouts (2007)), and economic growth analyses (Fröhwirth-Schnatter and Kaufmann (2008); Juárez and Steel (2010); Wang et al. (2013)). To the best of our knowledge, most of the above methods deal with independent univariate time series, and can be difficult to apply directly to network data.

In this study, we group users according to their dynamic network behaviors. The network information is embedded in the model. Section 2 introduces the GNAR model, including establishing the strict stationarity of $\mathbb{Y}_t$. In section 3, two estimation methods are developed: an EM algorithm, and a two-step (TS) estimation procedure. This section also presents the corresponding asymptotic properties. A number of simulation studies are presented in Section 4 to demonstrate the finite-sample performance of our methodology. Two real-data examples are discussed in Section 5. These are based on data on user postings on the Sina Weibo platform (the largest Twitter-type social media platform in China), and on $PM_{2.5}$ data recorded across mainland China, respectively. Section 6 concludes the paper. All technical proofs are left to the online Supplementary Material.

2. Grouped Network Vector Autoregression

2.1. Model and notation

The NAR model is defined in (1.1). Here we wish to model the dynamics of $\mathbb{Y}_t$. Note that the effects do not vary by node, implying that all nodes are homogenous. However, as discussed above, this assumption might be too stringent in practice. To address this problem, we assume the nodes in the network can be classified into K groups, where each group is characterized by a specific set of parameters $\theta_k = (\beta_{0k}, \beta_{1k}, \beta_{2k}, \gamma_k^\top)^\top \in \mathbb{R}^{p+3}$, for $1 \le k \le K$. Let $\mathcal{F}_t$ be the σ-field generated by $\{Y_{is} : 1 \le i \le N, 1 \le s \le t\}$. Given $\mathcal{F}_{t-1}$, $Y_{1t}, \ldots, Y_{Nt}$ are assumed to be independent, and to follow a mixture Gaussian distribution

$$\sum_{k=1}^{K} \alpha_k f\left(\beta_{0k} + \beta_{1k} n_i^{-1} \sum_j a_{ij} Y_{j(t-1)} + \beta_{2k} Y_{i(t-1)} + V_i^\top \gamma_k, \sigma_k^2\right), \tag{2.1}$$

where $\alpha_k \ge 0$ satisfying $\sum_{k=1}^{K} \alpha_k = 1$ is the group ratio, and $f(\mu, \sigma^2)$ is the probability density function for a normal distribution with mean μ and variance σ^2. Model (2.1) defines the GNAR model. Essentially, the model specifies a dynamic pattern for each group as a set of parameters. Following the NAR model, we refer to β_{0k}, β_{1k}, β_{2k}, and γ_k as the *grouped* baseline effect, network effect, momentum effect, and nodal effect, respectively.

The model in (2.1) does not specify the group to which each node belongs. Thus, we assume the ith node carries a latent variable $z_{ik} \in \{0, 1\}$. Specifically, $z_{ik} = 1$ if i is from the kth group, otherwise $z_{ik} = 0$. As a result, the GNAR

model (2.1) can be written as

$$Y_{it} = \sum_{k=1}^{K} z_{ik} \left(\beta_{0k} + \beta_{1k} n_i^{-1} \sum_j a_{ij} Y_{j(t-1)} + \beta_{2k} Y_{i(t-1)} + V_i^\top \gamma_k + \sigma_k \varepsilon_{it} \right), \quad (2.2)$$

where ε_{it} is an independent noise term the follows a standard normal distribution. In addition, we can represent the GNAR model in random-coefficient form, as follows:

$$Y_{it} = b_{0i} + b_{1i} n_i^{-1} \sum_j a_{ij} Y_{j(t-1)} + b_{2i} Y_{i(t-1)} + V_i^\top r_i + \delta_i \varepsilon_{it}, \quad (2.3)$$

where $b_{ji} = \sum_k z_{ik} \beta_{jk}$, for $0 \le j \le 2$, $r_i = \sum_k z_{ik} \gamma_k$, and $\delta_i = \sum_{ik} z_{ik} \sigma_k$. Note that (2.3) can be viewed as a generalized extension of the NAR model. There are two main differences between the models. First, the effects (i.e., coefficients) are all node-specific, reflecting the heterogenous characteristics of each node. Second, the parameters are all random (i.e., a linear combination of the latent variables z_{ik}). This makes the GNAR model (2.3) more flexible and realistic in practice.

Remark 1. The GNAR model (2.3) considers only one lag of information. As a flexible extension, one could consider the GNAR(p) model, which considers additional historical information, as follows:

$$Y_{it} = b_{0i} + \sum_{m=1}^{q} b_{1i}^{(m)} n_i^{-1} \sum_{j=1}^{N} a_{ij} Y_{j(t-m)} + \sum_{m=1}^{p} b_{2i}^{(m)} Y_{i(t-m)} + V_i^\top r_i + \delta_i \varepsilon_{it}, \quad (2.4)$$

where $b_{1i}^{(m)} = \sum_k z_{ik} \beta_{1k}^{(m)}$ and $b_{2i}^{(m)} = \sum_k z_{ik} \beta_{2k}^{(m)}$. The theoretical properties and estimation methods can be extended to the GNAR(p) model in (2.4) in a similar manner. In this work, we focus on the GNAR model with one lag, for simplicity.

Recall $\mathbb{Y}_t = (Y_{1t}, \ldots, Y_{Nt})^\top \in \mathbb{R}^N$ is the vector of responses at time t. Let $D_k = \mathrm{diag}\{z_{ik} : 1 \le i \le N\} \in \mathbb{R}^{N \times N}$, with $1 \le k \le K$. Furthermore, define $\mathbb{V} = (V_1, \ldots, V_N)^\top \in \mathbb{R}^{N \times p}$ and $\mathcal{B}_0 = \sum_{k=1}^K D_k (B_{0k} + \mathbb{V}\gamma_k) \in \mathbb{R}^N$, where $B_{0k} = \beta_{0k} \mathbf{1} \in \mathbb{R}^N$ and $\mathbf{1} = (1, \ldots, 1)^\top$ with compatible dimension. Similarly, write $\mathcal{B}_1 = \sum_{k=1}^K D_k B_{1k} \in \mathbb{R}^{N \times N}$ and $\mathcal{B}_2 = \sum_{k=1}^K D_k B_{2k} \in \mathbb{R}^{N \times N}$, where $B_{jk} = \beta_{jk} I \in \mathbb{R}^{N \times N}$, for $j = 1, 2$, and I is the identity matrix with compatible dimension. Then, the GNAR model can be written in vector form as

$$\mathbb{Y}_t = \mathcal{B}_0 + \mathcal{G} \mathbb{Y}_{t-1} + \mathcal{E}_t, \quad (2.5)$$

where $\mathcal{G} = \mathcal{B}_1 W + \mathcal{B}_2$, $W = \text{diag}\{n_1^{-1}, \ldots, n_N^{-1}\}A$ is a row-normalized adjacency matrix, and $\mathcal{E}_t = (\delta_1 \varepsilon_{1t}, \ldots, \delta_N \varepsilon_{Nt})^\top \in \mathbb{R}^N$ is a noise vector.

2.2. Strict stationarity of the GNAR model

In this section, we examine the strict stationarity of the GNAR model. When N is fixed, we have the following theorem.

Theorem 1. *Assume $E\|V_i\| < \infty$ and N is fixed. If $\max_{1\le k\le K}(|\beta_{1k}|+|\beta_{2k}|) < 1$, then there exists a unique stationary solution $\{\mathbb{Y}_t\}$ with $E\|\mathbb{Y}_t\| < \infty$ to the GNAR model* (2.5)*. The solution takes the form:*

$$\mathbb{Y}_t = (I - \mathcal{G})^{-1}\mathcal{B}_0 + \sum_{j=0}^{\infty} \mathcal{G}^j \mathcal{E}_{t-j}. \tag{2.6}$$

The proof of Theorem 1 is given in Section 2 of the Supplementary Material.

Remark 2. Given the group label $\boldsymbol{Z} = \{z_{ik} : 1 \le i \le N, 1 \le k \le K\}$, define the conditional expectation of $\mathbb{Y}_t$ as $\mu_Y = E(\mathbb{Y}_t|\boldsymbol{Z}) = (I - \mathcal{G})^{-1}b_0$, where $b_0 = (b_{01}, \ldots, b_{0N})^\top \in \mathbb{R}^N$. More specifically, denote $\mu_Y = (\mu_1, \ldots, \mu_N)^\top \in \mathbb{R}^N$. As discussed earlier, Y_{it} may, for example, denote the number of posts a node makes on a social network platform. As a result, μ_Y can be viewed as the nodal activeness level. Furthermore, let $\mathcal{M}_k = \{i_1, \ldots, i_{N_k}\}$ be the collection of node indices for the kth group, and $|\mathcal{M}_k| = N_k$ denote the group size. It can be verified that the conditional expectation for nodes belonging to the same group is identical; that is, $\mu_{i_1} = \cdots = \mu_{i_{N_k}} = \nu_k$.

Remark 3. In addition to the conditional mean, we also study the conditional covariance of $\mathbb{Y}_t$. For any integer h, define the auto covariance function of $\mathbb{Y}_t$, given $\boldsymbol{Z}$, as $\Gamma(h) = \text{cov}(\mathbb{Y}_t, \mathbb{Y}_{t-h}|\boldsymbol{Z})$. It can be verified that $\Gamma(0) = (I - \mathcal{G})^{-1}\Sigma_{\mathbb{V}}(I - \mathcal{G}^\top)^{-1} + \Sigma_{\mathcal{E}}$, where $\Sigma_{\mathbb{V}} = \text{diag}\{\sum_{k=1}^K z_{ik}(\gamma_k^\top \Sigma_V \gamma_k) : 1 \le i \le N\}$, with $\Sigma_V = \text{cov}(V_1)$, and $\text{vec}(\Sigma_{\mathcal{E}}) = (I - \mathcal{G}\otimes\mathcal{G})^{-1}\text{vec}(\Sigma_e)$, with $\Sigma_e = \text{diag}\{\sum_{k=1}^K z_{ik}\sigma_k^2 : 1 \le i \le N\}$. It can be further verified that $\Gamma(h) = \mathcal{G}^h\Gamma(0)$, for $h > 0$, and $\Gamma(h) = \Gamma(0)(\mathcal{G}^\top)^{-h}$, for $h < 0$.

To better understand (2.6), we consider a special network structure, namely, the "core-periphery" network. Specifically, there are two groups of nodes in this kind of network: the core (i.e., group 1) and the periphery (i.e., group 2). Nodes in the core group are often celebrities with many followers, whereas nodes in the periphery group tend to have very few followers and are influenced by the nodes in the core group. Figure 2 shows the core-periphery network.

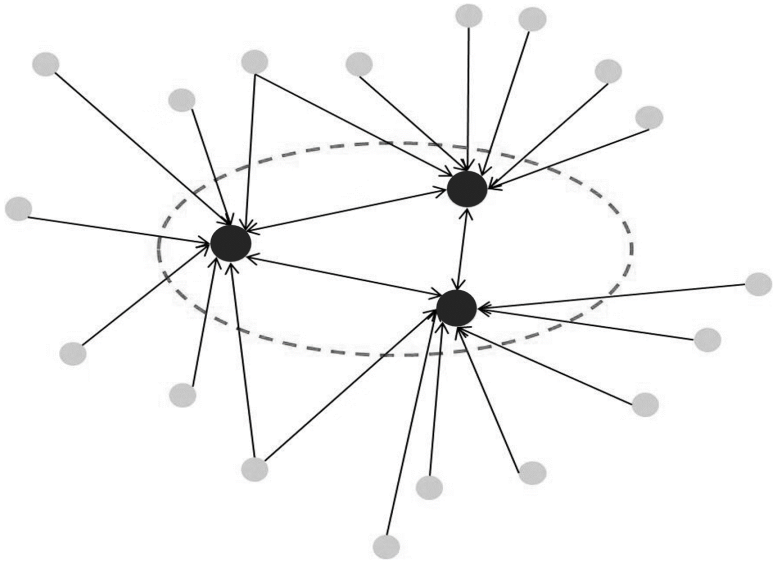

Figure 2. The core-periphery network structure. A black circle represents a core node, and a gray circle denotes a peripheral node. The core and the periphery can be viewed as two different groups, with their own regression coefficients. An arrow indicates the direction of the relationship.

Without loss of generality, let the first s nodes be the first group, and the remaining $N - s$ form the other. Accordingly, let $W = (W_{11}, W_{12}; W_{21}, W_{22})$ be the partition of the two groups. Edges are seldom observed from the core to the periphery, or among the periphery. Accordingly, we set $W_{12} = \mathbf{0}$ and $W_{22} = \mathbf{0}$. The conditional expectation for each group can be computed analytically as $\nu_1 = \beta_{01}/(1-\beta_{21}-\beta_{11})$ and $\nu_2 = (1-\beta_{22})^{-1}(\beta_{02}+\beta_{12}\nu_1)$, respectively. In such a case, the conditional mean for the core is determined only by its own coefficients (i.e., β_{01}, β_{11}, and β_{21}). However, the activeness level of the periphery is also influenced by the core through the term $\beta_{12}\nu_1$.

3. Parameter Estimation

In this section, we discuss the estimation of the GNAR model. Note that the group label z_{ik} is latent. Therefore, the parameter estimation and group detection need to be conducted at the same time. Because the procedure might not be straightforward, as a starting point, we assume the group label is known. In fact, this can be useful when the groups are predetermined by some preliminary knowledge.

3.1. Estimation when group label is known

Define $\mathbb{Y}_t^{(k)} = (Y_{it} : i \in \mathcal{M}_k)^\top \in \mathbb{R}^{N_k}$, $W^{(k)} = (w_{ij} : i \in \mathcal{M}_k, 1 \le j \le N) \in \mathbb{R}^{N_k \times N}$, $\mathbb{V}^{(k)} = (V_i : i \in \mathcal{M}_k)^\top \in \mathbb{R}^{N_k \times p}$, and $\mathcal{E}_t^{(k)} = (\varepsilon_{it} : i \in \mathcal{M}_k)^\top \in \mathbb{R}^{N_k}$. Then, the GNAR model (2.3) can be rewritten as

$$\mathbb{Y}_t^{(k)} = \beta_{0k} + \beta_{1k} W^{(k)} \mathbb{Y}_{t-1} + \beta_{2k} \mathbb{Y}_{t-1}^{(k)} + \mathbb{V}^{(k)} \gamma_k + \sigma_k \mathcal{E}_t^{(k)}, \tag{3.1}$$

for $k = 1, \ldots, K$. Let $X_{it} = (1, w_i^\top \mathbb{Y}_t, Y_{it}, V_i^\top)^\top \in \mathbb{R}^{p+3}$, where w_i is the ith row of W. Furthermore, let $\mathbb{X}_t^{(k)} = (X_{it}^\top : i \in \mathcal{M}_k) \in \mathbb{R}^{N_k \times (p+3)}$. Recall that $\theta_k = (\beta_{0k}, \beta_{1k}, \beta_{2k}, \gamma_k^\top)^\top \in \mathbb{R}^{p+3}$. Then, (3.1) can be written as $\mathbb{Y}_t^{(k)} = \mathbb{X}_t^{(k)} \theta_k + \sigma_k \mathcal{E}_t^{(k)}$. Subsequently, the ordinary least squares (OLS) estimator can be obtained for the kth group, as

$$\widehat{\theta}_k = \left(\sum_{t=1}^{T} \mathbb{X}_{t-1}^{(k)\top} \mathbb{X}_{t-1}^{(k)} \right)^{-1} \left(\sum_{t=1}^{T} \mathbb{X}_{t-1}^{(k)\top} \mathbb{Y}_t^{(k)} \right). \tag{3.2}$$

Next, we investigate the asymptotic properties of $\widehat{\theta}_k$.

Define $\mu_Y^{(k)} = (\mu_i : i \in \mathcal{M}_k)^\top \in \mathbb{R}^{N_k}$. In addition, let $\Sigma_Y = \Gamma(0) = (\sigma_{y,ij}) \in \mathbb{R}^{N \times N}$, $\Sigma_Y^{(k)} = (\sigma_{y,ij} : i \in \mathcal{M}_k, 1 \le j \le N) \in \mathbb{R}^{N_k \times N}$, and $\Sigma_Y^{(k,k)} = (\sigma_{y,ij} : i \in \mathcal{M}_k, j \in \mathcal{M}_k) \in \mathbb{R}^{N_k \times N_k}$. The following technical conditions are required.

(C1) (Group Size) Assume that $\min_k N_k = O(N^\delta)$, where $0 < \delta \le 1$.

(C2) (Independence Assumption) Assume that V_i are independent and identically distributed (i.i.d.) random vectors, with $E(V_1) = \mathbf{0}$, $\mathrm{cov}(V_1) = \Sigma_V \in \mathbb{R}^{p \times p}$, and a finite fourth-order moment. Assume ε_{it} are i.i.d. In addition, assume $\{V_i\}$ and $\{\varepsilon_{it}\}$ are mutually independent.

(C3) (Network Structure) Assume W is a sequence of matrices indexed by N, which we assume to be nonstochastic.

(C3.1) (Connectivity) Treat W as a transition probability matrix of a Markov chain, where the state space is the set of all nodes in the network (i.e., $\{1, \ldots, N\}$). Suppose the Markov chain is irreducible and aperiodic. In addition, define $\pi = (\pi_1, \ldots, \pi_N)^\top \in \mathbb{R}^N$ as the stationary distribution of the Markov chain, such that (a) $\pi_i \ge 0$, with $\sum_{i=1}^N \pi_i = 1$, and (b) $\pi = W^\top \pi$. Furthermore, $\sum_{i=1}^N \pi_i^2$ is assumed to converge to 0 as $N \to \infty$.

(C3.2) (Uniformity) Define $W^* = W + W^\top$ as a symmetric matrix. Assume $\lambda_{\max}(W^*) = O(\log N)$ and $\lambda_{\max}(WW^\top) = O(N^{\delta'})$, for $\delta' < \delta$, where

$\lambda_{\max}(M)$ denotes the largest absolute eigenvalue of an arbitrary symmetric matrix M, and δ is defined in (C1).

(C4) (Law of Large Numbers) Assume the following limits exist: $c_{1\beta}^{(k)} = \lim_{N_k\to\infty} N_k^{-1}(\mathbf{1}^\top W^{(k)}\mu_Y)$, $c_{2\beta}^{(k)} = \lim_{N_k\to\infty} N_k^{-1}(\mathbf{1}^\top \mu_Y^{(k)})$, $\Sigma_1^{(k)} = \lim_{N_k\to\infty} N_k^{-1}\{\mu_Y^{(k)\top}\mu_Y^{(k)} + \mathrm{tr}(W^{(k)\top}W^{(k)}\Sigma_Y)\}$, $\Sigma_2^{(k)} = \lim_{N_k\to\infty} N_k^{-1}\{(\mu_Y^{(k)\top}W^{(k)}\mu_Y) + \mathrm{tr}(W^{(k)}\Sigma_Y^{(k)\top})\}$, and $\Sigma_3^{(k)} = \lim_{N_k\to\infty} N_k^{-1}\{(\mu_Y^{(k)\top}\mu_Y^{(k)}) + \mathrm{tr}(\Sigma_Y^{(k,k)})\}$, for $k = 1, \ldots, K$.

Condition (C1) is an assumption on the group size that the diverging speed of all groups should be at least faster than $O(N^\delta)$ for $\delta > 0$. Note that an unbalanced group size is allowed, which is useful in practice. Condition (C2) is a regular assumption imposed on the nodal covariates Z_i and the noise term ε_{it}. Condition (C3) sets constraints on the network structure W. Specifically, Condition (C3.1) requires that a certain extent of connectivity should exist for the network. Here, a sufficient condition for the irreducibility of the Markov chain is that there should exist a path with finite length between two arbitrary nodes. Condition (C3.2) restricts the heterogeneity of the nodes in the network, requiring that the divergence rate of $\lambda_{\max}(W^*)$ and $\lambda_{\max}(WW^\top)$ should not be too fast. Lastly, Condition (C4) describes the law of large numbers condition for each group, and assumes that the limits of certain network features exist as $N_k \to \infty$, for $k = 1, \ldots, K$.

Theorem 2. *Assume* $\max_k(|\beta_{1k}| + |\beta_{2k}|) < 1$ *and that Conditions* (C1)–(C4) *hold. Then, we have* $\sqrt{N_k T}(\widehat{\theta}_k - \theta_k) = O_p(1)$ *as* $\min\{N_k, T\} \to \infty$.

The proof of Theorem 2 is given in Section 3 of the Supplementary Material. Note that the $\sqrt{N_k T}$-consistency can be obtained for the estimator $\widehat{\theta}_k$ from Theorem 2.

3.2. An EM algorithm

Although the OLS estimation in (3.2) is simple and straightforward, it can be limited if the group label is unknown. Recall that the latent variable $z_{ik} \in \{0, 1\}$ indicates whether the ith user belongs to the kth group. Denote Θ as the parameter space. The full likelihood function is given as

$$L(\Theta) = \prod_{i=1}^{N}\prod_{k=1}^{K}\left[\prod_{t=1}^{T}\alpha_k\phi\big\{\sigma_k^{-1}(Y_{it} - X_{it}^\top\theta_k)\big\}\right]^{z_{ik}}, \tag{3.3}$$

where $\phi(\cdot)$ is the probability density function of the standard normal distribution. We then adopt an EM algorithm for the parameter estimation. After setting an initial value $\widehat{\theta}^{(0)}$, we follow the procedure described below. Specifically, in the mth ($m \geq 1$) iteration, we have the following step:

E-Step. Estimate z_{ik} by its posterior mean $z_{ik}^{(m)}$. Here,

$$z_{ik}^{(m)} = E\big(z_{ik}|\widehat{\theta}^{(m-1)}\big) = \frac{\widehat{\alpha}_k^{(m-1)} \prod_{t=1}^{T} \phi(\widehat{\Delta}_{it,k}^{(m-1)})}{\sum_{k=1}^{K} \widehat{\alpha}_k^{(m-1)} \prod_{t=1}^{T} \phi(\widehat{\Delta}_{it,k}^{(m-1)})}, \tag{3.4}$$

where $\widehat{\Delta}_{it,k}^{(m-1)} = (Y_{it} - X_{i(t-1)}^{\top}\widehat{\theta}_k^{(m-1)})/\widehat{\sigma}_k^{(m-1)}$, and $\widehat{\theta}_k^{(m-1)}$ and $\widehat{\sigma}_k^{(m-1)}$ are the estimates from the $(m-1)$th iteration.

M-Step. Given $z_{ik}^{(m)}$, we then maximize (3.3) with respect to α_k, θ_k, and σ_k. Specifically, we have

$$\widehat{\theta}_k^{(m)} = \left(\sum_i z_{ik}^{(m)} \sum_t X_{it}X_{it}^{\top}\right)^{-1} \left(\sum_i z_{ik}^{(m)} \sum_t X_{it}Y_{it}\right), \tag{3.5}$$

$$\begin{aligned} \big(\widehat{\sigma}_k^2\big)^{(m)} &= \left(T\sum_i z_{ik}\right)^{-1} \left\{\sum_i z_{ik}^{(m)} \sum_t (Y_{it} - X_{it}^{\top}\widehat{\theta}_k^{(m)})^2\right\}, \\ \widehat{\alpha}_k^{(m)} &= N^{-1}\left(\sum_i^N z_{ik}^{(m)}\right). \end{aligned} \tag{3.6}$$

Repeat the above steps until the EM algorithm converges. The final results are the desired estimators.

Note that the estimation given in (3.5) is similar in spirit to (3.2). In particular, the EM estimation of θ_k can be treated as a weighted OLS estimator, where the weights are the latent group variables z_{ik}. In addition, the estimations of σ_k^2 and α_k in (3.6) can be viewed in a similar way.

3.3. A TS estimation method

In practice, the computation of the E-Step (3.4) might not be stable when the time dimension T is large. As a result, the estimation result in the M-Step might not be reliable. Note that (2.3) can be treated as a random coefficient model with node-specific coefficients. Motivated by this fact, we consider a TS estimation procedure as an alternative. In the first step, we estimate the coefficient at the nodal level. Then, we pool these estimates to obtain the parameter estimation $\widehat{\theta}_k$, for $k = 1, \ldots, K$. For convenience, we assume $(\beta_{1k}, \beta_{2k})^{\top}$ are not

the same between groups.

Let $b_i = (b_{0i} + V_i^\top \gamma_i, b_{1i}, b_{2i})^\top \in \mathbb{R}^3$. Write $\boldsymbol{X}_{it} = (1, w_i^\top \mathbb{Y}_t, Y_{it})^\top \in \mathbb{R}^3$. Then, the estimates for b_i can be obtained as

$$\widehat{b}_i = \left(\sum_{t=1}^{T} \boldsymbol{X}_{i(t-1)} \boldsymbol{X}_{i(t-1)}^\top \right)^{-1} \left(\sum_{t=1}^{T} \boldsymbol{X}_{i(t-1)} Y_{it} \right). \tag{3.7}$$

Note that (3.7) is the OLS estimation for each node. Intuitively, this estimate will approximate the true value b_i well when T is sufficiently large.

Theorem 3. *Assume $N = o(\exp(T))$, the stationary condition $\max_k(|\beta_{1k}| + |\beta_{2k}|) < 1$, and Conditions (C1)–(C4) hold. In addition, assume there exists $\tau > 0$, such that $\min_i\{(e_i^\top \Sigma_Y e_i)(w_i^\top \Sigma_Y w_i) - (e_i^\top \Sigma_Y w_i)^2\} \geq \tau$, with probability tending to one. Then, we have $\sup_{1 \leq i \leq N} \|\widehat{b}_i - b_i\| = o_p(1)$.*

The proof of Theorem 3 is given in Section 4 of the Supplementary Material. The above term, $(e_i^\top \Sigma_Y e_i)(w_i^\top \Sigma_Y w_i) - (e_i^\top \Sigma_Y w_i)^2$, can be rewritten as $\sum_i \sum_{j_1, j_2} \Delta_{ij_1j_2} w_{ij_1} w_{ij_2} (\widetilde{\sigma}_{y,j_1j_2} - \widetilde{\sigma}_{y,ij_1} \widetilde{\sigma}_{y,ij_2})$, where $\Delta_{ij_1j_2} = \sigma_{y,ii}\sigma_{y,j_1j_1}\sigma_{y,j_2j_2}$ and $\widetilde{\sigma}_{y,ij} = \mathrm{cor}(Y_{it}, Y_{jt})$. Then, the condition is satisfied if $\sigma_{y,ii}$ and $\widetilde{\Delta}_{ij_1j_2} = \widetilde{\sigma}_{y,j_1j_2} - \widetilde{\sigma}_{y,ij_1}\widetilde{\sigma}_{y,ij_2}$ are lower bounded away from zero, with probability tending to one for the triplets set $\{(i, j_1, j_2) : a_{ij_1} a_{ij_2} = 1, i \neq j_1, i \neq j_2\}$. Given the results in Theorem 3, the overall estimation bias (i.e., $\sup_i \|\widehat{b}_i - b_i\|$) can be controlled, because the time T diverges slightly faster than $\log(N)$ (i.e., log-transformed network size) does.

Based on the theoretical result of Theorem 3, we consider the second step of the estimation. Ideally, the estimated values $\widehat{b}_i$ will form K clusters (i.e., groups) as the output of the cluster algorithm. The corresponding group members are collected in $\widehat{\mathcal{M}}_k$, where $\widehat{N}_k = |\widehat{\mathcal{M}}_k|$. Then, the group ratio α_k can be estimated directly by $\widehat{\alpha}_k = \widehat{N}_k / N$. Subsequently, given this estimated group information, we can conduct the estimation using the procedure in (3.2) in Section 3.1. Specifically, θ_k can be estimated by

$$\widehat{\theta}_k^{TS} = \left(\sum_{t=1}^{T} \sum_{i \in \widehat{\mathcal{M}}_k} X_{i(t-1)} X_{i(t-1)}^\top \right)^{-1} \left(\sum_{t=1}^{T} \sum_{i \in \widehat{\mathcal{M}}_k} X_{i(t-1)} Y_{it} \right),$$

which is referred to as the TS estimator. Theoretically, one would expect a consistency result for $\widehat{\theta}_k^{TS}$ if all nodes are clustered into their true groups with proba-

Table 1. Parameter setup for Examples 1–3 in the simulation study.

	α	β_0	β_1	β_2	γ
Example 1 & 2					
Group 1	0.2	0.0	0.1	0.3	$(0.5, 0.7, 1.0, 1.5, -1.0)^\top$
Group 2	0.3	0.2	-0.3	0.2	$(0.1, 0.9, 0.4, -0.2, -1.5)^\top$
Group 3	0.5	0.5	0.2	0.7	$(0.2, -0.2, 1.4, -0.8, 0.5)^\top$
Example 3					
Group 1	0.2	5.0	0.2	0.1	$(0.5, 0.7, 1.0, 1.5, -1.0)^\top$
Group 2	0.3	-5.0	-0.4	0.2	$(0.1, 0.9, 0.4, -0.2, -1.5)^\top$
Group 3	0.5	0.0	0.2	0.4	$(0.2, -1.0, 2.0, 3.0, -2.0)^\top$

bility tending to one (Hartigan (1981); Pollard (1981); Von Luxburg, Belkin and Bousquet (2008)). This is guaranteed by the result of Theorem 3 when abundant time information can be obtained.

4. Numerical Studies

4.1. Simulation models

To demonstrate the finite-sample performance of our proposed methodology, we conduct a number of numerical studies in this section. Specifically, the first two examples are presented with different types of network structures. The third example investigates the parameter estimation and prediction accuracy when the number of groups is misspecified. In each example, different estimation methods (EM and TS) are employed and compared.

For each example, we fix the number of groups as $K = 3$, and generate the random innovations ε_{it} from a standard normal distribution. For convenience, we set $\delta_k = 1$, for $k = 1, \ldots, K$. In addition, the nodal covariates $V_i = (V_{i1}, \ldots, V_{i5})^\top \in \mathbb{R}^5$ are independently sampled from a multivariate normal distribution with mean $\mathbf{0}$ and covariance $\Sigma_v = (\sigma_{j_1 j_2})$, with $\sigma_{j_1 j_2} = 0.5^{|j_1 - j_2|}$. The true values of the parameters for each group are listed in Table 1. Furthermore, let $\sigma^2 = 1$ for Examples 1 and 2, and let $\sigma^2 = 4$ for Example 3. Given the initial value $\mathbb{Y}_0 = \mathbf{0}$, the time series $\mathbb{Y}_t$ is generated according to the GNAR model in (2.3). Here, the first 50 replications are dropped to ensure the time series achieves stationarity.

Note that different network and momentum effects are employed for each group in order to distinguish nodal behaviors. As shown in Table 1, Group 1 has a relatively lower activeness level, with small, positive network and momentum effects (i.e., β_1 and β_2). Group 2 is characterized by a negative network effect

(i.e., β_1), implying that nodal behaviors in this group exhibit a negative correlated pattern with their connected friends. Lastly, compared with the other two groups, Group 3 occupies a larger portion (i.e., α) and has a higher momentum effect (i.e., β_2). Next, we introduce two typical network structures employed in the simulation studies.

Example 1. (Stochastic Block Model) First, we consider the block structure network, also known as the stochastic block model (Wang and Wong (1987); Nowicki and Snijders (2001); Zhao, Levina and Zhu (2012)). This model assumes that nodes in the same block are more likely to be connected. To generate the model, we follow Zhu et al. (2017), setting $J \in \{5, 10, 20\}$ blocks, and randomly assigning each node a block label with equal probability. Next, let $P(a_{ij} = 1) = 0.3N^{-0.3}$ if i and j are from the same block, otherwise set $P(a_{ij} = 1) = 0.3N^{-1}$. Consequently, nodes within the same block have a higher probability of connecting with each other than they do of connecting with nodes from other blocks.

Example 2. (Power-law Model) In real networks, a small portion of nodes (e.g., superstars and opinion leaders) have many network links, whereas the majority tend to have few connections. This phenomenon is described by the power-law model (Barabási and Albert (1999)). Specifically, we generate the nodal in-degrees $d_i = \sum_j a_{ji}$ from a power-law distribution; that is, $P(d_i = d) = cd^{-\alpha}$, where c is a normalizing constant and α is the exponent parameter. We set $\alpha = 2.5$, as suggested by Clauset, Shalizi and Newman (2009), which is based on empirical studies of real social network data.

Example 3. (Number of Groups) In this example, we evaluate the impact on the parameter estimation and prediction accuracy when the number of groups K is incorrectly specified. Specifically, data are generated using the power-law model described in Example 2, with total time periods $(T + 20)$. The first T periods are used for the parameter estimation, and the remaining 20 periods are used for prediction. Lastly, we set $K = 1, 2, 3, 5, 7$, where $K = 3$ is the true number of groups.

4.2. Performance measurements and simulation results

For each simulation example, we consider network sizes $N = 100, 200, 500$. Accordingly, to evaluate the performance of the proposed estimation methods, we employ two settings of T: $T = N/2$, and $T = 2N$. To ensure a reliable result, we randomly repeat the simulation experiments $R = 1,000$ times. Let $(\widehat{\beta}_{0k}^{(r)}, \widehat{\beta}_{1k}^{(r)}, \widehat{\beta}_{2k}^{(r)}, \widehat{\gamma}_k^{(r)\top})^\top \in \mathbb{R}^{p+3}$ be the estimator of the kth group obtained from

the rth replication. In addition, for each node, we obtain its group label as $\widehat{z}_i^{(r)}$, for $i = 1, \ldots, N$. Specifically, for the EM algorithm, the group label is defined as $\widehat{z}_i^{(r)} = \arg\max_k\{\widehat{z}_{ik}\}$. For the TS estimation, the group label is the same as the cluster label after the first-step estimation. Next, we consider measurements with which to evaluate the numerical results.

First, for a given parameter, the root mean squared error (RMSE) is employed to evaluate the estimation accuracy. For example, consider the network effect $\beta_1 = (\beta_{1k} : 1 \le k \le K)^\top \in \mathbb{R}^K$. The RMSE is calculated over all groups as $\mathrm{RMSE}_{\beta_j} = \{(RK)^{-1}\sum_{k=1}^K\sum_{r=1}^R(\widehat{\beta}_{jk}^{(r)} - \beta_{jk})^2\}^{1/2}$. Similarly, the RMSE can be computed for the baseline effect (i.e., RMSE_{β_0}) and the momentum effect (i.e., RMSE_{β_2}). In addition, the RMSE for the nodal effect is defined as $\mathrm{RMSE}_\gamma = \{(RK)^{-1}\sum_{k=1}^K\sum_{r=1}^R\|\widehat{\gamma}_k^{(r)} - \gamma_k\|^2\}^{1/2}$. Next, given the estimated groups $\widehat{z}_i^{(r)}$, the misclassification rate (MCR) can be calculated as $\mathrm{MCR} = (NR)^{-1}\sum_{r=1}^R\sum_{i=1}^N I(\widehat{z}_i^{(r)} \ne z_i)$, where z_i is the true group label of node i. Lastly, the average network density (i.e., $\{N(N-1)\}^{-1}\sum_{i_1,i_2} a_{i_1i_2}$) is also reported.

When the number of groups K is misspecified (e.g., Example 3), we evaluate the impact on the parameter estimation and prediction accuracy. Denote $\widehat{\mathbb{Y}}_t^{(K)}$ as the fitted response for $t = 1, \ldots, T$ and the predicted value for $t = T+1, \ldots, T+20$, where the superscript K indicates the number of groups. In order to evaluate the parameter estimation accuracy, we compare the fitted value $\widehat{\mathbb{Y}}_t^{(K)}$ to the conditional expectation $E(\mathbb{Y}_t|\mathcal{F}_{t-1}, \boldsymbol{Z})$. This is because the comparison cannot be conducted directly for the parameter estimation error when the number of groups K is misspecified. Thus, we define the estimation error as

$$\mathrm{Err}_{est}^{(K)} = \left\{(NT)^{-1}\sum_{t=1}^T\left\|\widehat{\mathbb{Y}}_t^{(K)} - E(\mathbb{Y}_t|\mathcal{F}_{t-1}, \boldsymbol{Z})\right\|^2\right\}^{1/2},$$

where $\mathcal{F}_{t-1}$ is the σ-field generated by $\{\mathbb{Y}_s : s \le t-1\}$, and $E(\mathbb{Y}_t|\mathcal{F}_{t-1}, \boldsymbol{Z})$ is the conditional expectation based on the historical and group information. Next, the prediction error is measured by

$$\mathrm{Err}_{pred}^{(K)} = \left\{(20N)^{-1}\sum_{t=T+1}^{T+20}\left\|\widehat{\mathbb{Y}}_t^{(K)} - \mathbb{Y}_t\right\|^2\right\}^{1/2},$$

which is the RMSE for the predicted values. The median values of $\mathrm{Err}_{est}^{(K)}$ and $\mathrm{Err}_{pred}^{(K)}$ over all replications are reported.

The detailed results are given in Tables 2–4. For the first two examples, we

Table 2. Simulation results with 1,000 replications for the stochastic block model. The RMSE ($\times 10^2$) is reported for the EM and TS estimations. The network density (ND) and misclassification rate (MCR) are reported as percentages (%).

N	Est.	α	β_0	β_1	β_2	γ	ND	MCR
Scenario 1. $T = N/2$								
100	EM	3.63	30.80	10.96	14.56	49.64	2.2	11.1
	TS	8.92	110.00	28.13	38.91	175.10	2.2	42.4
200	EM	2.10	14.86	6.42	11.09	26.54	1.1	3.8
	TS	7.56	46.74	22.19	34.66	75.44	1.1	31.3
500	EM	0.82	7.07	3.06	5.71	11.04	0.4	0.9
	TS	6.72	19.00	12.56	22.58	48.59	0.4	14.7
Scenario 2. $T = 2N$								
100	EM	4.08	41.67	12.24	17.60	56.03	2.2	13.3
	TS	6.65	37.43	13.86	21.51	60.08	2.2	15.0
200	EM	2.49	17.37	6.90	12.48	30.03	1.1	4.7
	TS	4.49	12.33	7.20	11.57	28.34	1.1	4.8
500	EM	1.04	8.82	3.19	6.76	13.95	0.4	1.1
	TS	1.42	3.84	1.42	2.31	7.16	0.4	0.3

find that as the network size N and time period T increase, the RMSEs of all estimated parameters decrease toward zero for both the algorithm and the TS estimation. In addition, a similar pattern can be observed for the MCR, which drops as the network size and time period (i.e., N and T) increase. In the finite-sample comparison, the EM algorithm outperforms the TS estimation in Scenario 1, in which less time information is available (i.e., small T). Specifically, lower RMSE and MCR values are observed. However, the TS estimation outperforms the EM algorithm in Scenario 2 in terms of both parameter estimation and group classification. Lastly, in Example 3, we find that both the estimation and the prediction errors drop sharply from $K \leq 2$ to $K = 3$, where the model is correctly specified with $K = 3$. Furthermore, for $K \geq 3$, it is observed that the estimation and prediction errors perform relatively consistently.

5. Case Study

In this section, we conduct two case studies to evaluate our proposed methods. The first is based on users' posting behavior on a social network platform. The second is based on a study of dynamic and spatial patterns of $PM_{2.5}$. Here, the adjacency matrix between cities is constructed by taking advantage of their spatial locations.

Table 3. Simulation results with 1,000 replications for the power law model. The RMSE ($\times 10^2$) is reported for the EM and TS estimations. The network density (ND) and misclassification rate (MCR) are reported as percentages (%).

N	Est.	α	β_0	β_1	β_2	γ	ND	MCR
Scenario 1. $T = N/2$								
100	EM	3.21	28.42	9.69	12.75	43.40	2.3	9.4
	TS	14.22	72.19	39.86	35.14	116.84	2.3	32.0
200	EM	1.74	13.15	5.67	9.86	23.44	1.2	3.5
	TS	12.08	34.17	27.13	27.83	64.49	1.2	18.0
500	EM	0.78	5.94	2.67	5.55	11.00	0.5	0.8
	TS	7.15	15.46	12.04	13.17	32.13	0.5	4.5
Scenario 2. $T = 2N$								
100	EM	3.79	36.09	11.19	16.27	50.06	2.3	12.0
	TS	6.15	14.07	10.01	13.95	30.63	2.3	4.4
200	EM	2.33	17.64	6.65	11.67	27.50	1.2	4.7
	TS	2.99	6.20	4.00	6.14	14.08	1.2	0.9
500	EM	0.74	5.70	2.42	4.92	10.37	0.5	0.7
	TS	0.02	0.35	0.12	0.39	0.64	0.5	0.0

5.1. User behavior analysis: a Sina Weibo data set

We first apply the proposed GNAR model to a social network data set. The data are collected from Sina Weibo, the largest Twitter-type social media platform in China. Users can follow other users, create profiles, and post Weibo to express their opinions. In addition to ordinary users, celebrities, the media, and organizations may also register on Sina Weibo. The diversity among users leads to varying behavior patterns.

Data Description

To investigate users' behaviour on Weibo, we collect data on $N = 2,021$ followers of an official account for $T = 11$ consecutive weeks, from January 1, 2014. The response Y_{it} is defined as the $\log(1 + x)$-transformed average Weibo post length (i.e., the average number of characters posted by a user per week), which can be viewed as the nodal activeness level. A histogram of the response is displayed in Figure 3, where an approximately symmetric shape can be observed. In addition, two node-specific variables are recorded: the gender of the user (male = 1; female = 0), and the number of personal labels (i.e., keywords created by Weibo users to describe their life status and interests).

The network adjacency matrix A can be constructed as follows: $a_{ij} = 1$ if the ith user follows the jth user on Weibo; otherwise $a_{ij} = 0$. Note that, the

Table 4. Simulation results with 500 replications for different K (number of groups) for the power law distribution network. The true number of groups is set to $K = 3$. The median values of $\mathrm{Err}^{(K)}_{est}$ ($\times 10^2$) and $\mathrm{Err}^{(K)}_{pred}$ are reported.

		Estimation					Prediction				
N	Est.	$K=1$	$K=2$	$K=3$	$K=5$	$K=7$	$K=1$	$K=2$	$K=3$	$K=5$	$K=7$
Scenario 1. $T = N/2$											
100	EM	147.7	111.4	69.1	22.1	25.4	2.48	2.29	2.10	2.02	2.02
	TS	147.7	136.7	129.1	118.7	109.6	2.48	2.42	2.37	2.32	2.28
200	EM	148.0	112.3	8.3	10.0	11.1	2.49	2.29	2.01	2.00	2.00
	TS	148.0	122.2	109.8	95.3	86.6	2.49	2.34	2.29	2.22	2.18
500	EM	148.4	113.4	2.8	3.4	3.9	2.49	2.30	2.00	2.00	2.00
	TS	148.4	105.2	50.1	41.6	38.3	2.49	2.26	2.06	2.04	2.03
Scenario 2. $T = 2N$											
100	EM	147.8	112.2	86.5	10.9	11.3	2.48	2.29	2.16	2.03	2.01
	TS	147.8	104.0	54.8	36.6	26.5	2.48	2.25	2.07	2.04	2.02
200	EM	148.2	112.5	3.8	4.4	4.8	2.49	2.29	2.01	2.01	2.00
	TS	148.2	103.8	3.8	5.5	7.0	2.49	2.25	2.01	2.00	2.00
500	EM	148.2	113.2	1.4	1.7	2.3	2.49	2.29	2.00	2.01	2.02
	TS	148.2	104.1	1.4	2.2	2.9	2.49	2.25	2.00	2.00	2.00

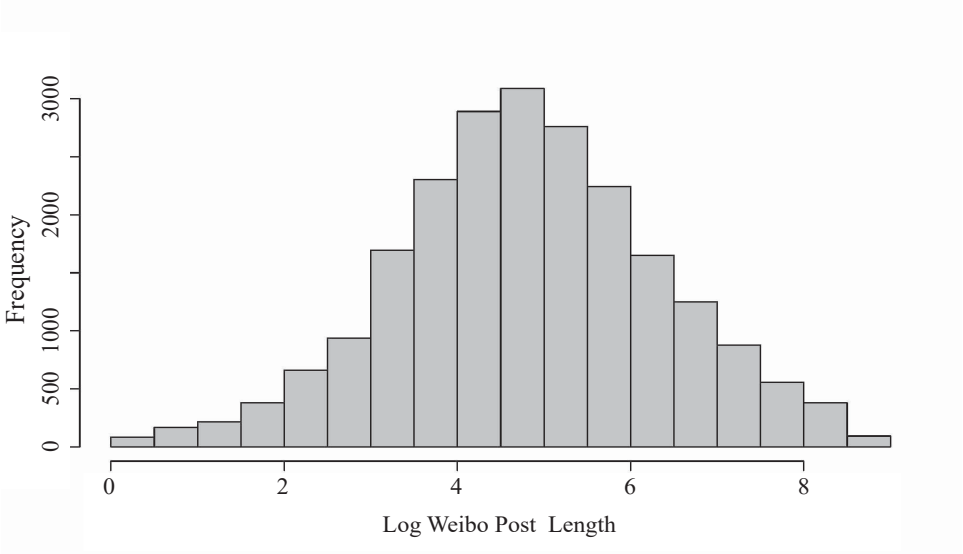

Figure 3. Histogram of responses (i.e., log-transformed Weibo post length).

adjacency matrix is asymmetric, because users are not required to be mutually connected on Weibo. We illustrate the distributions of the nodal in-degree (i.e., $a_{+i} = \sum_j a_{ji}$) and out-degree (i.e., $a_{i+} = \sum_j a_{ij}$) in Figure 4, which shows that the distribution of the in-degree is more skewed than that of the out-degree. This implies that there might exist users who attract many followers. In addition, the network density is 2.7% (i.e., $\sum_{i,j} a_{ij}/\{N(N-1)\}$), which indicates a relatively sparse network.

Model Estimation and Explanation

Next, we fit the GNAR model on this data set. We apply the EM algorithm only, because the network size N is much larger than the number of periods T. The number of groups is fixed as $K = 3$. The estimation results in Table 5 show that the estimated network effect and momentum effect are both positive for the three groups. This suggests that a user's activeness level is positively related to itself, and to that of its neighbors. Moreover, the momentum effect appears to be stronger than the network effect. Lastly, the estimated nodal effects indicate that male users with more self-created labels exhibit higher activeness levels.

For further illustration, we compare the groups. Note that Group 1 and Group 2 include a large portion of all network users (with larger estimated α values). Specifically, they both have larger network effects (i.e., estimated β_1

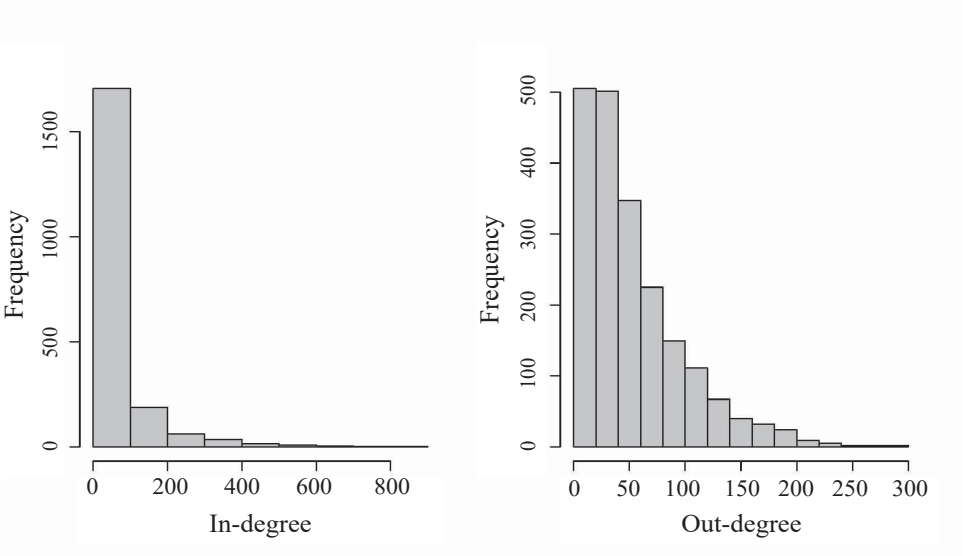

Figure 4. Histogram of nodal in- and out-degree of $N = 2,021$ nodes. A heavily skewed shape can be detected for the nodal in-degree, which indicates the existence of "superstars" in the network.

Table 5. GNAR analysis results for the Sina Weibo data set.

Regression coefficient	Group 1	Group 2	Group 3
Group Ratio (α)	0.447	0.361	0.192
Baseline Effect (β_0)	0.857	1.681	0.236
Network Effect (β_1)	0.031	0.026	0.002
Momentum Effect (β_2)	0.765	0.396	0.958
Gender (γ_1)	0.077	0.155	0.009
Number of Labels (γ_2)	0.006	0.018	0.002

values) than that of Group 3, implying that users in these two groups tend to be influenced by those they follow. With regard to the momentum effect (i.e., the estimated β_2 values), users in Groups 1 and 3 are more self-motivated than those in Group 2 are. In particular, Group 3 has the largest momentum effect, but the smallest network effect. This indicates that the user behavior of this group can be predicted well using historical information.

Moreover, we draw a box plot for the responses in a grouped manner in Figure 5. A higher activeness level can be found for Group 3. Users in this group are mostly media accounts and celebrities with many followers, such as "Sina Finance", "Xinhua Views", "Beijing Youth Daily", "Phoenix TV", and many others. These accounts generate content and release information on the platform

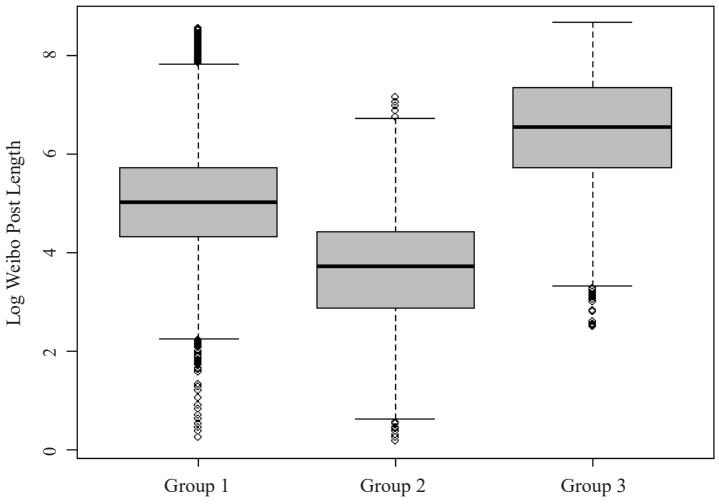

Figure 5. Box plot of log-transformed Weibo post length for each group.

in order to pass on information and influence other users. In contrast, most users in Group 1 and Group 2 are ordinary users, who play the role of information adopters. Lastly, we compare the performance of the proposed model with that of the network vector autoregression model (Zhu et al. (2017)) and the univariate autoregression (AR) model. The first nine weeks are used for model training, and the last two weeks are employed for prediction evaluation. The predictive RMSE is used to quantify the prediction accuracy of each model (0.809, 0.850, and 2.312), respectively. Here, we find that the predictive RMSE of the GNAR is lower than those of the NAR and AR, indicating that the GNAR model exhibits better prediction power.

5.2. Air pollution analysis: a $PM_{2.5}$ data set

In recent years, the issue of air pollution in China has drawn worldwide attention. Of particular concern is $PM_{2.5}$, which refers to airborne particles with an aerodynamic diameter of less than 2.5 μm. There is evidence that a high concentration of $PM_{2.5}$ may cause severe clinical symptoms, such as lung morbidity and respiratory and cardiovascular diseases. Hence, it is of great importance to understand the $PM_{2.5}$ distribution and diffusion pattern across China.

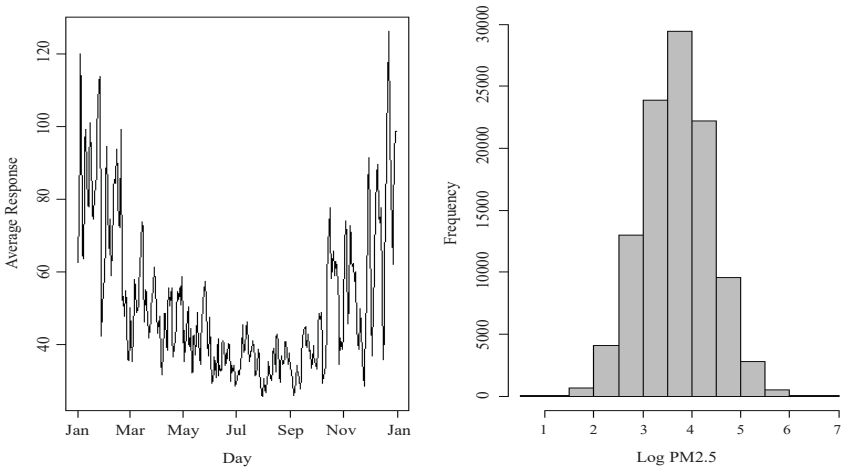

Figure 6. Left panel: daily average $PM_{2.5}$ in 2015; Right panel: histogram of log-$PM_{2.5}$.

Data Description

The $PM_{2.5}$ data are collected from air quality monitoring stations in 291 cities in mainland China. Specifically, the daily $PM_{2.5}$ index (unit: $\mu g/m^3$) is recorded for the period January 1, 2015, to December 31, 2015 with $T = 365$. The left side of Figure 6 gives the time series of the average daily $PM_{2.5}$ of all cities during 2015. A high $PM_{2.5}$ level is evident in November, December, and January, with the highest $PM_{2.5}$ being greater than 100 $\mu g/m^3$. Figure 7 shows the average $PM_{2.5}$ in each city, where darker regions imply higher $PM_{2.5}$ levels. Spatially, the northeastern regions in China (especially in Heibei province) exhibit higher concentrations of $PM_{2.5}$.

The response is defined as the log-transformed $PM_{2.5}$ level; see the histogram displayed on the right-hand side of Figure 6. A symmetric shape can be observed. In order to construct the network structure, we treat each city as a node. The adjacency matrix A is constructed using the spatial distances between two cities. Let $s_1, \ldots, s_N$ ($s_i \in \mathbb{R}^2$) be the locations of N cities. Then, a_{ij} is defined as $a_{ij} = 1/\|s_i - s_j\|$ for $i \neq j$, and $a_{ii} = 0$ for $i = 1, \ldots, N$.

Model Estimation and Explanation

Motivated by the descriptive analysis, we model the dynamic patterns of the seasons separately. We define the seasons as follows: spring (March to May), summer (June to August), autumn (September to November), and winter (January

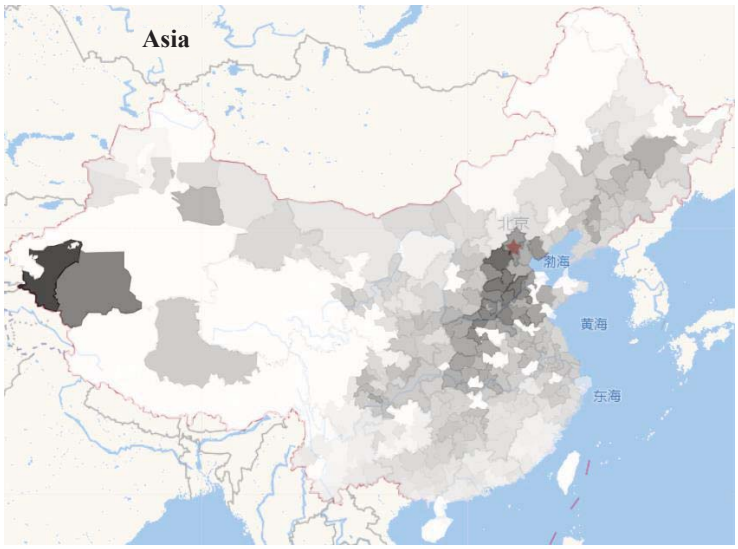

Figure 7. Average $PM_{2.5}$ for each city in 2015. White indicates an absence of $PM_{2.5}$ monitoring stations in corresponding cities.

to February). Intuitively, the number of groups should be large in winter because the pollution level is relatively high. As a result, we set $K = 3$ for winter, and $K = 2$ for the other seasons. The GNAR, NAR, and AR models are estimated in order to compare their predictions. The GNAR model is estimated using the proposed EM algorithm and the TS estimation method. For each season, the last 10 days are used to conduct predictions. The prediction RMSEs are summarized in Table 6. The results show that the EM algorithm always outperforms the other methods in terms of prediction accuracy. We next describe the estimation results for the EM algorithm in great detail.

The estimated regression coefficients are given in Table 7. We focus here on the results for winter. First, the number of cities in the three groups is unbalanced, with proportions of 0.32, 0.39, and 0.29, respectively. Note that the first group has a relatively large baseline effect, indicating that air pollution in these cities is much more severe. Figure 8 shows that cities in group 1 are located in northeastern China. Furthermore, cities in group 2 have large network effects, implying that these cities are more likely to be influenced by their spatial neighbors. With regard to the other seasons, the patterns in summer and autumn are very similar, mainly because the pollution level is relatively low in these two seasons.

There are two further remarks related to this example. First, the network

Table 6. The prediction RMSE for the $PM_{2.5}$ data set using the GNAR model (with EM and TS estimations), NAR model, and AR model.

	GNAR (EM)	GNAR (TS)	NAR	AR
Spring	0.375	0.387	0.388	0.739
Summer	0.328	0.328	0.330	0.941
Autumn	0.439	0.439	0.441	1.122
Winter	0.546	0.565	0.561	0.955

Table 7. Estimation results for the $PM_{2.5}$ data set from the EM algorithm. Two groups are set for spring, summer, and autumn and for winter, the number of groups is $K = 3$.

	Spring		Summer		Autumn		Winter		
Group	1	2	1	2	1	2	1	2	3
Group Ratio (α)	0.61	0.39	0.67	0.33	0.53	0.47	0.32	0.39	0.29
Baseline Effect (β_0)	1.26	0.77	0.46	0.55	0.25	0.41	2.04	0.37	0.25
Network Effect (β_1)	0.14	0.11	0.20	-0.04	0.32	0.11	-0.01	0.33	0.19
Momentum Effect (β_2)	0.55	0.65	0.67	0.87	0.62	0.76	0.52	0.59	0.72

structure is symmetric (i.e., $a_{ij} = a_{ji}$). Recall that when the network structure is asymmetric (as in the social network case), the term $n_i^{-1} \sum_j a_{ij} Y_{j(t-1)}$ represents the averaged responses of the nodes that i follows. As a result, the network effect β_1 can be viewed as the "influence" that i receives from the nodes it follows (i.e., those j with $a_{ij} = 1$). When the adjacency matrix is symmetric, as in this example, the term $n_i^{-1} \sum_j a_{ij} Y_{j(t-1)}$ represents the averaged responses of those nodes to which i is connected. The corresponding parameter β_1 can be understood as the "connection" or "correlation," rather than the "influence" that node i receives from its connected neighbours. Second, in this example, no node-specific covariates are utilized, owing to a lack of data. Thus, future research should incorporate nodal effect variables (i.e., V_i), such as temperature, humidity, and wind speed into the modeling framework.

6. Conclusion

In this study, we develop a novel GNAR model that incorporates group-specific network autoregression coefficients. To estimate the GNAR model, an EM algorithm and a TS estimation are designed. The results suggest that both methods produce consistent results, but vary in terms of their finite-sample performance. Sina Weibo and $PM_{2.5}$ data sets are analyzed for illustration purpose, where the nodes in the different groups show distinct behavioral patterns.

Several directions for future research are possible. First, although we have

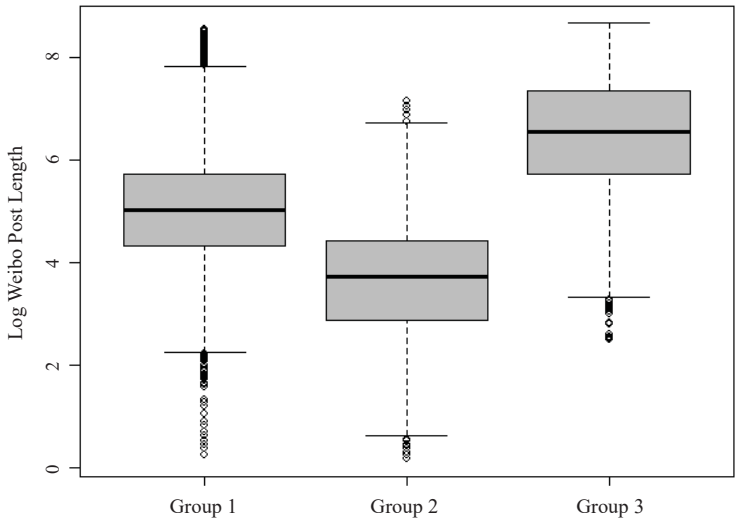

Figure 8. Different groups of cities detected by the EM algorithm for spring (left top panel), summer (right top panel), autumn (left bottom panel), and winter (right bottom panel). Cities in Groups 1, 2, and 3 are marked as light gray, gray, and dark gray, respectively.

developed estimation and group classification procedures, they are not sufficiently flexible for inferences on the estimated parameters. Next, for the proposed estimation methods of the GNAR model, the number of groups K needs to prespecified. Hence, how to select K remains to be a challenging task. Lastly, it is assumed that users can be grouped by their dynamic behavior patterns, which are further quantified by the network autoregression coefficients. As a further extension, one could consider incorporating user network structure information (e.g., the following-followee information of the focal user) to decide their groups.

Supplementary Material

The online Supplementary Materials contains the proofs of Theorems 1 to 3, as well as several useful lemmas.

Acknowledgments

Xuening Zhu was supported by National Nature Science Foundation of China (NSFC, U1911461), Shanghai Sailing Program for Youth Science and Technol-

ogy Excellence (No. 19YF1402700), Fudan Xinzailing joint research centre for big data, School of Data Science, Fudan University. Rui Pan was supported by the National Nature Science Foundation of China (NSFC, 11601539, 11631003, 71771224), China's National Key Research Special Program Grant (No. 2016YFC0207704), the Fundamental Research Funds for the Central Universities (No. QL18010), the Youth Talent Development Support Program (No. QYP1911) and the Program for Innovation Research in Central University of Finance and Economics.

References

Barabási, A.-L. and Albert, R. (1999). Emergence of scaling in random networks. *Science* **286**, 509–512.

Bauwens, L. and Rombouts, J. (2007). Bayesian clustering of many garch models. *Econometric Reviews* **26**, 365–386.

Bohn, A., Buchta, C., Hornik, K. and Mair, P. (2014). Making friends and communicating on facebook: Implications for the access to social capital. *Social Networks* **37**, 29–41.

Clauset, A., Shalizi, C. R. and Newman, M. E. (2009). Power-law distributions in empirical data. *SIAM Review* **51**, 661–703.

Frühwirth-Schnatter, S. and Kaufmann, S. (2006). How do changes in monetary policy affect bank lending? an analysis of austrian bank data. *Journal of Applied Econometrics* **21**, 275–305.

Fröhwirth-Schnatter, S. and Kaufmann, S. (2008). Model-based clustering of multiple time series. *Journal of Business & Economic Statistics* **26**, 78–89.

Hartigan, J. A. (1981). Consistency of single linkage for high-density clusters. *Journal of the American Statistical Association* **76**, 388–394.

Heard, N. A., Holmes, C. C. and Stephens, D. A. (2006). A quantitative study of gene regulation involved in the immune response of anopheline mosquitoes: An application of Bayesian hierarchical clustering of curves. *Journal of the American Statistical Association* **101**, 18–29.

Hofstra, B., Corten, R. and Buskens, V. (2015). Learning in social networks : Selecting profitable choices among alternatives of uncertain profitability in various networks. *Social Networks* **43**, 100–112.

Juárez, M. A. and Steel, M. F. (2010). Model-based clustering of non-gaussian panel data based on skew-t distributions. *Journal of Business & Economic Statistics* **28**, 52–66.

Lam, C. and Yao, Q. (2012). Factor modeling for high-dimensional time series: inference for the number of factors. *The Annals of Statistics* **40**, 694–726.

Lewis, K., Kaufman, J., Gonzalez, M., Wimmer, A. and Christakis, N. A. (2008). Tastes, ties, and time: A new social network dataset using facebook.com. *Social Networks* **30**, 330–342.

Luan, Y. and Li, H. (2003). Clustering of time-course gene expression data using a mixed-effects model with b-splines. *Bioinformatics* **19**, 474–482.

Nowicki, K. and Snijders, T. A. B. (2001). Estimation and prediction for stochastic block structures. *Journal of the American Statistical Association*, **96**, 1077–1087.

Pan, J. and Yao, Q. (2008). Modelling multiple time series via common factors. *Biometrika* **95**, 365–379.

Pollard, D. (1981). Strong consistency of k-means clustering. *The Annals of Statistics* **9**, 135–140.

Von Luxburg, U., Belkin, M. and Bousquet, O. (2008). Consistency of spectral clustering. *The Annals of Statistics* **36**, 555–586.

Wang, Y., Tsay, R. S., Ledolter, J. and Shrestha, K. M. (2013). Forecasting simultaneously high-dimensional time series: A robust model-based clustering approach. *Journal of Forecasting* **32**, 673–684.

Wang, Y. J. and Wong, G. Y. (1987). Stochastic blockmodels for directed graphs. *Journal of the American Statistical Association*, **82**, 8–19.

Wasserman, S. and Faust, K. (1994). *Social Network Analysis: Methods and Applications*. Cambridge university press.

Zhao, Y., Levina, E. and Zhu, J. (2012). Consistency of community detection in networks under degree-corrected stochastic block models. *The Annals of Statistics*, **40**, 2266–2292.

Zhu, X., Pan, R., Li, G., Liu, Y. and Wang, H. (2017). Network vector autoregression. *The Annals of Statistics* **45**, 1096–1123.

School of Data Science, Fudan University, Shanghai, China.

E-mail: xueningzhu@fudan.edu.cn

School of Statistics and Mathematics, Central University of Finance and Economics, Beijing, China.

E-mail: panrui_cufe@126.com

(Received June 2017; accepted September 2018)