Contrasting Multiple Social Network Autocorrelations for Binary Outcomes, With Applications To Technology Adoption

BIN ZHANG, Temple University
ANDREW C. THOMAS, Carnegie Mellon University
PATRICK DOREIAN, University of Pittsburgh and University of Ljubljana
DAVID KRACKHARDT and RAMAYYA KRISHNAN, Carnegie Mellon University

The rise of socially targeted marketing suggests that decisions made by consumers can be predicted not only from their personal tastes and characteristics, but also from the decisions of people who are close to them in their networks. One obstacle to consider is that there may be several different measures for closeness that are appropriate, either through different types of friendships, or different functions of distance on one kind of friendship, where only a subset of these networks may actually be relevant. Another is that these decisions are often binary and more difficult to model with conventional approaches, both conceptually and computationally. To address these issues, we present a hierarchical auto-probit model for individual binary outcomes that uses and extends the machinery of the auto-probit method for binary data. We demonstrate the behavior of the parameters estimated by the multiple network-regime auto-probit model (m-NAP) under various sensitivity conditions, such as the impact of the prior distribution and the nature of the structure of the network. We also demonstrate several examples of correlated binary data outcomes in networks of interest to information systems, including the adoption of caller ring-back tones, whose use is governed by direct connection but explained by additional network topologies.

Categories and Subject Descriptors: H.1.0 [**Models and Principles**]: General

General Terms: Algorithms, Management, Experimentation, Theory

Additional Key Words and Phrases: Social network, autocorrelation model, diffusion, Bayesian method

ACM Reference Format:
Zhang, B., Thomas, A. C., Doreian, P., Krackhardt, D., and Krishnan, R. 2013. Contrasting multiple social network autocorrelations for binary outcomes, with applications to technology adoption. *ACM Trans. Manage. Inf. Syst.* 3, 4, Article 18 (January 2013), 21 pages.
DOI:http://dx.doi.org/10.1145/2407740.2407742

1. INTRODUCTION

The prevalence and widespread adoption of online social networks has made the analysis of these networks, particularly the behavior of individuals embedded within them, an important topic of study in information systems [Agarwal et al. 2008; Oinas-Kukkonen et al. 2010], which builds on previous work in the context of technology diffusion [Brancheau and Wetherbe 1990; Chatterjee and Eliashberg 1990; Premkumar et al. 1994]. While past investigations into behavior in networks were typically limited to hundreds of people, contemporary data collection and retrieval technologies enable easy access to network data on a much larger scale. Analyzing the behavior of these individuals, such as their purchasing or technology adoption tendencies, requires

Author's address: B. Zhang; email: bzhang@temple.edu.

DOI:http://dx.doi.org/10.1145/2407740.2407742

statistical techniques that can handle both the scope and the complexity of the data.

The social network aspect is one such complexity. Researchers once assumed that an individual's decision to purchase a product or adopt a technology is solely associated with their personal attributes, such as age, education, and income [Allenby and Rossi 1998; Kamakura and Russell 1989], though this could be due both to a lack of social network data and a mechanism for handling it; indeed, recent developments have shown that their decisions are associated with the decisions of an individual's neighbors in their social networks [Bernheim 1994; Manski 2000; Smith and LeSage 2004]. This could be due to a "contagious" effect, where someone imitates the behavior of their friends, or an indication of latent homophily, in which some unobserved and shared trait drives both the tendency for two people to form a friendship and for each to adopt [Aral et al. 2009; Shalizi and Thomas 2011]; either social property will increase the ability to predict a person's adoption behavior beyond their personal characteristics.

Each of these produces outcomes that are correlated between members of the network who are connected. A popular approach to study this phenomenon is to use a model with explicit autocorrelation between individual outcomes, defined with a single network structure term. With the depth of data now available, an actor is very often observed to be a member of multiple distinct but overlapping networks, such as a friend network, a work colleague network, a family network, and so forth, and each of these networks may have some connection to the outcome of interest, so a model that condenses all networks into one relation will be insufficient. While models have been developed to include two or more network autocorrelation terms, such as Doreian [1989], these do not allow for the immediate and principled inclusion of binary outcomes, e.g. whether or not adopt a new technology; other methods to deal with binary outcomes on multiple networks, such as Yang and Allenby [2003], instead take a weighted average of other networks in the system, combining them into one, which has the side effect of constraining the sign of each network autocorrelation component to be identical, which may be undesirable if there are multiple effects thought to be in opposition to one another.

To deal with these issues, we construct a hierarchical autocorrelation model for binary outcomes that uses the probit framework, allowing us to represent these outcomes as if they are dichotomized outcomes from a multivariate Gaussian random variable; this is then presented as in Doreian [1989] to have multiple regimes of network autocorrelation. We consider two approaches for estimating the parameters of this model. We first use the Expectation-Maximization algorithm (EM) to find a maximum likelihood estimator for the model parameters, then use Markov Chain Monte Carlo (MCMC), a method from Bayesian statistics, to develop an alternate estimate based on the posterior mean. We also study the sensitivity of both solutions to the change of parameters' prior distributions. Preliminary experiments show that the EM solution to this model is degenerate, and cannot produce a usable variance-covariance matrix for parameter estimates, and so the MCMC method is preferred. We construct these algorithms in the R programming language, and validate the MCMC method with the posterior quantiles method of Cook et al. [2006]. We ensure that the parameter estimates from the model are correct by first testing on simulated data, before moving on to real examples of network-correlated behavior.

The rest of the article is organized as follows. We discuss the literature on the network autocorrelation model in Section 2. Our two estimation solutions for the multi-network auto-probit, based on EM and MCMC, are presented in Section 3. In Section 4, we present the results of experiments for software validation and parameter estimation behavior observation. Conclusions and suggestions for future work complete the article in Section 5.

2. BACKGROUND

Network models of behavior are developed to study the process of social influence on the diffusion of a behavior, which is the process "by which an innovation is communicated through certain channels over time among the members of a social system ... a special type of communication concerned with the spread of messages that are perceived as new ideas" [Rogers 1962]. These models have been widely used to study diffusion since the Bass [1969] model, a population-level approach that assumes that everyone in the social network has the same probability of interacting. Such assumption is not realistic because given a large social network, the probability of any random two nodes connecting to each other is not the same; for example, people with closer physical distance communicate more and are likely to exert greater influence on each other. A refinement to this approach is a model such as the simultaneous autoregressive model (SAR), where the outcomes of neighboring individuals are explicitly linked. The general method of SAR is described in Anselin [1988] and Cressie [1993]; it considers simultaneous autoregression on the residuals of the form

$$\mathbf{y} = \mathbf{X}\beta + \boldsymbol{\theta},\ \boldsymbol{\theta} = \rho\mathbf{W}\boldsymbol{\theta} + \boldsymbol{\epsilon},$$

where $\mathbf{y}$ is a vector of observed outcomes, in this case consumer choice, and $\mathbf{X}$ is a vector of explanatory variables. Rather than an independent error term, $\boldsymbol{\theta}$ represents error terms whose correlation is specified by $\mathbf{W}$, the social network matrix of interest, and ρ, the corresponding network autocorrelation, distributing a Gaussian error term ϵ_i. Some well accepted maximum likelihood estimate solutions are provided by Ord [1975], Doreian [1980, 1982], and Smirnov [2005].

Standard network autocorrelation models such as those of Burt [1987] and Leenders [1997], can only accommodate one network. However, an actor is very often under influence of multiple networks, such as that of friends and that of colleagues. So if research requires investigation of which autocorrelation term out of multiple networks plays the most significant role in consumers' decisions, none of these models are adequate, and a model that can accommodate two or more networks is necessary.

Cohesion and structural equivalence are two competing social network models to explain diffusion of innovation. In the cohesion model, a focal person's adoption is influenced by his/her neighbors in the network. In the structural equivalence model, a focal person's adoption is influenced by the people who have the same position in the social network, such as sharing many common neighbors. While considerable work has been done on these models on real data, the question of which network model best explains diffusion has not been resolved. To approach this, Doreian [1989] introduced a model for "two regimes of network effects autocorrelation"[1] for continuous outcomes. The model is specified as

$$\mathbf{y} = \mathbf{X}\beta + \rho_1\mathbf{W}_1\mathbf{y} + \rho_2\mathbf{W}_2\mathbf{y} + \boldsymbol{\epsilon},$$

where $\mathbf{y}$ is the dependent variable; $\mathbf{X}$ is a vector of explanatory variables; and each $\mathbf{W}$ represents a social structure underlying each autoregressive regime. This model takes both interdependence of actors and their attributes, such as demographics, into consideration; these interdependencies are each described by a weight matrix $\mathbf{W}_i$. Doreian's

[1]The term "network effects" can refer to two directly related concepts: the autocorrelation between individual behaviors on a network, and the increased impact of a technology to an individual when used by more people within a network. Our meaning is the first, though we use the term *partial network autocorrelation* to avoid ambiguity.

model can capture both actor's intrinsic opinion and influence from alters in his social network.

As this model takes a continuous dependent variable, Fujimoto and Valente [2011] present a plausible solution for binary outcomes by directly inserting an autocorrelation term **Wy** into the right hand side of a logistic regression

$$y_i \sim \mathrm{Be}(p_i)$$
$$\log\left(\frac{p_i}{1-p_i}\right) = \mathbf{X}\beta + \rho \sum_j \mathbf{W}_{ij}\mathbf{y}_j.$$

Due to its speed of implementation, this method is called "quick and dirty" (QAD) by Doreian [1982]. Although it may support a binary dependent variable and multiple network terms, this model does not satisfy the assumption of logistic regression—the observations are not conditionally independent, and the estimation results are biased. Thomas [2012] shows that this method has more consequences than expected for the estimation procedure beyond simple bias; for example, in cases where **W** is a directed graph, networks that are directional cannot be distinguished from their reversed counterparts.

Yang and Allenby [2003] propose a hierarchical Bayesian autoregressive mixture model to analyze the effect of multiple network autocorrelation terms on a binary outcome. Their model can only technically accommodate one network effect, composed of several smaller networks that are weighted and added together. This model therefore assumes that all component network coefficients must have the same sign,[2] and also be statistically significant or insignificant together. Such assumptions do not hold if the effects of any, but not all of the component networks are statistically insignificant, or of the opposite sign to the other networks; so a method that estimates coefficients for each **W** separately is necessary for our applications. We contrast our method with the Yang-Allenby grand **W** construction method, a finite mixture of coefficient matrices, in Appendix A.1.

3. METHOD

We propose a new auto-probit model, which accommodates multiple regimes of network autocorrelation terms for the same group of actors, which we call the multiple network auto-probit model (m-NAP). We then provide two methods to obtain estimates for our model. The first is the use of Expectation-Maximization, which employs a maximum likelihood approach, and the second is a Markov Chain Monte Carlo routine that treats the model as Bayesian. Detailed descriptions of both estimations are shown in Appendix A.1 and A.2.

3.1. Model Specification

The actors are assumed to have k different types of network connections between them, where $\mathbf{W}_i$ is the ith network in question $i \in \{1, ..., k\}$. $\mathbf{y}$ is the vector of length n of observed binary choices, and is an indicator function of the latent preference of

[2]It is of course possible to specify terms in the **W** matrix as negative, to represent anticorrelation on a tie, but this must be done *a priori*, and is redundant in our approach.

consumers **z**. If **z** is larger than a threshold 0, consumers choose **y** as 1; if **z** is smaller than 0, then consumers would choose **y** as 0.

$$\mathbf{y} = \mathbb{I}(\mathbf{z} > 0)$$
$$\mathbf{z} = \mathbf{X}\beta + \boldsymbol{\theta} + \boldsymbol{\epsilon},\ \boldsymbol{\epsilon} \sim \text{Normal}_n(0, I_n)$$
$$\boldsymbol{\theta} = \sum_{i=1}^{k} \rho_i \mathbf{W}_i \boldsymbol{\theta} + \mathbf{u},\ \mathbf{u} \sim \text{Normal}_n\left(0, \sigma^2 I_n\right),$$

where **z** is a function of both exogenous covariates **X**, autocorrelation term $\boldsymbol{\theta}$, and individual error. **X** is an $n \times m$ covariate matrix that includes a constant as its first column; these covariates could be the exogenous characteristics of consumers. β is an $m \times 1$ coefficient vector associated with **X**. $\boldsymbol{\theta}$ is the autocorrelation term, which is responsible for those nonzero covariances in the **z**. $\boldsymbol{\theta}$ can be described as the aggregation of multiple network structure $\mathbf{W}_i$ and coefficient ρ_i. Each $\mathbf{W}_i$ is a network structure describing connections and relationships among consumers.

Our model explicitly allows multiple competing networks that can be defined by different mechanisms on an existing basis of network ties; for example, $\mathbf{W}_1$ describes an effect acting directly on a declared tie, such as homophily or social influence, whereas $\mathbf{W}_2$ describes the structural equivalence due to those ties. It can also be that each $\mathbf{W}_i$ is defined by a different type of network edge, such as friendship, colleagueship, or mutual group membership—note that none of these relationships must be mutually exclusive. Each coefficient ρ_i describes the effect size of its corresponding network $\mathbf{W}_i$, so that we can compare the relative scales of competing network structures for the same group of actors embedded in social networks.

We model the error term as an augmented expression that consists of two parts, $\boldsymbol{\epsilon}$ and **u**. $\boldsymbol{\epsilon}$ is the unobservable error term of **z** that describes individual-level variation not shared on the network, and **u** is the error that is then distributed along each network, accounting for the nonzero covariance between units. If we marginalize this model by integrating out $\boldsymbol{\theta}$, all the unobserved interdependency will be isolated in a single expression for the distribution of **z**, given parameters β, ρ, and σ^2, as multivariate with mean $\mathbf{X}\beta$ and variance **Q**.

$$\mathbf{z} \sim \text{Normal}\left(\mathbf{X}\beta, \mathbf{Q}\right),$$

where

$$\mathbf{Q} = I_n + \sigma^2 \left(I_n - \sum_{i=1}^{k} \rho_i \mathbf{W}_i\right)^{-1} \left(\left(I_n - \sum_{i=1}^{k} \rho_i \mathbf{W}_i\right)^{-1}\right)^{\top}.$$

The nonstandard form of the covariance matrix can therefore pose a significant computational issue.

3.2. Expectation-Maximization Solution

We first develop an approach by maximizing the likelihood of the model obtain the maximum likelihood estimate for the model using EM. Since **z** is latent, we treat it as unobservable data, for which the EM algorithm is one of the most used methods. Detailed description of our solution for k regimes of network autocorrelation is in Appendix A.1.

The method consists of two steps: first, we estimate the expected value of functions of the unobserved **z** given the current parameter set $\boldsymbol{\phi}$, ($\boldsymbol{\phi} = \{\boldsymbol{\beta}, \boldsymbol{\rho}, \sigma^2\}$). Second, we use these estimates to form a complete data set $\{\mathbf{y}, \mathbf{X}, \mathbf{z}\}$, with which we estimate a new $\boldsymbol{\phi}$ by maximizing the expectation of the likelihood of the complete data.

We first initialize the parameters to be estimated,

$$\beta_i \sim \text{Normal}(\nu_\beta, \Omega_\beta);$$
$$\rho_j \sim \text{Normal}(\nu_\rho, \Omega_\rho);$$
$$\sigma^2 \sim \text{Gamma}(a, b),$$

where $i = 1, ..., m$, and $j = 1, ..., k$. Let these values equal $\boldsymbol{\phi}^{(0)}$.

For the E-step, we calculate the conditional expectation of the log-likelihood, with respect to the augmented data,

$$G\left(\boldsymbol{\phi} \mid \boldsymbol{\phi}^{(t)}\right) = \text{E}_{\mathbf{z}\|\mathbf{y},\boldsymbol{\phi}^{(t)}}[\log L(\boldsymbol{\phi} \mid \mathbf{z}, \mathbf{y})]$$
$$= -\frac{n}{2}\log 2\pi - \frac{n}{2}\log \mid \mathbf{Q} \mid - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\check{q}_{ij}\left(\text{E}\left[z_i z_j\right] - \text{E}[z_i]X_j\beta - \text{E}[z_j]X_i\beta + X_i X_j \beta^2\right),$$

where t is the current step number and $\check{q}_{ij}$ is element (i, j) in the matrix $\mathbf{Q}^{-1}$.

In the M-step, we maximize $G\left(\boldsymbol{\phi} \mid \boldsymbol{\phi}^{(t)}\right)$ to get $\boldsymbol{\beta}^{t+1}$, $\boldsymbol{\rho}^{t+1}$, and $\left[\sigma^2\right]^{(t+1)}$ for the next step.

$$\boldsymbol{\beta}^{(t+1)} = \arg\max_{\boldsymbol{\beta}} G\left(\boldsymbol{\beta} \mid \boldsymbol{\rho}^{(t)}, \left[\boldsymbol{\sigma}^{\mathbf{2}}\right]^{(t)}\right);$$
$$\boldsymbol{\rho}^{(t+1)} = \arg\max_{\boldsymbol{\rho}} G\left(\boldsymbol{\rho} \mid \boldsymbol{\beta}^{(t+1)}, \left[\boldsymbol{\sigma}^{\mathbf{2}}\right]^{(t)}\right);$$
$$\left[\boldsymbol{\sigma}^{\mathbf{2}}\right]^{(t+1)} = \arg\max_{[\sigma^2]} G\left(\left[\boldsymbol{\sigma}^{\mathbf{2}}\right] \mid \boldsymbol{\beta}^{(t+1)}, \boldsymbol{\rho}^{(t+1)}\right).$$

We replace $\boldsymbol{\phi}^{(t)}$ with $\boldsymbol{\phi}^{(t+1)}$ and repeat the E-step and M-step until all the parameters converge. Parameter estimates from the EM algorithm converge to the MLE estimates [Wu 1983].

It is worth noting that the analytical solution for all the parameters is not always possible. Consider the maximization with respect to the autocorrelation variance parameter σ^2

$$\left[\sigma^2\right]^{(t+1)} = \arg\max_{[\sigma^2]} G\left(\boldsymbol{\phi} \mid \boldsymbol{\phi}^{(t)}\right)$$
$$\frac{\partial \log L}{\partial[\sigma^2]} = \frac{\partial}{\partial[\sigma^2]}\left(-\frac{1}{2}\log \mid \mathbf{Q} \mid -\frac{1}{2}(\mathbf{z} - \mathbf{X}\beta)^\top \mathbf{Q}^{-1}(\mathbf{z} - \mathbf{X}\beta)\right). \qquad (1)$$

The first term on the the right hand side of Equation (1) is

$$\frac{\partial}{\partial\left[\sigma^2\right]}\log \mid \mathbf{Q} \mid = \frac{\partial}{\partial\left[\sigma^2\right]}\log\left| I_n + \left[\sigma^2\right]\left(I_n - \sum_{i=1}^{k}\rho_i \mathbf{W}_i\right)^{-1}\left(\left(I_n - \sum_{i=1}^{k}\rho_i \mathbf{W}_i\right)^{-1}\right)^\top\right|.$$

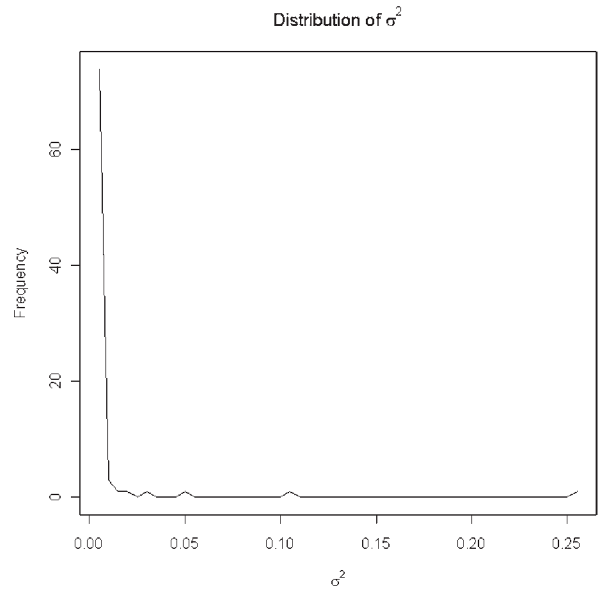


Fig. 1. An estimated probability distribution for σ^2, variance of $\boldsymbol{\theta}$. Maximum likelihood methods, such as the Expectation-Maximization method, will choose $\sigma^2 = 0$, a degenerate solution.

The second term is

$$\frac{\partial}{\partial\left[\sigma^2\right]}(\mathbf{z}-\mathbf{X}\beta)^\top \mathbf{Q}^{-1}(\mathbf{z}-\mathbf{X}\beta)$$

$$= \frac{\partial}{\partial\left[\sigma^2\right]}(\mathbf{z}-\mathbf{X}\beta)^\top \left(I_n + \left[\sigma^2\right] \left(I_n - \sum_{i=1}^{k} \rho_i W_i\right)^{-1} \left(\left(I_n - \sum_{i=1}^{k} \rho_i W_i\right)^{-1}\right)^\top \right)^{-1} (\mathbf{z}-\mathbf{X}\beta).$$

This is not solvable analytically, so numerical methods are needed to get the estimators for this parameter and for ρ.

As it happens, the EM algorithm produces a degenerate solution. This is because it estimates the mode of σ^2, the error term of the autocorrelation term θ, which is at 0 (see Figure 1), and produces a singular variance-covariance matrix estimate using the Hessian approximation. Thus we have to find another solution.

3.3. Full Bayesian Solution

We turn to Bayesian methods. Since the observed choice of a consumer is decided by his/her unobserved preference, this model has a hierarchical structure, so it is natural to think of using a hierarchical Bayesian method. In addition to the preceding model specification, prior distributions for each of the highest-level parameters in the model are also required. As before, $\mathbf{y}$ is the observed dichotomous choice and is calculated by the latent preference, $\mathbf{z}$. With Markov Chain Monte Carlo, we generate draws from a series of full conditional probability distributions, derived from the joint distribution. We summarize the forms of the full conditional distributions of all the parameters to estimate in Table I, and in full in Appendix A.2.

Given the observed choice of a consumer, the latent variable $\mathbf{z}$ is generated from a truncated normal distribution with a mean of $\mathbf{X}\beta + \boldsymbol{\theta}$ with unit error. The prior

Table I. Cyclical Conditional Sampling Steps for Markov Chain Monte Carlo

Parameter	Density	Draw Type
$\mathbf{z}$	$\mathrm{TrunNormal}_n(\mathbf{X}\boldsymbol{\beta}+\boldsymbol{\theta}, I_n)$	Parallel
$\boldsymbol{\beta}$	$\mathrm{Normal}_n(\boldsymbol{\nu}_\beta, \boldsymbol{\Omega}_\beta)$	Parallel
$\boldsymbol{\theta}$	$\mathrm{Normal}_n(\boldsymbol{\nu}_\theta, \boldsymbol{\Omega}_\theta)$	Parallel
σ^2	$\mathrm{InvGamma}(a, b)$	Single
ρ_i	Metropolis step	Sequential

distributions of the parameters (shown in Table I) are adapted from priors proposed by Smith and LeSage [2004].

— $\boldsymbol{\beta}$ follows a multivariate normal distribution with mean $\boldsymbol{\nu}_\beta$ and variance $\boldsymbol{\Omega}_\beta$.
— σ^2 follows an inverse gamma distribution with parameters a and b.
— Each ρ_i follows a normal distribution with mean ν_ρ and variance Ω_ρ.

The sampler algorithm was constructed in the R programming language, including a mechanism to generate data from the model. Validation of the algorithm was conducted using the method of posterior quantiles [Cook et al. 2006], ensuring the correctness of the code for all analyses. Posterior quantiles is a simulation-based method that generates data from the model and verifies that the software can generate parameter estimate randomly around the true parameter. For a detailed description of the implementation, please see Appendix A.3.

3.4. Sensitivity to Prior Specification

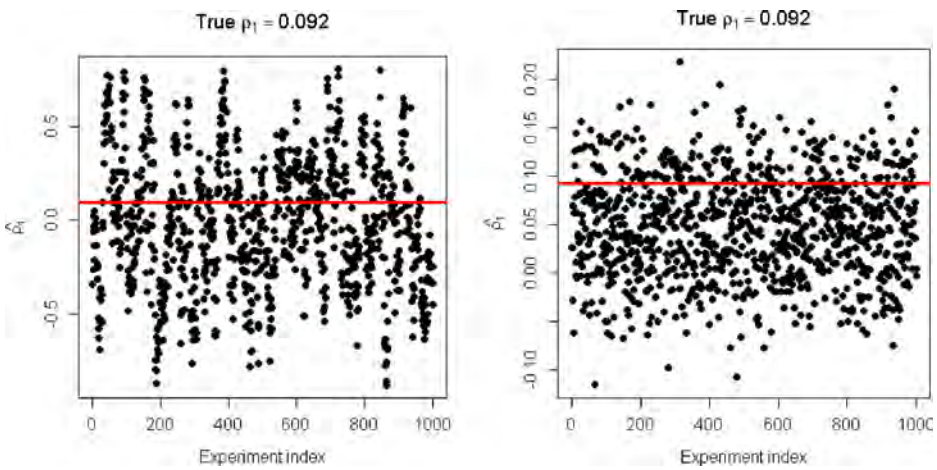
We test the performance of the sampler using prior distributions that are closer to our chosen model than the trivial priors used to check the model code in order to assess the behavior of the algorithm under nonideal conditions. We demonstrate on data simulated from the model, using two preexisting network configurations, and specify different prior distributions for each parameter. To demonstrate, we choose a prior distribution for ρ_1 with high variance, $\rho \sim \mathrm{Normal}(0, 100)$, . As shown in Figure 2(a), the posterior draws of ρ_1 have high temporal autocorrelation. To compare, we choose a narrow prior distribution for ρ_1, $\rho_1 \sim \mathrm{Normal}(0.05, 0.05^2)$; the posterior draws for ρ_1 are shown in Figure 2(b), and the temporal autocorrelation is considerably smaller. With the volume of data under consideration, it is clear that the posterior distribution of ρ is sensitive to its prior distribution.

In most of our examples, we do not have a great deal of prior information available on any network parameters, suggesting that most of our analyses will be conducted with minimally informative prior distributions. With such high autocorrelation between sequential draws, the effective sample size is extremely small. We therefore use a high degree of thinning to produce a series of uncorrelated draws from the posterior.

4. APPLICATIONS

4.1. Auto Purchase Data of Yang and Allenby [2003]

We use Yang's [2003] Japanese car data to compare the findings of our method with those in the original study. The data consists of information on 857 purchase decisions of midsize cars; the dependent variable is whether the car purchased was Japanese (y_m = 1) or otherwise (y_m = 0). All the car models in the data are substitutable and have roughly similar prices.



(a) $\rho \sim \text{Normal}(0, 100)$ (b) $\rho \sim \text{Normal}(0.05, 0.05^2)$

Fig. 2. Testing the sensitivity of the inference of an autocorrelation parameter ρ_1 to the prior distribution. (a) The Markov Chain for a weakly informative prior distribution is consistent with the "oracle" value ρ_1, but the chain has significant temporal autocorrelation. (b) The Markov Chain with a strongly informative prior distribution has much less temporal autocorrelation, but is beholden to its prior distribution more than the data.

An important question of interest is whether the preferences for Japanese cars among consumers are interdependent or not. The interdependence in the network is measured by geographical location, where $W_{ij} = 1$, if consumer i and j live in the same zip code, and 0, otherwise. Explanatory variables include actors' demographic information such as age, annual household income, ethnic group, education and other information such as the price of the car, whether the optional accessories are purchased for the car, latitude and longitude of the actor's location. To construct a network, Yang and Allenby use the fact that the consumers' home addresses are in the same zip code as the indicator of a connection. Thus the network structure **W**, the cohesion, is joint membership in the same geographic area.

By comparing the parameters of Yang and Allenby's model to those for m-NAP on the same dataset, with the same underlying definition of network structure, we contrast our approaches and demonstrate the value of separating the impact of various network autocorrelations. The comparison of the coefficient estimates from Yang and Allenby and our Bayesian solution is shown in Figure 3 , for both explanatory variables and for network autocorrelations. We specify a second network term $\mathbf{W}_2$ to be the structural equivalence of two consumers, calculated as the simple adjacency distance between the two vectors representing individuals' connections to other individuals in the network. In an undirected network with nonweighted edges, the adjacency distance between two nodes i and j is the number of individuals who have different relationships to i and j respectively,

$$d_{ij} = \sqrt{\sum_{k=1, k \neq i,j}^{N} \left(A_{ik} - A_{jk}\right)^2}, \tag{2}$$

where $A_{ik} = 1$ if nodes i and k are neighbors, and 0 otherwise. The larger the d between nodes i and j, the less structurally equivalent they are. We use the inverse of d_{ij} plus one, in order to construct a measure with a positive, finite relationship with role

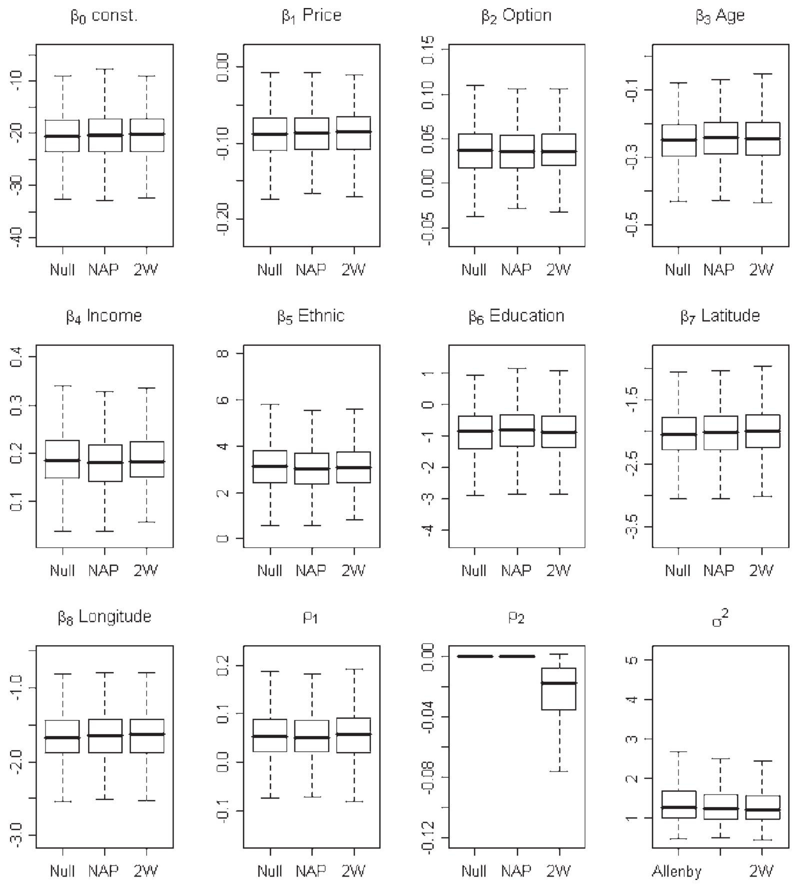


Fig. 3. A comparison of coefficient estimates between the Yang-Allenby method and m-NAP with 1 or 2 networks. The models give similar results, while noting that there is now a negative and statistically significant effect on the network representing structural equivalence. β_0: coefficient of constant term, β_1: coefficient of $\mathbf{X}_1$, car price; β_2: coefficient of $\mathbf{X}_2$, car's optional accessory; β_3: coefficient of $\mathbf{X}_3$, consumer's age; β_4: coefficient of $\mathbf{X}_4$, consumer's income; β_5: coefficient of $\mathbf{X}_5$, consumer's ethnicity; β_6: coefficient of $\mathbf{X}_6$, residence longitude; β_7: coefficient of $\mathbf{X}_7$, residence latitude; ρ_1: coefficient of first network autocorrelation term, $\mathbf{W}_1$, cohesion; ρ_2: coefficient of the second network autocorrelation term, $\mathbf{W}_2$, structural equivalence; σ^2: estimated variance of the error term in autocorrelation.

equivalence, so that $s_{ij} = \frac{1}{d_{ij}+1}$. In our setting, a random element A_{ij} in Equation (2) is from matrix $\mathbf{W}_1$, so d_{ij} is the adjacency distance between any two vectors $\mathbf{A}_i$ and $\mathbf{A}_j$. These represent consumer i's connections, and consumer j's connections to all the other consumers in the data, respectively. The inverse of d_{ij} with an addition of 1 (to avoid zero as denominator), s_{ij}, becomes an element of the structural equivalence matrix $\mathbf{W}_2$.

The comparison is shown in Figure 3. Each box contains the estimates of one parameter from three methods: from left to right, Yang and Allenby, NAP with 1 network, and NAP with 2 networks. All the coefficient estimates, $\hat{\beta}_i$, $\hat{\rho}_2$, and $\hat{\sigma}^2$ of the three methods have similar mean, standard deviation, and credible intervals. One interesting thing here is the effect size of the second network: structural equivalence, has a significant negative effect. This suggests a diminishing cluster effect; when the number of people in the cluster gets bigger, the influence does not increase proportionally.

4.2. Caller Ring-Back Tone Usage In A Mobile Network

We use m-NAP to investigate the purchase of Caller Ring Back Tones (CRBT) within a cellular phone network, a technology of increasing interest around the world. When someone calls the subscriber of a CRBT, the caller does not hear the standard ring-back tone but instead hears a song, joke, or other message chosen by the subscriber until the subscriber answers the phone, or the mailbox takes over. As soon as a CRBT is downloaded, it is set as the default ring-back tone, and triggered automatically by all phone calls. Our data were obtained from a large Indian telecommunications company (source and raw data confidential). We have cellular phone call records and CRBT purchase records over a three-month period, and phone account holders' demographic information such as age and gender. We extract a community of 597 users that are highly internally connected from a population with approximately 26 million unique users using the Transitive Clustering and Pruning (T-CLAP) algorithm [Zhang et al. 2011]. Within this cluster, network edges are specified between users who call each other during the period of observation, as mutual symmetric connection implies equal and stable relationships [Hanneman and Riddle 2005], rather than weaker relationships or calls related to businesses (inquiries or telemarketers).

We include several explanatory variables in this model:

— the gender of the cellular phone account holder;
— the age of the account holder;
— the number of unique outbound connections from the user (known as the outdegree).

From our original network, we derive two matrices corresponding to cohesion and structural equivalence. Cohesion assumes callers who make phone calls to each other will hear the called party's CRBT thus they would be more likely to buy that ring-back tone or get interested in CRBT and eventually adopt the technology. Since the number of people each caller calls differs drastically, we normalize the cohesion matrix by dividing each row by the total number of adopters, to make the matrix element the percentage of adoption. Structural equivalence is once again defined as the adjacency distance between two callers. Here it is less clear that there is an obvious mechanism for how structural equivalence can impact adoption, as it relates to a relationship that does not expose the caller to the CRBT.

We show estimates for each parameter of the model in Figure 4. Again, we observe a significant negative effect for structural equivalence. This new network autocorrelation, with a coefficient of opposite sign from that of the first network autocorrelation $\mathbf{W}_1$, cannot be identified by any earlier models.

5. CONCLUSION

We have introduced a new auto-probit model to study binary choice of a group of actors that have multiple network relationships among them. Such a model can be used to compare influences from different networks on binary choices of individuals, for example whether to adopt a new technology or not, or purchase a new product or not, when they are embedded in multiple types of networks. We specified the fitting of the model for both EM and hierarchical Bayesian methods. We found that the EM solution cannot estimate the parameters for this particular model, thus only the hierarchical Bayesian solution can be used here. We also validated our Bayesian solution by using the posterior quantiles method and the results show our software returns accurate estimates. Finally we compare the estimates returned by Yang and Allenby, NAP with one network effect (cohesion), and NAP with two network effects (cohesion and structural equivalence), by using real data.

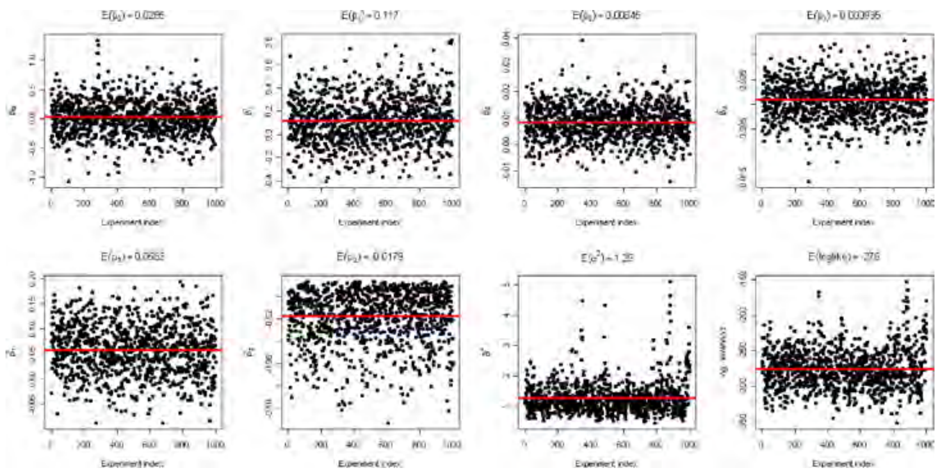


Fig. 4. Trace plot of CRBT network parameters. Description of parameters: β_0: coefficient of constant term; β_1: coefficient of consumer's gender; β_2: coefficient of consumer's age; β_3: coefficient of number of called contacts; ρ_1: coefficient of first network autocorrelation term, $\mathbf{W}_1$, cohesion; ρ_2: coefficient of the second network autocorrelation term, $\mathbf{W}_2$, structural equivalence; σ^2: estimated variance of the error term in autocorrelation; loglike: log-likelihood of $\mathbf{y}$.

We want to ensure that the approach can recover variability in the network effect size. Assuming $\mathbf{W}\boldsymbol{\theta}$ has strong effect, we will vary ρ's true value from a small number to a large number, and observe whether our solution can capture the variation.

Finally we also want to study how multicollinearities between $\mathbf{X}$s, and between $\mathbf{X}$ and $\mathbf{W}\boldsymbol{\theta}$ affect estimated results.

APPENDIXES

A.1. EM Solution Implementation

A.1.1. Deduction. First, get the distribution of $\boldsymbol{\theta}$.

$$\left(I_n - \sum_{i=1}^{k} \rho_i \mathbf{W}_i\right) \boldsymbol{\theta} = \mathbf{u}$$

$$\boldsymbol{\theta} = \left(I_n - \sum_{i=1}^{k} \rho_i \mathbf{W}_i\right)^{-1} \mathbf{u}$$

$$\boldsymbol{\theta} \sim \text{Normal}\left(0, \sigma^2 \left(I_n - \sum_{i=1}^{k} \rho_i \mathbf{W}_i\right)^{-1} \left(\left(I_n - \sum_{i=1}^{k} \rho_i \mathbf{W}_i\right)^{-1}\right)^{\top}\right).$$

Then get the distribution of $\mathbf{z}|\beta, \boldsymbol{\rho}, \sigma^2$:

$$\mathbf{z} \sim \text{Normal}\left(\mathbf{X}\beta, \mathbf{Q}\right), \text{ where } \mathbf{Q} = I_n + \sigma^2 \left(I_n - \sum_{i=1}^{k} \rho_i \mathbf{W}_i\right)^{-1} \left(\left(I_n - \sum_{i=1}^{k} \rho_i \mathbf{W}_i\right)^{-1}\right)^{\top}.$$

The joint distribution of $\mathbf{y}$ and $\mathbf{z}$ can transformed as

$$\begin{aligned} p(\mathbf{y}|\mathbf{z})p\left(\mathbf{z}|\beta,\boldsymbol{\rho},\sigma^2\right) &= p\left(\mathbf{y},\mathbf{z}|\beta,\boldsymbol{\rho},\sigma^2\right) \\ &= p\left(\mathbf{z}|\mathbf{y};\beta,\boldsymbol{\rho},\sigma^2\right)p(\mathbf{y}). \end{aligned} \tag{3}$$

The right side of Equation (3) contains two distributions we already have, as shown in the following.

$$\begin{aligned} p(\mathbf{y}) &= \frac{\frac{1}{\sqrt{2\pi}}\exp\left(-\frac{1}{2}(\mathbf{z}-\mathbf{X}\beta)^\top(\mathbf{z}-\mathbf{X}\beta)\right)}{\Phi(\mathbf{X}\beta)}\mathbb{I}(\mathbf{z}>0) \\ \mathbf{z}|\beta,\boldsymbol{\rho},\sigma^2 &\sim \mathrm{Normal}(\mathbf{X}\beta,\mathbf{Q}) \\ \mathbf{z}|\mathbf{y},\mathbf{X};\beta,\boldsymbol{\rho},\sigma^2 &\sim \mathrm{TrunNormal}(\mathbf{X}\beta,\mathbf{Q}). \end{aligned}$$

Consider parameter β only,

$$\begin{aligned} p(\beta,\mathbf{z}|\mathbf{y}) &= p(\beta|\mathbf{z},\mathbf{y})p(\mathbf{z}|\mathbf{y}) \\ \mathbf{z}|\mathbf{y},\mathbf{X};\beta &\sim \mathrm{TrunNormal}(\mathbf{X}\beta,\mathbf{Q}). \end{aligned}$$

Assume $\mathrm{Var}(\mathbf{z})$=1,

$$\begin{aligned} L(\beta|\mathbf{z}) &= \frac{1}{\sqrt{2\pi}}\sum_{i=1}^{n}\exp\left(-\frac{1}{2}(z_i-X_i\beta)^2\right) \\ \hat{\beta} &= \left(\mathbf{X}^\top\mathbf{X}\right)^{-1}\mathbf{X}^\top\mathbf{R},\ \text{where } \mathbf{R} = \mathrm{E}[\mathbf{z}|\boldsymbol{\theta},\mathbf{y}]. \end{aligned}$$

Then include parameters, $\boldsymbol{\rho}$ and σ^2.

$$\begin{aligned} \mathrm{E}[\mathbf{z}]^{(t+1)} &= \mathrm{E}\left[\mathbf{z}\,\middle|\,\mathbf{y},\beta^{(t)}\right] = f\left(\beta^{(t)},\mathbf{y}\right) \\ \log L\left(\beta,\boldsymbol{\rho},\sigma^2\,\middle|\,\mathbf{z}\right) &= \log p\left(\mathbf{z}\,\middle|\,\beta,\boldsymbol{\rho},\sigma^2\right) \\ &= \log\prod_{i=1}^{n}p\left(z_i\,\middle|\,\beta,\boldsymbol{\rho},\sigma^2\right) \\ &= \sum_{i=1}^{n}\log\frac{1}{\sqrt{2\pi|\mathbf{Q}|}} - \frac{1}{2}(\mathbf{z}-\mathbf{X}\beta)^\top\mathbf{Q}^{-1}(\mathbf{z}-\mathbf{X}\beta) \\ &= \sum_{i=1}^{n}\log\frac{1}{\sqrt{2\pi|\mathbf{Q}|}} - \left(\frac{1}{2}\mathbf{z}^\top\mathbf{Q}^{-1}\mathbf{z} - \mathbf{z}^\top\mathbf{Q}^{-1}\mathbf{X}\beta - \mathbf{X}^\top\beta\mathbf{Q}^{-1}\mathbf{z} + \mathbf{X}^\top\beta\mathbf{Q}^{-1}\mathbf{X}\beta\right). \end{aligned} \tag{4}$$

If we decompose these matrices as vector products, then

$$\begin{aligned} (4) &= \sum_{i=1}^{n}\log\frac{1}{\sqrt{2\pi|\mathbf{Q}|}} - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\left(z_i-X_i\beta\right)\check{q}_{ij}\left(z_j-X_j\beta\right) \\ &= \sum_{i=1}^{n}\log\frac{1}{\sqrt{2\pi|\mathbf{Q}|}} - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\check{q}_{ij}\left(z_iz_j - z_iX_j\beta - z_jX_i\beta + X_iX_j\beta^2\right), \end{aligned}$$

where $\check{q}_{ij}$ is the element in $\check{\mathbf{Q}}$, and $\check{\mathbf{Q}} = \mathbf{Q}^{-1}$.

A.1.2. Expectation Step. In the expectation step, get the expected log-likelihood of parameters.

$$
\begin{aligned}
Q\left(\phi|\phi^{(t)}\right) &= \mathrm{E}_{\mathbf{z}|\mathbf{y},\phi^{(t)}}[\log L(\phi|\mathbf{z},\mathbf{y})] \\
&= \mathrm{E}\left[\sum_{i=1}^{n}\log\frac{1}{\sqrt{2\pi|\mathbf{Q}}}\right] - \mathrm{E}\left[\frac{1}{2}(\mathbf{z}-\mathbf{X}\beta)^{\top}\mathbf{Q}^{-1}(\mathbf{z}-\mathbf{X}\beta)\right] \\
&= -\frac{n}{2}\log 2\pi - \frac{n}{2}\log|\mathbf{Q}| - \frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\check{q}_{ij}\left(\mathrm{E}\left[z_i z_j\right] - \mathrm{E}\left[z_i\right]X_j\beta - \mathrm{E}\left[z_j\right]X_i\beta + X_i X_j \beta^2\right),
\end{aligned}
$$

where ϕ is the parameter set, and t is the number of steps.

A.1.3. Maximization Step. In the maximization step, get the parameter estimates maximizing the expected log-likelihood. First, estimate β.

$$
\begin{aligned}
\beta^{(t+1)} &= \arg\max_{\boldsymbol{\beta}} Q\left(\phi|\phi^{(t)}\right) \\
&= \arg\max_{\boldsymbol{\beta}} \sum_{i=1}^{n}\log\frac{1}{\sqrt{2\pi|\mathbf{Q}|}} - \frac{1}{2}(\mathbf{z}-\mathbf{X}\beta)^{\top}\mathbf{Q}^{-1}(\mathbf{z}-\mathbf{X}\beta). \qquad (5)
\end{aligned}
$$

If we directly apply the analytical method to solve Equation (5), then

$$
\begin{aligned}
\frac{\partial\log L}{\partial\beta} &= \frac{\partial}{\partial\beta}\left(-\frac{1}{2}(\mathbf{z}-\mathbf{X}\beta)^{\top}\mathbf{Q}^{-1}(\mathbf{z}-\mathbf{X}\beta)\right) \\
\frac{\partial}{\partial\beta}(\mathbf{z}-\mathbf{X}\beta)^{\top}\mathbf{Q}^{-1}(\mathbf{z}-\mathbf{X}\beta) &= \frac{\partial}{\partial\beta}\left(\mathbf{z}^{\top}\mathbf{Q}^{-1}\mathbf{z} - \mathbf{z}^{\top}\mathbf{Q}^{-1}\mathbf{X}\beta - \beta^{\top}\mathbf{X}^{\top}\mathbf{Q}^{-1}\mathbf{z} + \beta^{\top}\mathbf{X}^{\top}\mathbf{Q}^{-1}\mathbf{X}\beta\right) \\
&= -\mathbf{z}^{\top}\mathbf{Q}^{-1}\mathbf{X} - \mathbf{X}^{\top}\mathbf{Q}^{-1}\mathbf{z} + \mathbf{X}^{\top}\mathbf{Q}^{-1}\mathbf{X}\beta. \qquad (6)
\end{aligned}
$$

Set Equation (6) as 0, then

$$
\begin{aligned}
-\mathbf{z}^{\top}\mathbf{Q}^{-1}\mathbf{X} - \mathbf{X}^{\top}\mathbf{Q}^{-1}\mathbf{z} + \mathbf{X}^{\top}\mathbf{Q}^{-1}\mathbf{X}\beta &= 0 \\
\hat{\beta} &= \left(\mathbf{X}^{\top}\mathbf{Q}^{-1}\mathbf{X}\right)^{-1}\mathbf{X}^{\top}\mathbf{Q}^{-1}\mathbf{R}.
\end{aligned}
$$

Second, estimate parameter $\boldsymbol{\rho}$

$$
\boldsymbol{\rho}^{(t+1)} = \arg\max_{\boldsymbol{\rho}} Q\left(\phi|\phi^{(t)}\right).
$$

Assume $\boldsymbol{\rho} = \{\rho_1, ..., \rho_k\}$, without losing any generalizabiliy, ρ_1 can be estimated as

$$
\rho_1^{(t+1)} = \arg\max_{\rho_1} Q\left(\phi|\phi^{(t)}\right)
$$

$$
\begin{aligned}
\frac{\partial\log L}{\partial\rho_1} &= \frac{\partial}{\partial\rho_1}\left(-\frac{1}{2}\log|\mathbf{Q}| - \frac{1}{2}(\mathbf{z}-\mathbf{X}\beta)^{\top}\mathbf{Q}^{-1}(\mathbf{z}-\mathbf{X}\beta)\right) \\
\frac{\partial}{\partial\rho_1}\log|\mathbf{Q}| &= -\mathrm{tr}\left(\mathbf{W}_1\mathbf{Q}^{-1}\right) \\
\frac{\partial}{\partial\rho_1}(\mathbf{z}-\mathbf{X}\beta)^{\top}\mathbf{Q}^{-1}(\mathbf{z}-\mathbf{X}\beta) &= \frac{\partial}{\partial\rho_1}\left(\mathbf{z}^{\top}\mathbf{Q}^{-1}\mathbf{z} - \mathbf{z}^{\top}\mathbf{Q}^{-1}\mathbf{X}\beta - \beta^{\top}\mathbf{X}^{\top}\mathbf{Q}^{-1}\mathbf{z} + \beta^{\top}\mathbf{X}^{\top}\mathbf{Q}^{-1}\mathbf{X}\beta\right).
\end{aligned}
$$

It is impossible to get an analytical solution for ρ_i.

Third, estimate parameter σ^2. Let $\sigma^2 = \left[\sigma^2\right]$

$$\left[\sigma^2\right]^{(t+1)} = \arg\max_{[\sigma^2]} Q\left(\phi|\phi^{(t)}\right)$$
$$\frac{\partial \log L}{\partial[\sigma^2]} = \frac{\partial}{\partial[\sigma^2]}\left(-\frac{1}{2}\log|\mathbf{Q}| - \frac{1}{2}(\mathbf{z}-\mathbf{X}\beta)^\top \mathbf{Q}^{-1}(\mathbf{z}-\mathbf{X}\beta)\right). \quad (7)$$

The first term in the the right-hand side of the preceeding equation is

$$\frac{\partial}{\partial\left[\sigma^2\right]}\log|\mathbf{Q}| = \frac{\partial}{\partial\left[\sigma^2\right]}\log\left|I_n + \left[\sigma^2\right]\left(I_n - \sum_{i=1}^{k}\rho_i \mathbf{W}_i\right)^{-1}\left(\left(I_n - \sum_{i=1}^{k}\rho_i \mathbf{W}_i\right)^{-1}\right)^\top\right|.$$

The second term is:

$$\frac{\partial}{\partial\left[\sigma^2\right]}(\mathbf{z}-\mathbf{X}\beta)^\top \mathbf{Q}^{-1}(\mathbf{z}-\mathbf{X}\beta)$$
$$= \frac{\partial}{\partial\left[\sigma^2\right]}(\mathbf{z}-\mathbf{X}\beta)^\top\left(I_n + \left[\sigma^2\right]\left(I_n - \sum_{i=1}^{k}\rho_i W_i\right)^{-1}\left(\left(I_n - \sum_{i=1}^{k}\rho_i W_i\right)^{-1}\right)^\top\right)^{-1}(\mathbf{z}-\mathbf{X}\beta).$$

This is again not solvable by using an analytical method.

A.2. Markov Chain Monte Carlo estimation

The Markov Chain Monte Carlo method generates a sequence of draws that approaches the posterior distribution of interest. Our solution consists of the following steps.

Step 1. Generate $\mathbf{z}$; $\mathbf{z}$ follows a truncated normal distribution.

$$\mathbf{z} \sim \text{TrunNormal}_n(\mathbf{X}\beta + \boldsymbol{\theta}, I_n),$$

where I_n is the $n \times n$ identity matrix. If $y_i = 1$, then $z_i \geq 0$, if $y_i = 0$, then $z_i < 0$.

Step 2. Generate β, $\beta \sim \text{Normal}(\boldsymbol{\nu}_\beta, \boldsymbol{\Omega}_\beta)$.

(1) define β_0, where

$$\beta_0 = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

(2) define $\mathbf{D} = hI_n$, $\mathbf{D}$ is a baseline variance matrix, corresponding to the prior $p(\beta)$, where h is a large constant, e.g. 400.

$$\mathbf{D}^{-1} = \begin{bmatrix} \sigma_0^2 & 0 & \dots & 0 \\ 0 & \sigma_0^2 & \dots & 0 \\ \vdots & \vdots & \dots & \vdots \\ 0 & 0 & \dots & \sigma_0^2 \end{bmatrix}$$

Set σ_0^2 as $\frac{1}{400}$, a small number close to 0, compared with $\mathrm{Normal}(0, 1)$, where $\sigma_0^2 = 1$

(3) $\mathbf{\Omega}_\beta = \left(\mathbf{D}^{-1} + \mathbf{X}^\top\mathbf{X}\right)^{-1}$.

This is because

$$\mathbf{z} = \mathbf{X}\beta + \boldsymbol{\theta} + \boldsymbol{\epsilon}$$
$$\beta = \mathbf{X}^{-1}(\mathbf{z} - \boldsymbol{\theta} - \boldsymbol{\epsilon})$$

$\therefore \boldsymbol{\beta} \sim \mathrm{Normal}\left(\mathbf{X}^{-1}(\mathbf{z} - \boldsymbol{\theta}),\ (\mathbf{X}^\top\mathbf{X})^{-1}\right)$.

Based on law of initial values, $\mathbf{\Omega}_\beta = \left(\mathbf{D}^{-1} + \mathbf{X}^\top\mathbf{X}\right)^{-1}$

(4) Then $\boldsymbol{\nu}_\beta$ can be represented by $\boldsymbol{\nu}_\beta = \mathbf{\Omega}_\beta\left(\mathbf{X}^\top(\mathbf{z} - \boldsymbol{\theta}) + \mathbf{D}^{-1}\right)$.

Step 3. Generate $\boldsymbol{\theta}$, $\boldsymbol{\theta} \sim \mathrm{Normal}(\boldsymbol{\nu}_\theta, \mathbf{\Omega}_\theta)$.

(1) First, define $\mathbf{B} = I_n - \sum_i \rho_i \mathbf{W}_i$

$$\boldsymbol{\theta} = \sum_i \rho_i \mathbf{W}_i + \mathbf{u}$$
$$\left(I_n - \sum_i \rho_i \mathbf{W}_i\right)\boldsymbol{\theta} = \mathbf{u}$$
$$\mathbf{B}\boldsymbol{\theta} = \mathbf{u}$$
$$\boldsymbol{\theta} = \mathbf{B}^{-1}\mathbf{u}.$$

Let $\mathrm{Var}(\mathbf{u}) = \sigma^2 I_n$

$$\begin{aligned}\mathrm{Var}(\boldsymbol{\theta}) &= \mathrm{Var}\left(\mathbf{B}^{-1}\mathbf{u}\right)\\ &= (\mathbf{B}^\top\mathbf{B})^{-1}\sigma^2 I_n\\ &= \left(\frac{\mathbf{B}^\top\mathbf{B}}{\sigma^2}\right)^{-1}.\end{aligned}$$

(2) Then $\mathbf{\Omega}_\theta = \left(I_n + \frac{\mathbf{B}^\top\mathbf{B}}{\sigma^2}\right)^{-1}$. We then add an offset I_n to $\frac{\mathbf{B}^\top\mathbf{B}}{\sigma^2}$. So $\mathbf{\Omega}_\theta = \left(I_n + \frac{\mathbf{B}^\top\mathbf{B}}{\sigma^2}\right)^{-1}$.

(3) $\boldsymbol{\nu}_\theta = \mathbf{\Omega}_\theta(\mathbf{z} - \mathbf{X}\beta)$, since $\boldsymbol{\theta} = (\mathbf{z} - \mathbf{X}\beta) - \boldsymbol{\epsilon}$.

Step 4. Generate σ^2, $\sigma^2 \sim \mathrm{InvGamma}(a, b)$

$$a = s_0 + \frac{n}{2}$$
$$b = \frac{2}{\boldsymbol{\theta}^\top\mathbf{B}^\top\mathbf{B}\boldsymbol{\theta} + \frac{2}{q_0}},$$

where s_0 and q_0 are the parameters for the conjugate prior of σ^2, and n is the size of data.

Step 5. Finally we generate coefficient ρ_i for $\mathbf{W}$, using Metropolis-Hasting sampling with a random walk chain.

$$\rho_i^{new} = \rho_i^{old} + \Delta_i,$$

where the increment random variable $\Delta_i \sim \mathrm{Normal}(\nu_\Delta, \Omega_\Delta)$.

The accepting probability α is obtained by

$$\min\left(\frac{|\mathbf{B}_{new}|\exp\left(-\frac{1}{2\sigma^2}\boldsymbol{\theta}^\top\mathbf{B}_{new}^\top\mathbf{B}_{new}\boldsymbol{\theta}\right)}{|\mathbf{B}_{old}|\exp\left(-\frac{1}{2\sigma^2}\boldsymbol{\theta}^\top\mathbf{B}_{old}^\top\mathbf{B}_{old}\boldsymbol{\theta}\right)}, 1\right).$$

A.3. Validation of Bayesian Software

One challenge of Bayesian methods is getting an error-free implementation. Bayesian solutions often have high complexity, and a lack of software causes many researchers to develop their own, greatly increasing the chance of software error; many models are not validated, and many of them have errors and do not return correct estimations. So it is very necessary to confirm that the code returns correct results. The validation of Bayesian software implementations has a short history; we wrote a program using a standard method, the method of posterior quantiles [Cook et al. 2006], to validate our software. This method again is a simulation-based method. The idea is to generate data from the model and verify that the software will properly recover the underlying parameter in a principled way. First, we draw the parameter θ from its prior distribution $p(\Theta)$, then generate data from distribution $p(y \mid \theta)$. If the software is correctly coded, the quantiles of each true parameter should be uniformly distributed with respect to the algorithm output. For example, the 95% credible interval should contain the true parameter with probability 95%. Assume we want to estimate the parameter θ in Bayesian model $p(\theta \mid y) = p(y \mid \theta)p(\theta)$, where $p(\theta)$ is the prior distribution of θ, $p(y \mid \theta)$ is the distribution of data, and $p(\theta \mid y)$ is the posterior distribution. The estimated quantile can be defined as

$$\hat{q}(\theta_0) = \hat{P}(\theta < \theta_0) = \frac{1}{N}\sum_{i=1}^{N}\mathbb{I}(\theta_i < \theta_0),$$

where θ_0 is the true value drawn from the prior distribution; $\hat{\theta}$ is a series of draws from the posterior distribution generated by the software to-be-tested; N is the number of draws in MCMC. The quantile is the probability of a posterior sample smaller than the true value, and the estimated quantile is the number of posterior draws generated by software smaller than the true value. If the software is correctly coded, then the quantile distribution for parameter θ, $\hat{q}(\theta_0)$, should approach $\mathrm{Uniform}(0, 1)$, when $N \to \infty$ [Cook et al. 2006]. The whole process up to now is defined as one replication. If we run a number of replications, we expect to observe a uniform distribution $\hat{q}(\theta_0)$ around θ_0, meaning the posterior should be randomly distributed around the true value.

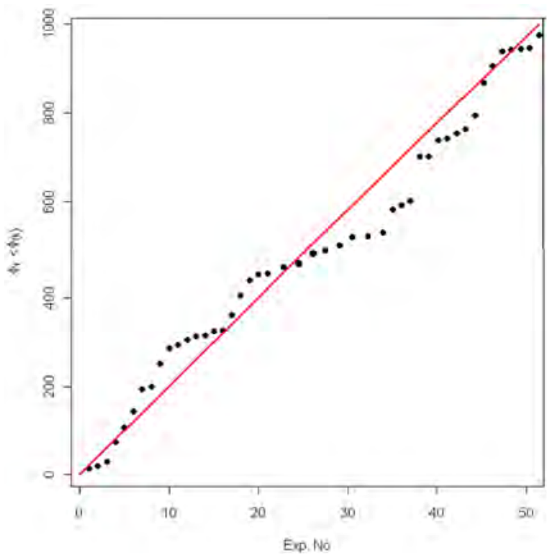


Fig. 5. Distribution of sorted quantiles of parameters, $\beta_1, \beta_2, \rho_1, \rho_2, \sigma^2$, 10 replications of posterior quatiles experiments.

We then demonstrate the simulations we ran. Assume the model we want to estimate is

$$\mathbf{z} = \mathbf{X}_1\beta_1 + \mathbf{X}_2\beta_2 + \boldsymbol{\theta} + \boldsymbol{\epsilon};$$
$$\boldsymbol{\theta} = \rho_1\mathbf{W}_1\boldsymbol{\theta} + \rho_2\mathbf{W}_2\boldsymbol{\theta} + \mathbf{u}.$$

We then specified a prior distribution for each parameter, and use MCMC to simulate the posterior distributions.

$$\boldsymbol{\beta} \sim \mathrm{Normal}(0, 1);$$
$$\sigma^2 \sim \mathrm{InvGamma}(5, 10);$$
$$\boldsymbol{\rho} \sim \mathrm{Normal}\left(0.05, 0.05^2\right).$$

We performed a simulation of 10 replications to validate our hierarchical Bayesian MCMC software. The generated sample size for **X** is 50, so the size of the network structure **W** is 50 by 50. In each replication, we generated 20000 draws from the posterior distribution of all the parameters in $\boldsymbol{\phi}\left(\boldsymbol{\phi} = \left\{\beta_1, \beta_2, \rho_1, \rho_2, \sigma^2\right\}\right)$, and kept one from every 20 draws, yielding 1000 draws for each parameter. We then count the number of draws larger than the true parameters in each replication. If the software is correctly written, each estimated value should be randomly distributed around the true value, so the number of estimates larger than the true value should be uniformly distributed among the 10 replications. We pooled all these quantiles for the five parameters, 50 in total, and the sorted results are shown in Figure 5. The X-axis is the total number of replications of the five parameters—50. The Y-axis is the number of draws larger than true parameters in each replication. The solid red line represents the uniform distribution line. As we can see, the combined results of the five parameters are all uniformly distributed around the true value, thus confirming that our Bayesian software is correctly written; hence we can apply our software to experiments and return correct estimates.

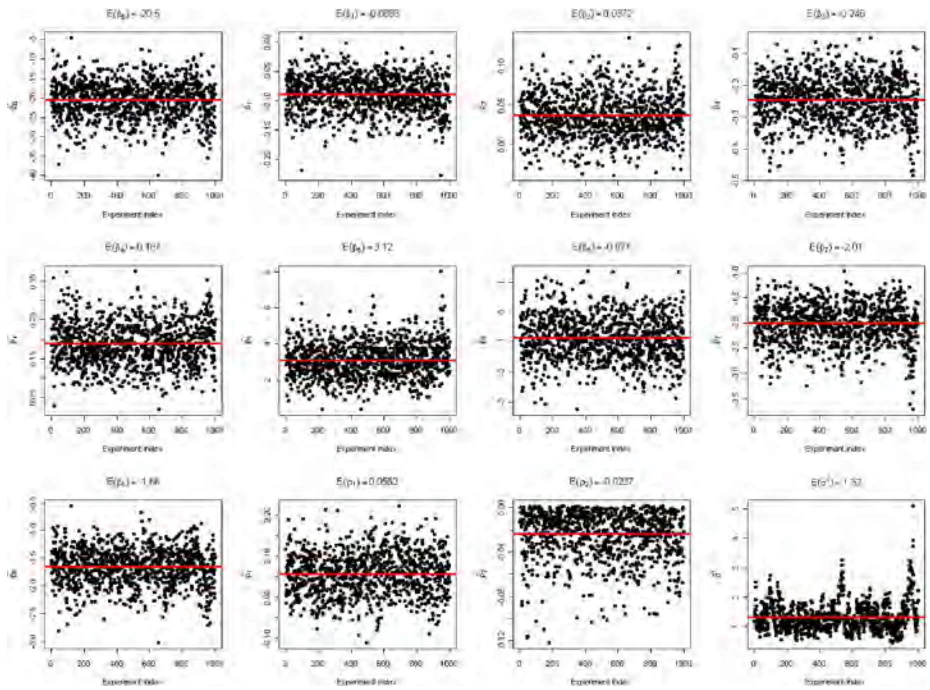


Fig. 6. Trace plot of a two-network auto-probit model. β_0: coefficient of constant term, β_1: coefficient of car price; β_2: coefficient of car's optional accessory; β_3: coefficient of consumer's age; β_4: coefficient of consumer's income; β_5: coefficient of consumer's ethnicity; β_6: coefficient of residence longitude; β_7: coefficient of residence latitude; ρ_1: coefficient of first network autocorrelation term, $\mathbf{W}_1$, cohesion; ρ_2: coefficient of the second network autocorrelation term, $\mathbf{W}_2$, structural equivalence; σ^2: estimated variance of the error term in autocorrelation.

A.4. Solution Diagnostic

We run the MCMC experiment to confirm there is no autocorrelation among draws of each parameter. In this experiment, we set the length of the MCMC chain as 30,000, burn-in as 10,000, and thinning as 20, which is used for removing the autocorrelations between draws. The trace plots generated from our code for the 1000 draws after burn-in and thinning are listed in Figure 6.

We have 12 plots total. Each plot depicts draws for a particular parameter estimation. The first 9 plots, from left to right and top to bottom, are the trace for the β_i, coefficient of independent variables. Each point represents the value of estimated coefficient $\hat{\beta}_i$, and the solid red line represents the mean. We observe all $\hat{\beta}_i$s are randomly distributed around the mean, and the mean is significant, showing the estimation results are valid. The 10th and 11th plots are for the two estimated network effect coefficients $\hat{\rho_1}$ and $\hat{\rho_2}$. We found both $\hat{\rho_i}$ are also significant, and randomly distributed around their means. The only coefficient showing autocorrelation is σ^2.

Note that not all values of ρ_1 and ρ_2 can make $\mathbf{B}$ ($\mathbf{B} = I_n - \rho_1\mathbf{W}_1 - \rho_2\mathbf{W}_2$) invertible. The following plot shows the relationship between the values of ρ_1 and ρ_2, and the invertibility of $\mathbf{B}$. The green area is where $\mathbf{B}$ is invertible, and the red area is otherwise. If the limit draws to the green area, we will have correlated ρ_1 and ρ_2. When we draw ρ_1 and ρ_2 using bivariate normal, there is no apparent correlation between them (see Figure 7). We understand the correlation between ρ_1 and ρ_2 comes from the definition of $\mathbf{W}_1$ and $\mathbf{W}_2$, not the prior noncorrelation.

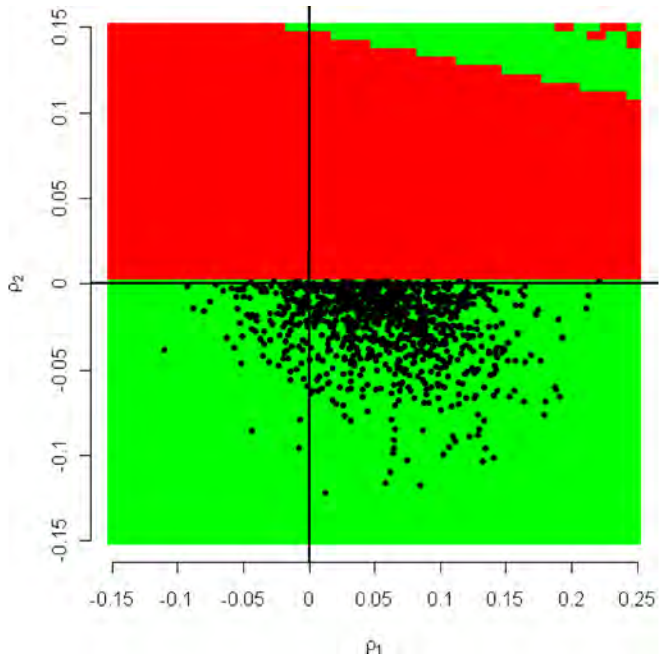


Fig. 7. Scatter plot of ρ_1 and ρ_2 on the valid region for invertible **B**.

A.5. Was a Mixture of Matrices

Yang and Allenby [2003] specified the autoregressive matrix W as a finite mixture of coefficient matrices, each related to a specific covariate.

$$\mathbf{W} = \sum_{i=1}^{n} \phi_i \mathbf{W}_i$$
$$\sum_{i=1}^{n} \phi_i = 1,$$

where i represents the indices of the covariates, $i = 1 \ldots n$. ϕ_i is the corresponding weight of component matrix $\mathbf{W}_i$. $\mathbf{W}_i$ is associated with covariate $\mathbf{X}_i$.

ACKNOWLEDGMENTS

This work was supported in part by AT&T and the iLab at Heinz College, Carnegie Mellon University.

REFERENCES

Agarwal, R., Gupta, A. K., and Kraut, R. 2008. Editorial overview—the interplay between digital and social networks. *Inform. Syst. Res. 19,* 3, 243–252.

Allenby, G. M. and Rossi, P. E. 1998. Marketing models of consumer heterogeneity. *J. Econometrics 89*, 1–2, 57–78.

Anselin, L. 1988. *Spatial Econometrics: Methods and Models.* Springer.

Aral, S., Muchnik, L., and Sundararajan, A. 2009. Distinguishing Influence Based Contagion from Homophily Driven Diffusion in Dynamic Networks. *Proc. Nat. Acad. Sci. 106,* 51, 21544.

Bass, F. M. 1969. A new product growth for model consumer durables. *Manage. Sci. 15*, 5, 215–227.

Bernheim, B. D. 1994. A theory of conformity. *J. Polit. Econ. 102*, 5, 841–877.

Brancheau, C. J. and Wetherbe, C. J. 1990. The adoption of spreadsheet software: Testing innovation diffusion theory in the context of end-user computing. *Inform. Syst. Res. 1*, 2, 115–143.

Burt, R. S. 1987. Social contagion and innovation: Cohesion versus structural equivalence. *Amer. J. Sociology 92,* 6, 1287.

Chatterjee, R. and Eliashberg, J. 1990. The innovation diffusion process in a heterogeneous population: A micromodeling approach. *Manage. Sci. 36,* 9, 1057–1079.

Cook, S. R., Gelman, A., and Rubin, D. B. 2006. Validation of software for Bayesian models using posterior quantiles. *J. Comput. Graph. Statist. 15,* 675–692.

Cressie, N. A. C. 1993. *Statistics for Spatial Data* Revised Ed. Wiley-Interscience, New York.

Doreian, P. 1980. Linear models with spatially distributed data: Spatial disturbances or spatial effects. *Sociol. Methods Res. 9,* 1, 29–60.

Doreian, P. 1982. Maximum likelihood methods for linear models: Spatial effects and spatial disturbance terms. *Sociol. Methods Res. 10,* 3, 243–269.

Doreian, P. 1989. Two regimes of network effects autocorrelation. In *The Small World*, M. Kochen Ed., Ablex Publishing, Norwood, NJ, 280–295.

Fujimoto, K. and Valente, T. W. 2011. Network influence on adolescent alcohol use: Relational, positional, and affiliation-based peer influence. Unpublished manuscript.

Hanneman, R. and Riddle, M. 2005. *Introduction to Social Network Methods*. University of California, Riverside, CA.

Kamakura, W. A. and Russell, G. J. 1989. A probabilistic choice model for market segmentation and elasticity structure. *J. Market. Res. 26,* 4, 379–390.

Leenders, R. T. 1997. *Longitudinal Behavior of Network Structure and Actor Atributes: Modeling Interdependence of Contagion and Selection.* Gordon and Breach, New York, 165–184.

Manski, C. F. 2000. Economic analysis of social interactions. *J. Econ. Persp. 14,* 3, 115–136.

Oinas-Kukkonen, H., Lyytinen, K., and Yoo, Y. 2010. Social networks and information systems: Ongoing and future research streams. *J. Assoc. Inform. Syst. 11*, 2, 61–68.

Ord, K. 1975. Estimation methods for models of spatial interaction. *J. Amer. Statist. Assoc. 70*, 349, 120–126.

Premkumar, G., Ramamurthy, K., and Nilakanta, S. 1994. Implementation of electronic data interchange: An innovation diffusion perspective. *J. Manage. Inform. Syst.* (Special Section: Strategic and Competitive Information Systems Archive) *11*, 2, 157–186.

Rogers, E. M. 1962. *Diffusion of Innovations*. Free Press, New York.

Shalizi, C. R. and Thomas, A. C. 2011. Homophily and contagion are generically confounded in observational social network studies. *Sociol. Meth. Res. 40*, 2, 211–239.

Smirnov, O. A. 2005. Computation of the information matrix for models with spatial interaction on a lattice. *J. Comput. Graph. Statist. 14,* 4, 910–927.

Smith, T. E. and LeSage, J. P. 2004. A Bayesian probit model with spatial dependencies. In *Advances in Econometrics*, Vol. 18: Spatial and Spatiotemporal Econometrics, K. R. Pace and J. P. LeSage Eds., Elsevier, United Kingdom, 127–160.

Thomas, A. C. 2012. The social contagion hypothesis. Comment on *Social Contagion Theory: Examining Dynamic Social Networks and Human Behavior*. In press at *Statistics in Medicine* (to appear).

Wu, C. F. J. 1983. On the convergence properties of the em algorithm. *Annals Statist. 11,* 1, 95–103.

Yang, S. and Allenby, G. M. 2003. Modeling interdependent consumer preferences. *J. Market. Res. XL,* 282–294.

Zhang, B., Krackhardt, D., Krishnan, R., and Doreian, P. 2011. An effective and efficient subpopulation extraction method in large social networks. In *Proceedings of the International Conference on Information Systems*.

Received April 2012; revised October 2012; accepted October 2012