Journal of Computational and Graphical Statistics

Bayesian Fused Lasso Regression for Dynamic Binary Networks

Brenda Betancourt, Abel Rodríguez & Naomi Boyd

Taylor & Francis
Taylor & Francis Group

Check for updates

Bayesian Fused Lasso Regression for Dynamic Binary Networks

Brenda Betancourt[a], Abel Rodríguez[a], and Naomi Boyd[b]

[a]Department of Statistics, UC Santa Cruz, Santa Cruz, CA; [b]Department of Finance, West Virginia University, Morgantown, WV

ABSTRACT

We propose a multinomial logistic regression model for link prediction in a time series of directed binary networks. To account for the dynamic nature of the data, we employ a dynamic model for the model parameters that is strongly connected with the fused lasso penalty. In addition to promoting sparseness, this prior allows us to explore the presence of change points in the structure of the network. We introduce fast computational algorithms for estimation and prediction using both optimization and Bayesian approaches. The performance of the model is illustrated using simulated data and data from a financial trading network in the NYMEX natural gas futures market. Supplementary material containing the trading network dataset and code to implement the algorithms is available online.

1. Introduction

Network data, in which observations correspond to the interactions among a group of nodes, have become pervasive in disciplines as diverse as social, physical, and biological sciences. Accordingly, there has been a growing interest in developing tools for the analysis of network data, particularly from a model-based perspective (for excellent reviews see Newman 2003; Goldenberg et al. 2009; Snijders 2011). The focus on this article is on models for time series of binary directed networks that involve the same set of subjects at each time point. In particular, our work is motivated by the study of financial trading networks (FTNs), which capture the pattern of buy and sell transactions in a financial market. A primary goal in the analysis of this type of dynamic network data is link prediction at future times, going as far as predicting the structure of the whole network. An additional goal is to provide a simple model to explore the evolution of the network, and possibly identify change-points in the network dynamic. To accomplish these goals, we extend the idea of p_1 models initially proposed by Holland and Leinhardt (1981) for static binary networks.

Consider a directed binary network among n nodes, $\mathbf{Y} = [y_{i,j}]$, where $y_{i,j} = 1$ if there is a link directed from node i to node j, and $y_{i,j} = 0$ otherwise. Holland and Leinhardt's model assumes conditional independence between pairs of nodes (dyads) and focuses on modeling the pairs $(y_{i,j}, y_{j,i})$ jointly for $i < j$, $j = 1, \ldots, n$ as follows:

$$p\left(y_{i,j}, y_{j,i}\right) \propto \exp\left\{\theta_1 y_{i,j} + \theta_2 y_{j,i} + \theta_3 y_{i,j} y_{j,i}\right\}. \quad (1)$$

This class of models has been extended to a dynamic setting by introducing Markov dependency upon past observations (e.g., see Banks and Carley 1996). In contrast, in the modeling approach discussed in this article, the model parameters are set to be time dependent to add flexibility and account for alterations in the network evolution over time. One challenging feature that is often present in model-based approaches to network data is high-dimensionality. In particular, the number of parameters in our proposed model is larger than the number of available observations. To deal with this issue, we resort to fused lasso regression by imposing an L_1 penalty on the difference between neighboring model parameters (Tibshirani et al. 2005). In a Bayesian setting, this is equivalent to assuming a double exponential prior on the differences of the coefficients in contiguous time points. Here, we explore two different computational approaches for our model. First, full Bayesian inference is presented and implemented using two different sampling schemes. However, the heavy computational load of a full Bayesian analysis is a challenging task as the number of nodes in the network increases. As an alternative, we also carry out maximum a posteriori (MAP) estimation using an optimization approach.

The remainder of the article is organized as follows: Section 2 describes our modeling approach. Section 3 describes the computational algorithms for estimation and prediction from optimization and Bayesian perspectives. Section 4 discusses other related work. Section 5 presents three illustrations, two based on simulated data and a third one that focuses on trading networks from the natural gas futures market in the New York Mercantile Exchange (NYMEX). Finally, a short discussion is presented in Section 6.

2. Modeling Approach

Consider a sequence of binary directed networks $\mathbf{Y}_1, \ldots, \mathbf{Y}_T$, each one observed over a common set of n nodes. The adjacency matrix of the network at time t is therefore an $n \times n$ binary matrix $\mathbf{Y}_t = [y_{i,j,t}]$, where $y_{i,j,t} = 1$ if there is a link directed

from node i to node j at time t, and $y_{i,j,t} = 0$ otherwise. We adopt the convention $y_{i,i,t} \equiv 0$ so that there are no loops within the network. In the illustration we discuss in Section 5.3, the nodes in the network correspond to traders in the NYMEX natural gas futures market, so that $y_{i,j,t} = 1$ if trader i sold to trader j at least once during week t.

We consider an extension of (1) in which the pairs $\{(y_{i,j,t}, y_{j,i,t}) : i < j\}$ are modeled independently using a logistic model of the form

$$p\left(y_{i,j,t}, y_{j,i,t}\right) \propto \exp\left\{\theta_{i,j,t,1} y_{i,j,t} + \theta_{i,j,t,2} y_{j,i,t} + \theta_{i,j,t,3} y_{i,j,t} y_{j,i,t}\right\},$$
(2)

where $\theta_{i,j,t,1}$ and $\theta_{i,j,t,2}$ represent the baseline probabilities of a directed link between nodes i and j, and $\theta_{i,j,t,3}$ controls the level of dependence between $y_{i,j,t}$ and $y_{j,i,t}$. For example, $\theta_{i,j,t,3} = 0$ implies that $y_{i,j,t}$ and $y_{j,i,t}$ are conditionally independent with $\Pr(y_{i,j,t} = 1) = \exp\{\theta_{i,j,t,1}\}/(1 + \exp\{\theta_{i,j,t,1}\})$ and $\Pr(y_{j,i,t} = 1) = \exp\{\theta_{i,j,t,2}\}/(1 + \exp\{\theta_{i,j,t,2}\})$. On the other hand, $\theta_{i,j,t,3} > 0$ favors outcomes in which $y_{i,j,t} = y_{j,i,t}$ (a phenomenon often called positive reciprocity in the network literature), while $\theta_{i,j,t,3} < 0$ favors situations in which $y_{i,j,t} \neq y_{j,i,t}$ (often called negative reciprocity). Hence, by allowing the values of $y_{i,j,t}$ and $y_{j,i,t}$ to be potentially correlated the model can accommodate reciprocity.

The parameters in the multinomial logistic model we just described are time dependent. Hence, it is natural and useful to take into account the information about their temporal correlation structure in the estimation process. In particular, we are interested in a random walk model with double exponential priors of the form

$$\theta_{i,j,t,r} = \theta_{i,j,t-1,r} + \epsilon_{i,j,t,r}, \quad \epsilon_{i,j,t,r} \sim \mathrm{DE}(0, 1/\lambda),$$

where DE represents the double exponential distribution, and $\lambda > 0$ is the parameter that controls the shrinkage level in the differences of the coefficients. A dynamic model of this type of the parameters leads to the joint prior

$$p\left(\boldsymbol{\Theta}_{i,j,r} \mid \lambda\right) \propto \exp\left\{-\lambda \sum_{t=1}^{T} |\theta_{i,j,t,r} - \theta_{i,j,t-1,r}|\right\},$$

where $\boldsymbol{\Theta}_{i,j,r} = (\theta_{i,j,0,r}, \theta_{i,j,1,r}, \ldots, \theta_{i,j,T,r})$ is the vector of parameters for class r and pair of nodes (i, j), and $\theta_{i,j,0,r} = \hat{\theta}_{r,0}$ is assumed known. This pairwise difference prior belongs to the class of Markov random fields and corresponds to a scale mixture of conditionally autoregressive (CAR) priors, which are frequently used in time series, spatial statistics, and image processing (e.g., see Rue and Held 2005). By assuming that $\theta_{i,j,0,r}$ is known, we ensure that the prior distribution, and therefore the associated posterior, is proper. Indeed, note that the more common choice of a flat (improper) prior on $\theta_{i,j,t,r}$ leads in this case to an improper posterior distribution (Sun, Tsutakawa, and Speckman 1999). In addition, assuming double exponential priors is equivalent to imposing L_1 penalty functions on the differences of the parameters in contiguous time points. This penalty type is commonly referred to as the fused lasso with tuning parameter λ. An extensive review of the fused lasso and its theoretical properties is presented in Rinaldo (2009).

We propose to set the hyperparameters $\hat{\theta}_{1,0}$, $\hat{\theta}_{2,0}$, and $\hat{\theta}_{3,0}$ using a procedure reminiscent of empirical Bayes. In particular,

we assume values of $\hat{\theta}_{1,0}$, $\hat{\theta}_{2,0}$, and $\hat{\theta}_{3,0}$ so that the probabilities of the (unobserved) events $(y_{i,j,0}, y_{j,i,0}) = (0, 0)$, $(y_{i,j,0}, y_{j,i,0}) = (1, 0)$, $(y_{i,j,0}, y_{j,i,0}) = (0, 1)$, and $(y_{i,j,0}, y_{j,i,0}) = (1, 1)$ correspond to their time-average probabilities, that is,

$$\hat{\theta}_{1,0} = \frac{\hat{p}_{1,0}}{\hat{p}_{0,0}} \quad \hat{\theta}_{2,0} = \frac{\hat{p}_{0,1}}{\hat{p}_{0,0}} \quad \hat{\theta}_{3,0} = \frac{\hat{p}_{1,1}}{\hat{p}_{0,0}} - \frac{\hat{p}_{1,0}}{\hat{p}_{0,0}} - \frac{\hat{p}_{0,1}}{\hat{p}_{0,0}},$$

where

$$\hat{p}_{0,0} = \frac{2}{n(n-1)} \sum_{t=1}^{T} \sum_{i=1}^{I} \sum_{j=i+1}^{J} I(y_{i,j,t} = 0, y_{j,i,t} = 0),$$

$$\hat{p}_{1,0} = \frac{2}{n(n-1)} \sum_{t=1}^{T} \sum_{i=1}^{I} \sum_{j=i+1}^{J} I(y_{i,j,t} = 1, y_{j,i,t} = 0),$$

$$\hat{p}_{0,1} = \frac{2}{n(n-1)} \sum_{t=1}^{T} \sum_{i=1}^{I} \sum_{j=i+1}^{J} I(y_{i,j,t} = 0, y_{j,i,t} = 1),$$

$$\hat{p}_{1,1} = \frac{2}{n(n-1)} \sum_{t=1}^{T} \sum_{i=1}^{I} \sum_{j=i+1}^{J} I(y_{i,j,t} = 1, y_{j,i,t} = 1),$$

and $I(\cdot)$ represents the indicator function. Other appealing default alternatives are possible, and we use them to study the sensitivity of the model to the prior specification. For example, we could specify $\hat{\theta}_{1,0}$ as the logit of the average probability of an incoming link over the whole history of the network, $\hat{\theta}_{2,0}$ as the logit of the average probability of an outgoing link, and $\hat{\theta}_{3,0} = 0$ to reflect our assumption of no reciprocity a priori. Finally, we also tried setting $\theta_{1,0} = \theta_{2,0} = \theta_{3,0} = 0$, which is consistent with the idea that all categories have the same probability a priori at time 0.

3. Estimation and Prediction

Let $\boldsymbol{\Theta}_{i,j} = \{\boldsymbol{\Theta}_{i,j,2}, \boldsymbol{\Theta}_{i,j,3}, \boldsymbol{\Theta}_{i,j,4}\}$ be the vector of all nonzero parameters for the pair of nodes (i, j). The log-posterior distribution of the parameters is given by

$$\sum_{i<j} \left\{V_{i,j}(\boldsymbol{\Theta}_{i,j}) - \lambda \sum_{r=2}^{4} \|\mathbf{L}\boldsymbol{\Theta}_{i,j,r}\|_1\right\},$$
(3)

where

$$V_{i,j}(\boldsymbol{\Theta}_{i,j}) = \sum_{t=1}^{T} \left\{y_{i,j,t}\theta_{i,j,t,1} + y_{j,i,t}\theta_{i,j,t,2} + y_{i,j,t}y_{j,i,t}\theta_{i,j,t,3}\right.$$
$$- \log\left(1 + \exp\left\{\theta_{i,j,t,1}\right\} + \exp\left\{\theta_{i,j,t,2}\right\}\right.$$
$$\left.\left. + \exp\left\{\theta_{i,j,t,1} + \theta_{i,j,t,2} + \theta_{i,j,t,3}\right\}\right)\right\}$$

is the (unpenalized) log-likelihood, $\|\cdot\|_1$ denotes the L_1-norm, and $\mathbf{L}$ is a pairwise difference matrix of dimension $T \times (T + 1)$ of the form

$$\mathbf{L} = \begin{bmatrix} -1 & 1 & 0 & \cdots & 0 & 0 \\ 0 & -1 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & -1 & 1 \end{bmatrix}.$$

Given λ, (3) can be broken down into $n(n-1)/2$ estimation problems, each one corresponding to fitting a multinomial regression for each pair of nodes in the network.

In the sequel, we focus on algorithms that can be used to solve each of these independent problems, which are then naively implemented in a parallel environment. First, we describe two different sampling algorithms for full Bayesian inference. Estimation results with these sampling schemes are identical but we are interested in comparing their efficiency (see Section 5). We also present a faster optimization alternative for point estimation and prediction that allows implementation of the model in big data settings.

3.1. Full Bayesian Inference

To perform Bayesian inference with a multinomial likelihood, we exploit the data-augmentation method based on Pólya-Gamma latent variables proposed by Polson, Scott, and Windle (2013). Using this approach, the multinomial likelihood can be represented as a mixture of normals with Pólya-Gamma mixing distribution. This approach allows for a full conjugate hierarchical representation of the model and posterior inference through relatively simple Markov chain Monte Carlo (MCMC) algorithms.

For the Bernoulli case, the contribution of the observation $y_t \in \{0, 1\}$ to the likelihood can be written as

$$L(\psi_t) = \frac{\exp(y_t \psi_t)}{1 + \exp(\psi_t)} \propto \exp(\kappa_t \psi_t)$$
$$\int_0^\infty \exp\{-\omega_t \psi_t^2 / 2\} p(\omega_t) d\omega_t,$$

where ψ_t is the log odds of $y_t = 1$, $\kappa_t = y_t - 1/2$, and $p(\omega_t)$ is the Pólya-Gamma density with parameters $(1, 0)$. Hence, by augmenting the model with the latent variable ω_t, conditional Gaussianity for the Bernoulli likelihood can be easily achieved.

Similarly, for the multinomial case, conditional on $\omega_{i,j,t,r}$, the full conditional *likelihood* of each $\theta_{t,r}$ is given by

$$L(\theta_{i,j,t,r} \mid \theta_{i,j,t,-r}) \propto \exp\left\{ -\frac{\omega_{i,j,t,r}}{2}(\theta_{t,r} + C_{i,j,t,r})^2 \right.$$
$$\left. + \kappa_{i,j,t,r}(\theta_{i,j,t,r} + C_{i,j,t,r}) \right\}$$

with

$$C_{t,1} = \log \frac{1 + \exp\{\theta_{t,2} + \theta_{t,3}\}}{1 + \exp\{\theta_{t,2}\}} \quad \kappa_{t,1} = y_{i,j} - 1/2$$

$$C_{t,2} = \log \frac{1 + \exp\{\theta_{t,1} + \theta_{t,3}\}}{1 + \exp\{\theta_{t,1}\}} \quad \kappa_{t,2} = y_{j,i} - 1/2$$

$$C_{t,3} = \log \frac{\exp\{\theta_{t,1} + \theta_{t,1}\}}{\exp\{\theta_{t,1}\} + \exp\{\theta_{t,2}\}} \quad \kappa_{t,3} = y_{i,j} y_{j,i} - 1/2$$

and $\omega_{i,j,t,r} \mid \Theta \sim \text{PG}(1, \theta_{i,j,t,r} + C_{i,j,t,r})$. In the previous expression, PG denotes a Pólya-Gamma distribution. Hence, conditionally on the latent variable $\omega_{i,j,t,r}$ we obtain an augmented Gaussian likelihood with observations $y^*_{i,j,t,r} \sim$ Normal $(\theta_{i,j,t,r}, \omega^{-1}_{i,j,t,r})$, where $y^*_{i,j,t,r} = \kappa_{i,j,t,r}/\omega_{i,j,t,r} - C_{i,j,t,r}$.

Hereinafter, we simplify notation by dropping the subindex i and j associated with the subject pair.

3.1.1. Latent Variables Approach

Using the fact that the double exponential distribution can be expressed as a scale mixture of normals with exponential mixing density (Park and Casella 2008):

$$\frac{a}{2} \exp(-a|x|) = \int_0^\infty \frac{1}{\sqrt{2\pi\tau}} \exp\left(\frac{x^2}{2\tau}\right) \frac{a^2}{2} \exp\left(-\frac{a^2\tau}{2}\right) d\tau,$$

the proposed model can be expressed as a simple hierarchical extension of a dynamic linear model

$$y^*_{t,r} = \theta_{t,r} + \epsilon_{t,r}, \quad \epsilon_{t,r} \sim \text{Normal}\,(0, \omega^{-1}_{t,r}),$$
$$\theta_{t,r} = \theta_{t-1,r} + \varepsilon_{t,r}, \quad \varepsilon_{t,r} \sim \text{Normal}\,(0, \tau^2_{t,r}),$$

for $2 \le t \le T$, where $\theta_{1,r} \sim \text{Normal}\,(\hat{\theta}_{0,r}, \tau^2_{1,r})$ and $\tau^2_{1,r}$ is exponentially distributed a priori with mean $\frac{2}{\lambda^2}$.

We rely on the dynamic linear model representation to update the parameters in a component-wise fashion using a forward filtering backward sampling (FFBS) algorithm (Carter and Kohn 1994; Frühwirth-Schnatter 1994). Furthermore, the latent parameters $\tau_{t,r}$ for $t = 0, \ldots, T - 1$ are independent a posteriori and updated as

$$\left(1/\tau^2_{t,r} | \Theta_r, \lambda\right) \sim \text{IGau}\left(\sqrt{\frac{\lambda^2}{(\theta_{t,r} - \theta_{t-1,r})^2}}, \lambda^2\right),$$

where IGau denotes the Inverse Gaussian distribution (Kyung et al. 2010).

3.1.2. Direct Sampling

Note that the full conditional prior on $\theta_{t,r}$ only involves its two nearest neighbors, so that for $1 \le t \le T - 1$:

$$\pi(\theta_{t,r}|\theta_{t-1,r}, \theta_{t+1,r}) \propto \exp\left\{-\lambda(|\theta_{t,r} - \theta_{t-1,r}| + |\theta_{t+1,r} - \theta_{t,r}|)\right\}.$$

Hence, the full conditional posterior distribution of $\theta_{t,r}$ is a mixture of truncated normal distributions with three components:

$$(\theta_{t,r} \mid y^*_{t,r}, \theta_{t-1,r}, \theta_{t+1,r}, \omega_{t,r}) \sim w_1 \text{TN}(\mu^{(1)}_{t,r}, \sigma_{t,r}; \theta_{t,r} < \xi_{t,r})$$
$$+ w_2 \text{TN}(\mu^{(2)}_{t,r}, \sigma_{t,r}; \theta_{t,r} > \zeta_{t,r})$$
$$+ w_3 \text{TN}(\mu^{(3)}_{t,r}, \sigma_{t,r}; \xi_{t,r} < \theta_{t,r} < \zeta_{t,r}),$$

where $\sigma_{t,r} = 1/\sqrt{\omega_{t,r}}$, $\xi_{t,r} = \min\{\theta_{t-1,r}, \theta_{t+1,r}\}$, $\zeta_{t,r} = \max\{\theta_{t-1,r}, \theta_{t+1,r}\}$, the means of the truncated normal distributions are given by the following expressions:

$$\mu^{(1)}_{t,r} = y^*_{t,r} + \frac{2\lambda}{\omega_{t,r}}, \quad \mu^{(2)}_{t,r} = y^*_{t,r} - \frac{2\lambda}{\omega_{t,r}}, \quad \mu^{(3)}_{t,r} = y^*_{t,r},$$

and the conditional posterior probabilities of the components of the mixture are given by

$$w_1 = \exp\left\{\frac{\omega_{t,r}}{2}\mu^{(1)}_{t,r} - \lambda(\xi_{t,r} + \zeta_{t,r})\right\} \Phi\left(\frac{\xi_{t,r} - \mu^{(1)}_{t,r}}{\sigma_{t,r}}\right)$$

$$w_2 = \exp\left\{\frac{\omega_{t,r}}{2}\mu^{(2)}_{t,r} + \lambda(\xi_{t,r} + \zeta_{t,r})\right\} \Phi\left(\frac{-\zeta_{t,r} - \mu^{(2)}_{t,r}}{\sigma_{t,r}}\right)$$

$$w_3 = \exp\left\{\frac{\omega_{t,r}}{2}\mu_{t,r}^{(3)} - \lambda(\zeta_{t,r} - \xi_{t,r})\right\}$$
$$\left[\Phi\left(\frac{\zeta_{t,r} - \mu_{t,r}^{(3)}}{\sigma_{t,r}}\right) - \Phi\left(\frac{\xi_{t,r} - \mu_{t,r}^{(3)}}{\sigma_{t,r}}\right)\right],$$

where Φ represents the Gaussian cumulative distribution function.

The direct characterization of the posterior distribution for our model is similar to the work of Hans (2009) on Bayesian lasso regression with Gaussian likelihoods. In principle, the efficiency of this algorithm is limited by the use of the full conditional distributions for posterior sampling. However, this approach avoids the introduction of the latent variables $\{\tau_{t,r}\}$ discussed in Section 3.1.1.

3.1.3. Penalty Parameter Estimation
The value of the penalty parameter λ has a direct impact on the quality of the estimates and predictions generated by the model. Hence, under the Bayesian version of our model we assign λ a Gamma hyperprior, $\lambda \sim \text{Gam}(a, b)$. This choice is conditionally conjugate and the full-conditional posterior is simply

$$\lambda \mid \cdots \sim \text{Gam}\left(a + \frac{3(T-1)n(n-1)}{4},\right.$$
$$\left. b + \frac{1}{2}\sum_{t=2}^{T}\sum_{i=1}^{n}\sum_{j=i+1}^{n}\sum_{r=1}^{3}\left|\theta_{i,j,t,r} - \theta_{i,j,t-1,r}\right|\right).$$

Because the parameters are on the logistic scale, we select values of a and b such that, marginally, $\text{var}\{\theta_{i,j,t,r} \mid \theta_{i,j,t-1,r}\}$ is no larger than 1 (so that we do not favor link probabilities that are very close to either 0 or 1). We suggest $a = 1$ and $b = 1/5$, so that the median of $\text{var}\{\theta_{i,j,t,r} \mid \theta_{i,j,t-1,r}, \lambda\}$ is approximately 0.4, and perform a sensitivity analysis to investigate the effect of our choice in the quality of the predictions.

3.1.4. Link Prediction
As we discussed in the introduction, one of the goals of our analysis is short-term link prediction. For either of our sampling algorithms, Monte Carlo posterior samples of the parameters at a future time $T + 1$ can be obtained as

$$\theta_{i,j,T+1,r}^{(b)} \sim \text{DE}\left(\theta_{i,j,T,r}^{(b)}, 1/\lambda^{(b)}\right), \quad b = 1, \ldots, B.$$

Hence, we can estimate (for $i < j$) the probability of a directed link from node i to node j at time $T + 1$ as

$$\hat{p}\left(y_{i,j,T+1} = 1 \mid \mathbf{Y}_T\right)$$
$$= \frac{1}{B}\sum_{b=1}^{B}\left\{p\left[(y_{i,j,T+1}, y_{j,i,T+1}) = (1, 0) \mid \boldsymbol{\theta}_{i,j,T+1}^{(b)}\right]\right.$$
$$\left. + p\left[(y_{i,j,T+1}, y_{j,i,T+1}) = (1, 1) \mid \boldsymbol{\theta}_{i,j,T+1}^{(b)}\right]\right\},$$

with a similar expression being valid for $\hat{p}(y_{j,i,T+1} = 1 \mid \mathbf{Y}_T)$.

3.2. Posterior Mode Estimation
Markov chain Monte Carlo algorithms allow for full posterior inference on our model, but can be too slow to be of practical applicability in large datasets. This issue is particularly pronounced in the case of network data because the number of observations grows as the square of the number of nodes. As an alternative, we develop an optimization algorithm for maximum a posteriori estimation and prediction. Our algorithm is an extension of the split Bregman method proposed by Ye and Xie (2011) to solve general optimization problems with convex loss functions and L^1 penalized parameters (see also Goldstein and Osher 2009). The algorithm is iterative and involves the reformulation of (3) as a constrained problem with the linear restriction $\mathbf{L}\boldsymbol{\Theta} = \mathbf{b}$, and the introduction of a vector of dual variables $\mathbf{v}$ used to split the optimization problem into more tractable steps. Furthermore, we also rely on a second-order Taylor approximation to the multinomial likelihood for the implementation.

The proposed algorithm consists on repeating the following steps until convergence for each vector of parameters $\boldsymbol{\Theta}_r$:

(i) $\boldsymbol{\Theta}_r^{(m+1)} = \arg\max_{\boldsymbol{\Theta}} V(\boldsymbol{\Theta}^{(m)}) - \langle\mathbf{v}_r^{(m)}, \mathbf{L}\boldsymbol{\Theta}_r^{(m)} - \mathbf{b}_r^{(m)}\rangle - \frac{\mu}{2}\|\mathbf{L}\boldsymbol{\Theta}_r^{(m)} - \mathbf{b}_r^{(m)}\|_2^2$

(ii) $\mathbf{b}_r^{(m+1)} = \mathfrak{T}_{\lambda_2\mu^{-1}}(\mathbf{L}\boldsymbol{\Theta}_r^{(m+1)} + \mu^{-1}\mathbf{v}_r^{(m)})$

(iii) $\mathbf{v}_r^{(m+1)} = \mathbf{v}_r^{(m)} + \delta(\mathbf{L}\boldsymbol{\Theta}_r^{(m+1)} - \mathbf{b}_r^{(m+1)})$,

where $\mathbf{v}_r$ is a vector of dual variables, and $\mathfrak{T}_\lambda(\mathbf{w}) = [t_\lambda(w_1), t_\lambda(w_2), \ldots]'$ is a thresholding operator with $t_\lambda(w_i) = \text{sgn}(w_i)\max\{0, |w_i| - \lambda\}$, and $0 < \delta \leq \mu$. We follow previous literature and set $\delta = \mu$ for our implementation noting that convergence of the algorithm is guaranteed for any value of μ (Goldstein and Osher 2009; Ye and Xie 2011).

Efficiency of this algorithm is mainly constrained by the maximization of $\boldsymbol{\Theta}_r$ in the first step. To accelerate it, we replace $V(\boldsymbol{\Theta})$ by its second-order Taylor expansion around the current iterate and proceed to perform component-wise optimization (e.g., see Krishnapuram and Hartemink 2005). Using this substitution, subproblem (i) is differentiable and the estimate of a component $\theta_{t,r}$ of $\boldsymbol{\Theta}_r$ for $1 < t < T - 1$ is updated as

$$\hat{\theta}_{t,r}^{(m+1)} = \left(G_{t,r}^{(m)} - 2\mu\right)^{-1}\left[G_{t,r}^{(m)}\hat{\theta}_{t,r}^{(m)} - g_{t,r}^{(m)} - (v_{t,r}^{(m)} - v_{t-1,r}^{(m)})\right.$$
$$\left. - \mu(\hat{\theta}_{t+1,r}^{(m)} + \hat{\theta}_{t-1,r}^{(m)} + b_{t-1,r}^{(m)} - b_{t,r}^{(m)})\right],$$

where $g_{t,r}^{(m)} = \frac{\partial V}{\partial \theta_{t,r}}\big|_{\boldsymbol{\Theta}^{(m)}}$ and $G_{t,r}^{(m)} = -\frac{\partial^2 V}{\partial \theta_{t,r}^2}\big|_{\boldsymbol{\Theta}^{(m)}}$ are the gradient and the information in the direction of $\theta_{t,r}$ evaluated in the current iterate values. The updates for $t = T$ are obtained in a similar fashion with some minor adjustments.

Note that in the maximum a posteriori estimates obtained in this fashion, the coefficient differences ($\mathbf{b} = \mathbf{L}\boldsymbol{\Theta}$) can be exactly zero. This induces a block partition of the parameters that is suitable for change-point identification (Harchaoui and Levy-Leduc 2008; Rojas and Wahlberg 2014).

3.2.1. Selection of the Penalty Parameter
The penalty λ can be selected through cross-validation by training the model on an observed sample $\mathbf{Y}_1, \ldots, \mathbf{Y}_t$, and performing a one-step-ahead prediction for $\mathbf{Y}_{t+1}$ for a grid of values of λ. This procedure can be repeated to obtain a set of predicted

networks $\hat{\mathbf{Y}}_{t+1}, \ldots, \hat{\mathbf{Y}}_{t+m}$ for $t + m \leq T$, each of these predictions can then be compared against the respective observed networks, the number of false and true positives is computed, and a receiver operating characteristic (ROC) curve is constructed. Finally, the optimal penalty parameter can be chosen as the value of λ in the grid that provides the highest area under the curve (AUC) average over the m predicted networks in the testing dataset.

One potential drawback of this approach is that selection of the optimal tuning parameter through cross-validation can be computationally expensive (Tibshirani et al. 2005). A popular alternative method that can be used with our MAP estimation procedure is to use model selection criteria (e.g., AIC, BIC). Our approach is to select the penalty λ from among a prespecified grid of values by maximizing the Bayesian information criteria (BIC)

$$\text{BIC}_{\lambda} = \sum_{i<j} [2V_{i,j}(\hat{\boldsymbol{\Theta}}_{i,j}) - \mathcal{K}_{i,j}(\lambda) \log(T - 1)],$$

where $\mathcal{K}_{i,j}(\lambda)$ is an estimate of the number of degrees of freedom when the penalty parameter λ is used to compute the MAP estimate. In the case of the fused lasso, Tibshirani and Taylor (2011) showed that the number of nonzero blocks of coefficients in $\hat{\boldsymbol{\Theta}}_{i,j}$ is a rough unbiased estimate of the degrees of freedom.

3.2.2. Link Prediction

Given a point estimate $\hat{\boldsymbol{\theta}}_{i,j,T}$ based on an observed sample $\mathbf{Y}_1, \ldots, \mathbf{Y}_T$, the probability of a directed link from node i to node j at time $T + 1$ is estimated as

$$\hat{p}(y_{i,j,T+1} = 1 \mid \mathbf{Y}_T) = p[(y_{i,j,T+1}, y_{j,i,T+1}) = (1, 0) \mid \hat{\boldsymbol{\theta}}_{i,j,T}]$$
$$+ p\left[(y_{i,j,T+1}, y_{j,i,T+1}) = (1, 1) \mid \hat{\boldsymbol{\theta}}_{i,j,T}\right],$$

with a similar expression being valid for $\hat{p}(y_{j,i,T+1} = 1 \mid \mathbf{Y}_T)$.

4. Related Work

4.1. Computation

The literature on algorithms for parameter estimation for linear regression with a fused lasso penalty is extensive. This is a challenging problem because the fused lasso penalty is not a separable and smooth function, and traditional optimization methods fail under these conditions. In particular, some algorithms that provide a solution path for sequential increments of the regularization parameter have been developed for the fused lasso signal approximator (FLSA) where the design matrix is $\mathbf{X} = \mathbf{I}$ (Friedman et al. 2007; Höfling 2010b), and for a general full rank design matrix $\mathbf{X}$ (Tibshirani and Taylor 2011) only in the case of Gaussian regression.

In this work, we are interested in fused lasso penalized multiclass logistic regression. Friedman, Hastie, and Tibshirani (2010) explored coordinate descent regularization paths for logistic and multinomial logistic regression by using iteratively reweighted least squares (IRLS) but only for lasso, ridge, and elastic net penalties (see also Krishnapuram and Hartemink 2005). Höfling (2010a) proposed a coordinate-wise algorithm

for the fused lasso that can be extended to logistic regression using iterative reweighted least squares (IRWLS), but no path solution algorithms have been fully developed for the multinomial logistic regression setting that is the focus of this article. Recently, Yu et al. (2013) introduced a majorization-minimization (MM) algorithm for fused lasso penalized generalized linear models that benefits from parallel processing. They also presented a good comparison with other existing algorithms including regularization path and first-order methods. For a fixed set of penalization parameters, several optimization algorithms have been proposed for fused lasso problems with general smooth and convex loss functions but not for the specific case of multinomial logistic regression. Liu, Yuan, and Ye (2010) proposed an efficient fused lasso algorithm (EFLA), which solves an FLSA subproblem via a subgradient finding algorithm. Goldstein and Osher (2009) used the split Bregman iteration method to deal with a set of image processing problems that can be treated as general L_1 penalized problems. Motivated by this idea, Ye and Xie (2011) developed the split Bregman-based algorithm for the generalized fused lasso with Gaussian likelihoods. We further extend split Bregman algorithms by introducing a version of the approach for categorical and, in particular, dyadic data likelihoods. In our experience, these kinds of algorithms tend to converge faster and avoid local modes that offer difficulties to most of the other algorithms mentioned above.

From a Bayesian perspective, a general hierarchical model for penalized linear regression that includes the fused lasso penalty is presented in Kyung et al. (2010) for the Gaussian case (see also Park and Casella 2008; Hans 2009). In contrast, the MCMC algorithms discussed in Section 3.1 are designed to deal with categorical data. Furthermore, the latent variable approach from Section 3.1.1 exploits the particular Markovian structure of the problem at hand to generate a much more efficient algorithm than the naive implementation of Kyung et al. (2010) would suggest. On the other hand, and to the best of our knowledge, the direct sampling algorithm of Section 3.1.2, which extend that of Hans (2009) from the regular to the fussed lasso has never been described in the literature before. It is also worth mentioning the work of Scott and Pillow (2012), who used a data augmentation approach for full Bayesian inference of neural spike data counts observed over time by proposing a dynamic negative-binomial factor model with an autoregressive structure. Although both kinds of problems share time-dependent parameters and their algorithm shares some features with our latent variable sample, the structure of our dyad-based likelihood is quite different, and the Pólya-Gamma augmentation scheme required for our network represents a nontrivial extension.

Alternative sparsity inducing priors for time-varying parameters to the fused lasso have been introduced in Chan et al. (2012), who discussed priors for model selection in dynamic contexts, and by Frühwirth-Schnatter and Wagner (2010), Kalli and Griffin (2014), and Belmonte, Koop, and Korobilis (2014), who derived a continuous shrinkage prior that aggressively shrink small coefficients without explicitly zeroing them out. All these techniques were developed in the context of dynamic regression models. Although they could be adapted to identify change points by considering differences between parameter levels, implementing them would come at a significant additional computational cost.

4.2. Models for Dynamic Networks

Sarkar, Chakrabarti, and Jordan (2012) presented a nonparametric link prediction algorithm for sequences of directed binary networks where each observation in time is modeled using a moving window, and the function is estimated through kernel regression. They also incorporated pair specific features, and a spatial dimension using local neighborhoods for each node. Huang and Lin (2009) presented an autoregressive integrated moving average model (ARIMA), and combined it with link occurrence scores based on similarity indices of network topology measures for link prediction in temporal weighted networks (see also da Silva and Bastos 2012). More recently, Bliss et al. (2014) proposed a method based on similarity indices and node attributes joined with a covariance matrix adaptation evolution strategy for link prediction in networks with a large number of nodes.

Other relevant approaches include the dynamic versions of the latent space model of Hoff, Raftery, and Handcock (2002) presented in Sarkar and Moore (2005) and Sewell and Chen (2015), and the work of Xing, Fu, and Song (2010) developing the temporal extension of the mixed membership block-model first introduced in Airoldi et al. (2008) for community identification in social networks. Betancourt, Rodríguez, and Boyd (2015) extended the Bayesian infinite-dimensional model of Kemp et al. (2006) by linking different time periods through a hidden Markov model. On the other hand, Hanneke, Fu, and Xing (2010) introduced a temporal version of the Exponential random graph model (tERGM) first introduced in Frank and Strauss (1986). This temporal model can be used to infer links but its prediction ability is poor unless node attributes or dyadic covariates are included in the model in addition to traditional static network statistics (e.g., reciprocity, transitivity, and popularity statistics). Cranmer and Desmarais (2011) presented a more general temporal ERGM that includes node and dyad-level covariates with applications to political science (see also Snijders, Steglich, and van de Bunt 2010). In this extension, the square root of the indegree and outdegree are added as node attributes at every time point, and functions of past networks can be used as a dyadic covariates.

A key feature of our model is its scalability and efficiency. Because the model structure is relatively simple and dyads are modeled as conditionally independent, estimation and prediction algorithms are fast and can be easily implemented in parallel environments. This means that our model can more easily be scaled to long series of large networks than those discussed above. Conditional independence does have some drawbacks. In particular, although the model directly models reciprocity, it does not explicitly account for transitivity. In spite of this, the illustrations we present in the following sections suggest that our model is at least competitive and, in some cases, superior from a predictive point of view to other state-of-the-art models.

5. Illustrations

The purpose of this section is to evaluate the performance of our model and compare it with the temporal exponential random graph (tERGM) in terms of its link prediction ability. We used the xergm package in R to estimate the tERGM (Leifeld, Cranmer, and Desmarais 2014). More specifically, the tERGM

is estimated with the btergm function, which implements the bootstrapped pseudolikelihood procedure presented in Desmarais and Cranmer (2012). The model we fit includes all the typical ERGM terms, the square root of in- and outdegrees as node covariates, and the lagged network and the delayed reciprocity to model cross-temporal dependencies.

We start by evaluating the performance of the two sampling schemes for Bayesian inference using the effective sample size (ESS) and execution time metrics, and comparing their efficiency with the optimization method for posterior mode estimation using simulated data. We then move on to evaluate the predictive capabilities of the models on both simulated and real data examples consisting of $n = 71$ actors and $T = 201$ observations in time. For this purpose, we carry out out-of-sample cross-validation exercises where we hold out the last 10 weeks in the dataset and make one-step-ahead predictions for the structure of the held-out networks. More specifically, for each $t = 191, 192, \ldots, 200$ we use the information contained in $\mathbf{Y}_1, \ldots, \mathbf{Y}_t$ to estimate the model parameters and obtain predictions $\hat{\mathbf{Y}}_{t+1}$. Using a simple 0/1 utility function, a future link from node i to node j is predicted as $\hat{y}_{i,j,T+1} = \mathbb{I}(\hat{p}(y_{i,j,T+1} = 1 \mid \mathbf{Y}_T) > f)$, for some threshold f that reflects the relative cost associated with false positive and false negative links. Each of these predictions is compared against the observed network $\mathbf{Y}_{t+1}$ to construct a receiver operating characteristic (ROC) curve. For the tERGM, these results are based on 1000 MCMC simulations with other function parameters left as the default values (see btergm documentation for more details).

5.1. MCMC Performance

To asses and compare the performance of the latent variable FFBS and the direct sampling MCMC algorithms, we simulated data from our model. The parameters across all the pairs of nodes were randomly drawn from double exponential distributions as $\theta_{t,r} \sim \mathrm{DE}(\theta_{t,r-1}, 1/\lambda)$ with a true concentration parameter value of $\lambda = 3$. As a measure of efficiency, we use the ESS computed as

$$\mathrm{ESS} = \frac{B}{1 + 2\sum_{k=1}^{K} \rho(k)},$$

where B is the number of post burn-in samples, $\rho(k)$ is the autocorrelation at lag k, and K is the cutoff lag point according to the initial monotone sequence estimator (Geyer 1992).

We computed the effective sample size and the CPU run time in seconds for each pair of nodes based on 20,000 iterations after a burn-in period of 2000 iterations. Table 1 shows the results obtained by averaging over five runs for each sampling scheme, including the relative efficiency of the algorithms standardizing for CPU run time. From these results, it is clear that the latent parameters scheme for the fused lasso is much more efficient than the direct sampler that uses the mixture of truncated normals. Based on these results, in the following

Table 1. Average ESS and CPU times per pair of nodes for MCMC algorithms.

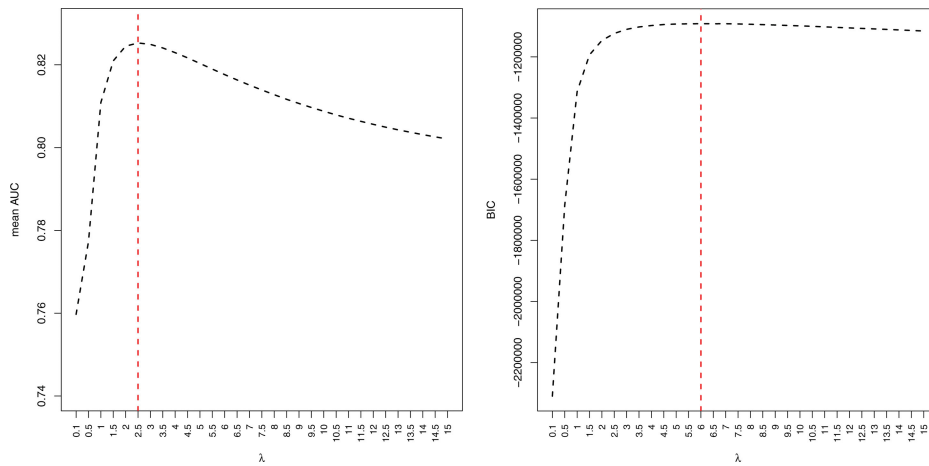| Scheme	ESS	CPU(s)	Rel.ESS
Direct	350	1827.11	0.192
FFBS	3171	818.77	3.878

Figure 1. Simulation 1: Mean AUC over $t = 182, 183, \ldots, 191$ (left panel), and BIC values (right) using the optimization method over a grid of values of λ for simulated dataset. The vertical lines indicate the optimal values of λ.

sections we perform time series cross-validation and prediction for the Bayesian approach using the latent variable FFBS algorithm.

It is also useful to contrast the execution time of the MCMC algorithms with that of the optimization method, which is only 8.03 sec on average for each pair of nodes using a stopping criteria of 10^{-5} for the relative error. Hence, execution times for the MAP algorithm appear to be at least two orders of magnitude smaller than the fastest version of our MCMC algorithms.

5.2. Simulation Studies

We first evaluate our model using two simulations. In the first setting, the parameters across all the pairs of nodes were randomly drawn from double exponential distributions so that $\theta_{t,r} \sim \mathrm{DE}(\theta_{t-1,r}, 1/\lambda)$ with a true penalty parameter value of $\lambda = 12$ using initial values $\theta_{r,0} = 0$. Because the initial value of $\theta_{r,0} = 0$ implies a relatively high initial link probability and the evolution variance $1/\lambda$ is relatively small, the resulting network is relatively dense (average number of links of 2682 at each time point, out of 4970 possible ties). A simple descriptive analysis of the networks shows that they also tend to exhibit low reciprocity and high transitivity.

As discussed in Section 3.2.1, we evaluate two methods to select the penalty parameter λ for the split Bregman optimization algorithm and evaluate the predictive ability of the model. First, we use a setup similar to calibration cross-validation (CCV) by partitioning the data into three sets. The first set is used for modeling and consists of the first 181 observations. Selection of the optimal penalization parameter was performed on the calibration set corresponding to observations $t = 182, \ldots, 191$, by searching the value of λ that maximizes the mean AUC over the predictions of these middle 10 observations. The search for λ was conducted over a grid of 31 values between 0.1 and 15; as shown in the left panel of Figure 1, the optimal value is 2.5. Finally, we report out-of-sample prediction accuracy on the validation set consisting of the last 10 observations, $t = 192, \ldots, 201$. Second, we used the first 191 observations to estimate the model and search the value of λ that optimizes BIC over the same grid of 31 values between 0.1 and 15. The

resulting optimal parameter value in this case is $\lambda = 6$ (see right panel of Figure 1). Again, we evaluate the out-of-sample prediction accuracy of the model on the last 10 observations.

Following Section 3.1.3, for the Bayesian scheme, we used a prior $\lambda \sim \mathrm{Gam}(1, 1/5)$ (mean 5.0, 95% prior symmetric credible interval (0.12, 18), which is similar to the range of values used to select λ under the optimization algorithm). The MCMC algorithm is first used to fit our model to the first 191 observations, and then an out-of-sample prediction for observation 192 is generated. This process is repeated by fitting 192 observations and then predicting observation 193, and so on. The posterior mean of λ is around 9, and varies only very slightly over time. Figure 2 shows the 10 operating characteristic curves associated with the out-of-sample predictions for the last 10 observations using the full Bayesian approach of our model (FFBS algorithm). The right panel of Figure 2 shows the AUC values for the FFBS approach, the tERGM, and MAP predictions generated by using the optimal value of λ obtained from cross-validation (denoted by Bregman-CV) and BIC (denoted by Bregman-BIC). The prediction accuracies for the FFBS algorithm and the Bregman optimization algorithm with cross-validation are almost identical and quite stable over time (both approaches show a good, roughly constant AUC around 83%). On the other hand, Bregman-BIC performs slightly worse than our two other approaches. Furthermore, in this scenario our model outperforms the tERGM, which shows only a fair predictive performance with an average AUC of 72%.

For our second simulation, we generated data with similar characteristics to the trading network dataset. The network is sparse with an average of 784 links over time, consistently shows relatively high reciprocity and includes a structural change around time 85, which can be seen in a shift from low to moderate transitivity (see left panel of Figure 3).

In this case, the search for the optimal penalization parameter for the optimization algorithm was performed by searching the value of λ over a grid of 21 values between 0.1 and 10. Figure 4 shows that the optimal values using the optimization algorithm are $\lambda = 1.5$ for cross-validation, and $\lambda = 3.5$ using BIC. For the Bayesian approach, assuming a hyperprior $\lambda \sim \mathrm{Gam}(1, 1/5)$, the posterior mean for λ over all pairs of nodes is
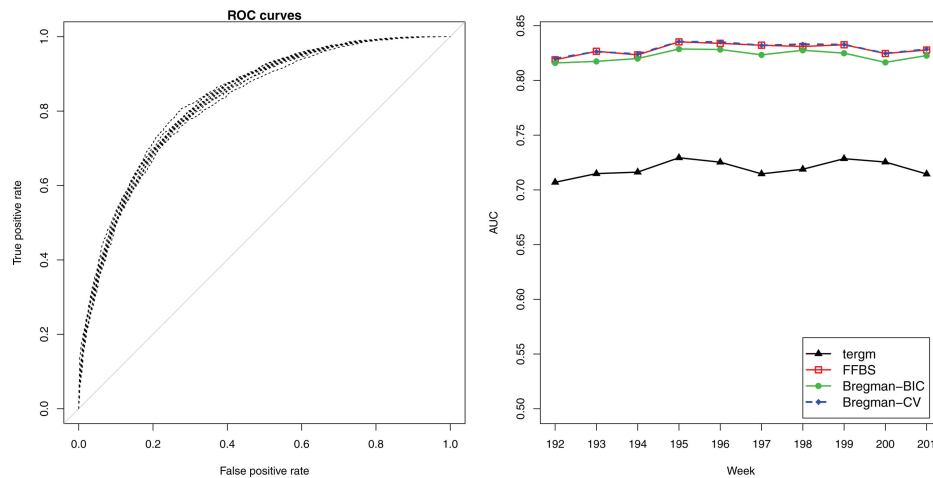
Figure 2. Simulation 1: Plots of the 10 operating characteristic curves associated with one-step-ahead out of sample predictions from the fused lasso model with FFBS algorithm (left panel). Area under the curves (AUC) for the temporal ERGM, and the fused lasso model with FFBS algorithm and Bregman optimization for simulated data. CV (cross-validation) and BIC represent the two methods for tuning parameter selection.

3.7. Figure 5 shows the 10 operating characteristic curves associated with the out-of-sample predictions for the last 10 observations using the full Bayesian approach of our model (FFBS algorithm). The right panel of Figure 5 shows the AUC values for the tERGM and the different algorithms for our model. As before, the prediction accuracies for the FFBS algorithm and the Bregman optimization algorithm with cross-validation are very good (roughly 91% for both approaches), and Bregman-BIC performs just slightly worse. In this scenario, our model again outperforms the tERGM, which shows a good predictive performance with an average AUC of 80%.

As we mentioned in Section 3.2, the maximum a posteriori estimates of the parameters in the fused lasso regression model can be used to identify changes in the network structure over time. In particular, we use an indicator variable that assigns a value of 1 if at least one of the three parameters for pair (i, j) change from time $t − 1$ to time t, and 0 otherwise. The fraction of these indicators over all pairs of nodes provides a rough estimate of the chances that a change-point has occurred on a given week t. The right panel of Figure 3 shows how that proportion changes over time for our second simulation study, which includes a clear change-point around week 85. As expected, the

proportion of dyads showing changes in their parameters peaks on the week the change-point occurs.

5.3. Inference for Financial Trading Networks

In this section, we analyze a sequence of $T = 201$ weekly financial trading networks constructed from *proprietary* trades in the natural gas futures market on the New York Mercantile Exchange (NYMEX) between January 2005 and December 2008. The directed binary networks were constructed by setting $y_{i,j,t} = 1$ if there was at least one transaction in which trader i sold a contract to trader j during week t.

One particularity of this market is that futures were traded on the New York Mercantile Exchange (NYMEX) only through traditional open-outcry trades until September 5, 2006, and as a hybrid market that included electronic trading conducted via the CME Globex platform after that date. Our analysis focuses on 71 traders we identified as being present in the market (although not necessarily active) during the whole period. These trading network is sparse with an average of 826 links each week, and consistently shows very high reciprocity, moderate transitivity, mixing patterns, and community structure (Betancourt,
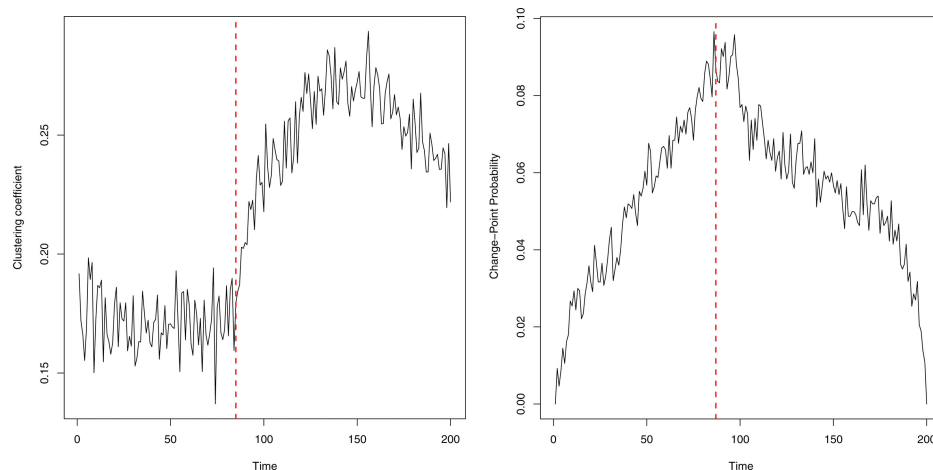

Figure 3. Left panel: Clustering coefficient for the networks in our second simulated dataset. Right panel: Time series of the estimated change-point probability for second simulated dataset. The vertical line represents a structural change at time point 85.
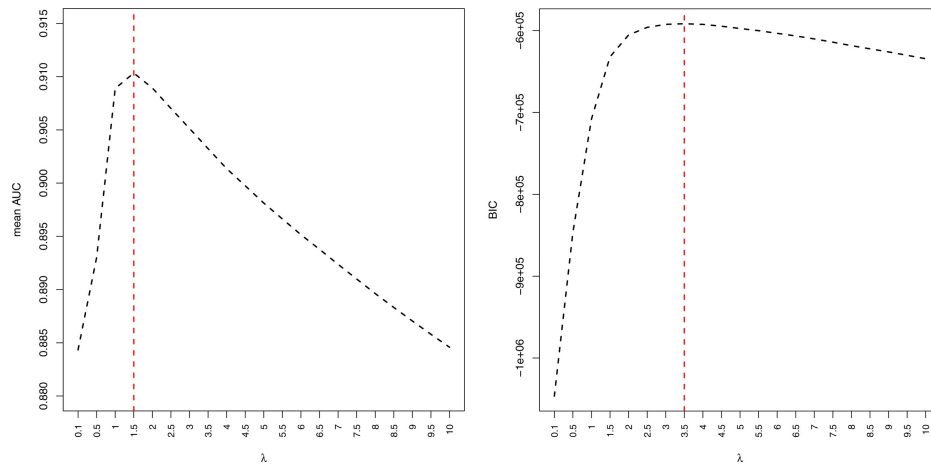
Figure 4. Simulation 2: Mean AUC over $t = 182, 183, \ldots, 191$ (left panel), and BIC values (right) using the optimization method over a grid of values of λ for simulated dataset. The vertical lines indicate the optimal values of λ.
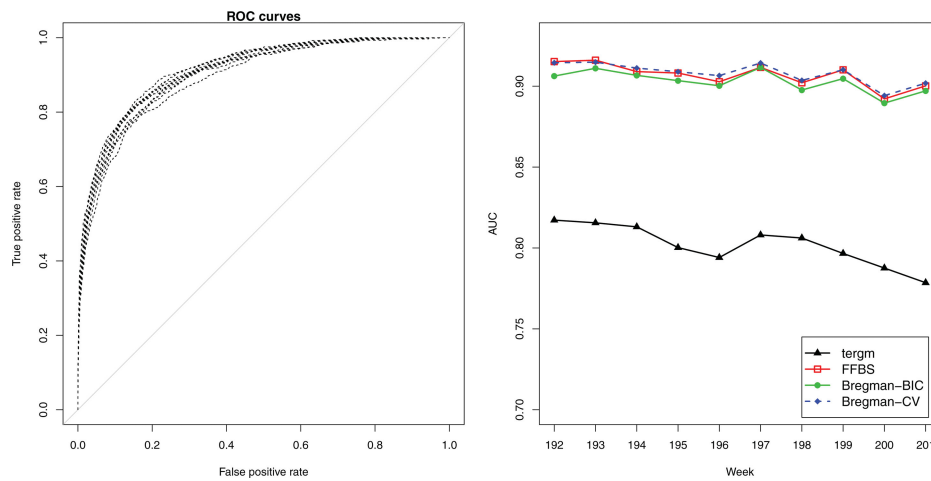

Figure 5. Simulation 2: Plots of the 10 operating characteristic curves associated with one-step-ahead out of sample predictions from the fused lasso model with FFBS algorithm (left panel). Area under the curves (AUC) for the temporal ERGM, and the fused lasso model with FFBS algorithm and Bregman optimization for simulated data. CV (cross-validation) and BIC represent the two methods for tuning parameter selection.

Rodríguez, and Boyd 2015). Analogous to the previous section, selection of the optimal penalization parameter for the optimization algorithm was performed by searching over a grid of 21 values between 0.1 and 10. The value of λ that maximizes the mean AUC over the predictions of 10 weeks $t = 182, \ldots, 191$

(left panel of Figure 6) is $\lambda = 1.5$, while the optimal value for BIC over the first 191 weeks is $\lambda = 3$. Similarly, for the MCMC algorithm we employ the same Gam$(1, 1/5)$ prior we used in our simulations. The left panel in Figure 7 shows the operating characteristic curves associated with the out-of-sample predictions
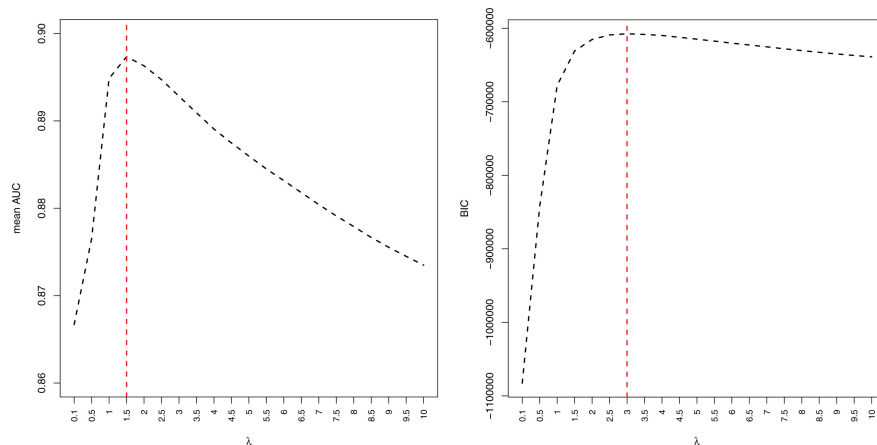

Figure 6. Mean AUC over $t = 182, 183, \ldots, 191$ (left panel), and BIC values (right) using the optimization method over a grid of values of λ for trading network. The vertical lines indicate the optimal values of λ.
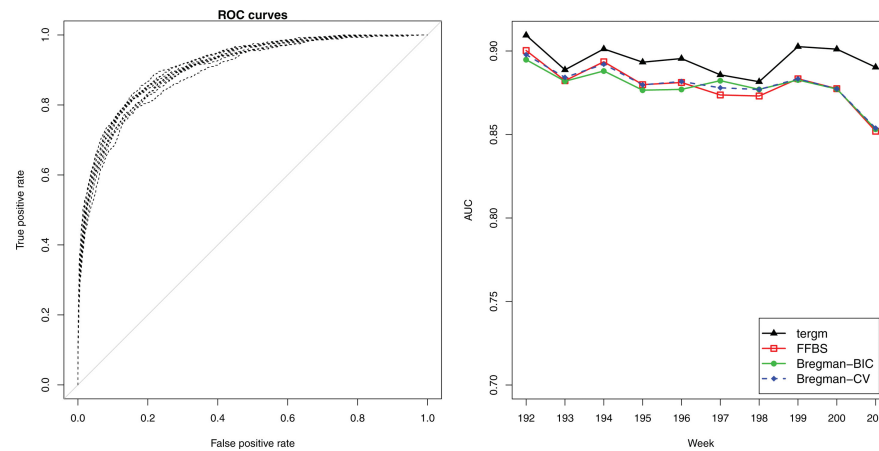
Figure 7. Plots of the 10 operating characteristic curves associated with one-step-ahead out of sample predictions from the fused lasso model with FFBS algorithm (left panel). Area under the curves (AUC) for the temporal ERGM, and the fused lasso model with FFBS algorithm and Bregman optimization for the trading network. CV (cross-validation) and BIC represent the two methods for tuning parameter selection.

generated by our model fitted using the full Bayesian approach. Note that all the curves are very similar, showing that the performance of our model is quite stable over time. In the same spirit, the right panel of Figure 7 shows weekly AUCs for the FFBS algorithm, the tERGM, Bregman-CV, and Bregman-BIC. The results show that our model performs quite well, with AUC values between 86% and 90% on every week. However, in this particular case the tERGM performs slightly but consistently better, with AUC values around 2% higher. Furthermore, as in the simulations, the performance of the FFBS and the Bregman-CV algorithm is very similar over all 10 weeks, and the Bregman-BIC performs slightly worse particularly during the first 6 weeks.

As discussed in our simulation study, the fraction of dyads for which at least one of the three parameters presents a change point at time t provides a rough estimate of the chances that a change-point has occurred on a given week. Figure 8 presents the time series of the fraction of pairs that show at least one parameter change each week under the Bregman-CV algorithm
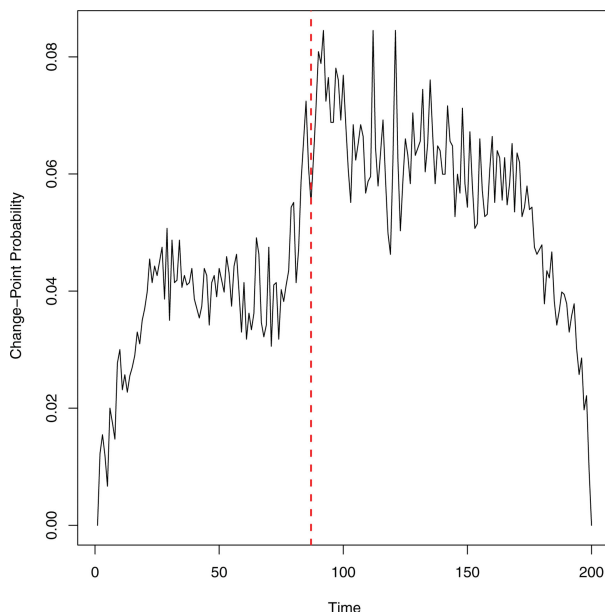
with the optimal cross-validated $\lambda = 1.5$. The vertical line corresponds to the date of introduction of electronic trading. Note that the maximum of the time series over the 201 weeks appears right after the introduction of the electronic market and that a second, less marked peak appear around week 124. These results are consistent with previous analyses of these data (Betancourt, Rodríguez, and Boyd 2015).

6. Discussion

We have discussed a flexible and powerful model for prediction on dynamic networks. Indeed, the model we present shows competitive performance in the trading network dataset and superior performance in the simulation studies while being much more computationally efficient than alternatives available in the literature. Furthermore, the model can be easily extended to weighted networks by replacing the multinomial likelihood with an appropriate member of the exponential family. Similarly, a variation of the model can be devised for undirected networks.

Interestingly, the results on both the simulated and the trading network data showed that the prediction ability of the optimization approach is very similar to that of the Bayesian method, while being far more computationally efficient. In addition, the optimization approach also provides a way of exploring the presence of change points in the network dynamics. On the other hand, the cross-validation approach for tuning parameter selection provides slightly better results than the BIC method, but the computational cost is considerably higher.

Supplementary Materials

C++ Code: Code to implement the algorithms for estimation and prediction described in this article. Please refer to the README file contained in the zip file for more details. (.cpp files)

Trading network dataset: Dataset used in the illustration in Section 5.3. (.txt file)

Figure 8. Time series of the estimated change-point probability for the trading network. The vertical line represents the introduction of electronic trading in the market at week 85.

Funding

References

Airoldi, E., Blei, D. M., Fienberg, S. E., and Xing, E. P. (2008), "Mixed Membership Stochastic Blockmodels," *Journal of Machine Learning Research*, 9, 1981–2014. [845]

Banks, D., and Carley, K. M. (1996), "Models for Network Evolution," *Journal of Mathematical Sociology*, 21, 173–196. [840]

Belmonte, M. A., Koop, G., and Korobilis, D. (2014), "Hierarchical Shrinkage in Time-Varying Parameter Models," *Journal of Forecasting*, 33, 80–94. [844]

Betancourt, B., Rodríguez, A., and Boyd, N. (2015), "Modelling and Prediction of Financial Trading Networks: An Application to the NYMEX Natural Gas Futures Market," Technical Report UCSC-SOE-13-07, University of California - Santa Cruz. [845,848,849]

Bliss, C., Frank, M., Danforth, C., and Dodds, P. S. (2014), "An Evolutionary Algorithm Approach to Link Prediction in Dynamic Social Networks," *Journal of Computer Science and Technology*, 2, 750–764. [845]

Carter, C., and Kohn, R. (1994), "On Gibbs Sampling for State Space Models," *Biometrika*, 81, 541–553. [842]

Chan, J. C., Koop, G., Leon-Gonzalez, R., and Strachan, R. W. (2012), "Time Varying Dimension Models," *Journal of Business & Economic Statistics*, 30, 358–367. [844]

Cranmer, S., and Desmarais, B. (2011), "Inferential Network Analysis with Exponential Random Graph Models," *Political Analysis*, 19, 66–86. [845]

da Silva, P., and Bastos, R. (2012), "Time Series Based Link Prediction," in *IEEE World Congress on Computational Intelligence*, Brisbane, Australia, pp. 784–790. [845]

Desmarais, B., and Cranmer, S. (2012), "Statistical Mechanics of Networks: Estimation and Uncertainty," *Physica A: Statistical Mechanics and its Applications*, 391, 1865–1876. [845]

Frank, O., and Strauss, D. (1986), "Markov Graphs," *Journal of the American Statistical Association*, 81, 832–842. [845]

Friedman, J., Hastie, T., Hoefling, H., and Tibshirani, R. (2007), "Pathwise Coordinate Optimization," *Annals of Applied Statistics*, 2, 302–332. [844]

Friedman, J., Hastie, T., and Tibshirani, R. (2010), "Regularization Paths for Generalized Linear Models via Coordinate Descent," *Journal of Statistical Software*, 33, 1–22. [844]

Frühwirth-Schnatter, S. (1994), "Data Augmentation and Dynamic Linear Models," *Journal of Time Series Analysis*, 15, 183–202. [842]

Frühwirth-Schnatter, S., and Wagner, H. (2010), "Stochastic Model Specification Search for Gaussian and Partial Non-Gaussian State Space Models," *Journal of Econometrics*, 154, 85–100. [844]

Geyer, C. (1992), "Practical Markov Chain Monte Carlo," *Statistical Science*, 7, 473–483. [845]

Goldenberg, A., Zheng, A. X., Fienberg, S. E., and Airoldi, E. M. (2009), "A Survey of Statistical Network Models," *Foundations and Trends in Machine Learning*, 2, 129–233. [840]

Goldstein, T., and Osher, S. (2009), "The Split Bregman Method for L1-Regularized Problems," *SIAM Journal on Imaging Sciences*, 2, 323–343. [843,844]

Hanneke, S., Fu, W., and Xing, E. P. (2010), "Discrete Temporal Models of Social Networks," *Electronical Journal of Statistics*, 4, 585–605. [845]

Hans, C. (2009), "Bayesian Lasso Regression," *Biometrika*, 96, 835–845. [843,844]

Harchaoui, Z., and Levy-Leduc, C. (2008), "Catching Change-Points with Lasso." [843]

Hoff, P., Raftery, A., and Handcock, M. (2002), "Latent Space Approaches to Social Network Analysis," *Journal of the American Statistical Association*, 97, 1090–1098. [845]

Höfling, H. (2010a), "A Coordinate-Wise Optimization Algorithm for the Fused Lasso," arXiv:1011.6409 [stat.CO]. [844]

—— (2010b), "A Path Algorithm for the Fused Lasso Signal Approximator," *Journal of Computational and Graphical Statistics*, 19, 984–1006. [844]

Holland, P., and Leinhardt, K. (1981), "An Exponential Family of Probability Distributions for Directed Graphs," *Journal of the American Statistical Association*, 76, 33–65. [840]

Huang, Z., and Lin, D. (2009), "The Time-Series Link Prediction Problem with Applications in Communication Surveillance," *INFORMS Journal on Computing*, 21, 286–303. [845]

Kalli, M., and Griffin, J. E. (2014), "Time-Varying Sparsity in Dynamic Regression Models," *Journal of Econometrics*, 178, 779–793. [844]

Kemp, C., Tenenbaum, J. B., Griffiths, T., Yamada, T., and Ueda, N. (2006), "Learning Systems of Concepts with an Infinite Relational Data," in *Proceedings of the 22nd Annual Conference on Artificial Intelligence (UAI 2006)*, Cambridge, pp. 381–388. [845]

Krishnapuram, B., and Hartemink, A. (2005), "Sparse Multinomial Logistic Regression: Fast Algorithms and Generalization Bounds," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27, 957–968. [843,844]

Kyung, M., Gill, J., Ghosh, M., and Casella, G. (2010), "Penalized Regression, Standard Errors, and Bayesian Lassos," *Bayesian Analysis*, 5, 369–412. [842,844]

Leifeld, P., Cranmer, S., and Desmarais, B. (2014), *xergm: Extensions of Exponential Random Graph Models, R Package*. Available at http://CRAN.R-project.org/package=xergm. [845]

Liu, J., Yuan, L., and Ye, J. (2010), "An Efficient Algorithm for a Class of Fused Lasso Problems," in *The ACM SIG Knowledge Discovery and Data Mining*, Washington, DC, pp. 323–332. [844]

Newman, M. E. J. (2003), "The Structure and Function of Complex Networks," *SIAM Review*, 45, 167–256. [840]

Park, T., and Casella, G. (2008), "The Bayesian Lasso," *Journal of the American Statistical Association*, 103, 681–686. [842,844]

Polson, N., Scott, J., and Windle, J. (2013), "Bayesian Inference for Logistic Models using Polya-Gamma Latent Variables," arXiv:1205.0310v3; *Journal of the American Statistical Association*, 108, 1339–1349. [842]

Rinaldo, A. (2009), "Properties and Refinements of the Fused Lasso," *Annals of Statistics*, 37, 2922–2952. [841]

Rojas, C., and Wahlberg, B. (2014), "On Change Point Detection using the Fused Lasso Method," arXiv:1401.5408 [math.ST]. [843]

Rue, H., and Held, L. (2005), *Gaussian Markov Random Fields: Theory and Applications*, Boca Raton, FL: CRC Press. [841]

Sarkar, P., Chakrabarti, D., and Jordan, M. (2012), "Nonparametric Link Prediction in Dynamic Networks," in *Proceedings of the 29th International Conference in Machine Learning*, Edinburgh, Scotland, UK, pp. 1–8. [845]

Sarkar, P., and Moore, A. (2005), "Dynamic Social Network Analysis using Latent Space Models," *SIGKDD Explorations: Special Edition on Link Mining*, 7, 31–40. [845]

Scott, J., and Pillow, J. W. (2012), "Fully Bayesian Inference for Neural Models with Negative-Binomial Spiking," in *Advances in Neural Information Processing Systems* (Vol. 25), pp. 1898–1906. [844]

Sewell, D. K., and Chen, Y. (2015), "Analysis of the Formation of the Structure of Social Networks by using Latent Space Models for Ranked Dynamic Networks," *Journal of the Royal Statistical Society*, Series C, 611–633. [845]

Snijders, T. A. B. (2011), "Statistical Models for Social Networks," *Annual Review of Sociology*, 37, 129–151. [840]

Snijders, T. A. B., Steglich, C., and van de Bunt, G. (2010), "Introduction to Actor-Based Models for Network Dynamics," *Social Networks*, 32, 44–60. [845]

Sun, D., Tsutakawa, R. K., and Speckman, P. L. (1999), "Posterior Distribution of Hierarchical Models using CAR (1) Distributions," *Biometrika*, 86, 341–350. [841]

Tibshirani, R., Saunders, M., Rosset, S., Zhu, J., and Knight, K. (2005), "Sparsity and Smoothness via the Fused Lasso," *Journal of the Royal Statistical Society*, Series B, 67, 91–108. [840,844]

Tibshirani, R., and Taylor, J. (2011), "The Solution Path of the Generalized Lasso," *The Annals of Statistics*, 39, 1335–1371. [844]

Xing, E. P., Fu, W., and Song, L. (2010), "A State-Space Mixed Membership Blockmodel for Dynamic Network Tomography," *Annals of Applied Statistics*, 4, 535–566. [845]

Ye, G., and Xie, X. (2011), "Split Bregman Method for Large Scale Fused Lasso," *Computational Statistics and Data Analysis*, 55, 1552–1569. [843,844]

Yu, D., Won, J., Lee, T., Lim, J., and Yoon, S. (2013), "High-Dimensional Fused Lasso Regression using Majorization-Minimization and Parallel Processing," arXiv:1306.1970v2 [stat.ME] *Journal of Computational and Graphical Statistics*, 24, 121–153. [844]