Auto-probit Model for Multiple Regimes of Network Effects

Bin Zhang
Heinz College, iLab
Carnegie Mellon University

Andrew C. Thomas
Department of Statistics, iLab
Carnegie Mellon University

David Krackhardt
Heinz College, iLab
Carnegie Mellon University

Patrick Doreian
Department of Sociology
University of Pittsburgh
and
Faculty of Social Sciences
University of Ljubljana

Ramayya Krishnan
Heinz College, iLab
Carnegie Mellon University

Abstract

Many researchers believe that consumers' decisions are not only decided by their personal tastes, but also by the decisions of people who are in their networks. On the other hand, social scientists are more interested in consumers' dichotomous choice. So an auto-probit model accommodating multiple networks are very useful. However, Current methods to investigate multiple autocorrelated network effects on the same group of actors, embedded in social networks, are primitive. Few solutions have been done for two networks (e.g. Doreian 1989), but not easily on more than three. Even fewer solutions when the dependent variable is binary. We developed two solutions, Expectation-Maximization (E-M) and hierarchical Bayesian, for auto-probit models that accommodate multiple network structures. Both solutions are one of the first in their kinds. The behaviors of the solutions, such as the impact of prior distribution, network structures, and sizes of network effects etc, on parameter estimation will also be studied, by using both real and simulated data.

Contents

1 Introduction

In earlier times, researchers believed consumers' preference and choice are decided by their attributes only, such as age, education, income etc (Kamakura and Russell, 1989; Allenby and Rossi, 1998). More and more researchers realized that those decisions are not only decided by individuals' personal taste, but also by the decisions of people who are connected to, i.e. decisions of neighbors in individuals' social networks, such effect has been called as "peer influence" (Duncan et al., 1968), "neighborhood effects" (Case, 1991), and "conformity" (Bernheim, 1994) etc. Thus autocorrelation models with a network structure term were born to allow researchers to investigate such problems. Eventually such model became insufficient because an actor very often is under the influence of multiple networks. If we want to compare which network effect plays a more influential role in individual's decision, we have very limited tools. Although some models were developed to include two network autocorrelations terms, such as Doreian (1989) two regimes of network effect autocorrelation model. On the other hand, social scientists usually are more interested in consumers' dichotomous choice, such as purchase a product or not, adopt a technology or not etc. With such dichotomous dependent variable, if we have more than two network effects to compare, for example, we want to compare which network of a group of actors', friend, colleague, or family, plays a more influential role on their decision of purchasing iPad or not, no model can solve this kind of problem.

We develop a probit model with multi-network autocorrelation terms to study the competing effects of network. We first use Expectation-Maximization (E-M) algorithm, which is similar to maximum likelihood, a traditional method widely adopted, then use hierarchical Bayesian, a Bayesian statistics, to develop two solutions. Both solutions are one of the first in their kinds. we also study the behaviors of both solutions. for example, how sensitive is the solution with regard to the change of parameters' prior distribution. Preliminary experiments show that E-M method cannot obtain variance-covariance matrix for parameters, thus hierarchical Bayesian is the only option. Our software is also validated by using posterior quantiles method (Cook et al., 2006). We also study whether the solutions can return correctly estimated parameters by using real and simulated data.

The rest of the paper is organized as follows. We discuss the literature on the network effects model in Section 2; Our two solutions for multi-network auto-probit, E-M and hierarchical Bayesian, 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 paper in Section 5.

2 Literatures

The theoretical foundation for our research is social influence on the diffusion process. Diffusion 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). The network model has been widely used to study diffusion since the Bass (1969) model. A history of network models used to study diffusion of innovations

is reviewed by Valente (2005). He categorized the evolution of network models as three stages: macro models, spatial autocorrelation and network (effect) models. In the early era, the probability of adoption is only related to the time that an actor gets exposed to the object of diffusion. In 1969, Bass proposed a famous model to include both rate of innovation and imitation. This model can estimate both the influence from the social network and innovativeness, and is shown in the equation below. Let Y_{t-1} be the proportion that has adopted at time period $t-1$; y_t be the proportion of new adoption at t; b_0 be the coefficient of innovation, which is the probability of initial adoption; and b_1 be the coefficient of innovation.

$$\frac{y_t}{1-Y_{t-1}} = b_0 + b_1 Y_{t-1}$$
$$y_t = b_0 + (b_1 - b_0)Y_{t-1} - b_1 Y_{t-1}^2$$

Bass' model is still at the population level. Its assumption is that everyone in the social network has the same probability of interacting. Such an 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. It seems fair to assume people with closer physical distance communicate more and exert greater influence on each other thus spatial autocorrelation was brought in to the models used in the literature. Simultaneous autoregressive models (SAR) is widely used. The general method of SAR are described in Anselin (1988), and Cressie (1993). The SAR model is an autocorrelation model as follow. We observed actor i's choice y_i, $(i = 1, ..., n)$. $\mathbf{X}$ is a vector of explanatory variables.

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

where coefficient of explanatory variable $\boldsymbol{\beta} = (\beta_1, ..., \beta_m)^\top$ and m is the number of explanatory variables; $\boldsymbol{\epsilon} = (\epsilon_1, ..., \epsilon_n)$; each ϵ_i follows normal distribution Normal$(0, \sigma_i^2)$. The most common method used to fit SAR models has been maximum likelihood, see Ord (1975), Doreian (1980, 1982), and Smirnov (2005).

SAR is still a model for measure of diffusion at the population level, and does not account for whether one actor is more or less likely to adopt based on his network position. It does not show how network structure influence diffusion either. So researchers turned to network models to more accurately reflect these influences. The diffusion network model explains that the initial adoption is based on actor's innovativeness and exposure to sources of influence, and that this influence originates from alters who have already adopted and are able to persuade nonusers to adopt. Before moving on, we need to take a look at event history analysis, which offers some quite useful tools for network analysis. Sometimes the factors that affect adoption also affect the formation of network. Thus it is necessary to collect data at different time periods (panel data), and bring in event history analysis. The purpose of event history analysis is to explain why certain individuals are at a higher risk of experiencing the event of interest than others. The most commonly used analysis methods include failure-time models, survival models, and hazard models etc. (An event is a transition from one status to another, e.g. from non-adoption to adoption.)

Network influences are captured by contagion model. Social contagion is the interpersonal connection over which innovation is transmitted (Burt, 1987). The probability of each actor's adoption increases

when the number or proportion of the adopters in his network increases. The network exposure is defined as below, where E_i is the proportion of actor i's neighbors who have adopted; y is the variable of adoption; and w is social network structure matrix.

$$E_i = \frac{\sum_{j=1}^{N} w_{ij} y_j}{\sum_{i=1}^{N} w_i}$$

Currently most of the network effect models can only accommodate one network, for example Burt's model (1987), and Leenders' model (1997). However, an actor is very often under influence of multiple networks. For example, friends and colleagues. So if a research requires investigation of which effect out of multiple networks plays a more significant role in consumers' decision, none of these models could work, thus 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. While considerable work has been done on these models, the question of which network model explains diffusion has not been resolved.

Doreian (1989) introduced two regimes of network effects autocorrelation model. Such method allows us investigate effects of two network effect on consumers' choices. The network autocorrelation model takes both interdependence of actors and their attributes such as demographics into consideration. Such interdependence are described by a weight matrix. Doreian's model can capture both actor's intrinsic opinion and influence from alters in his social network. The model is described as below:

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

where $\mathbf{y}$ is the dependent variable, an integer variable; $\mathbf{X}$ is a vector of explanatory variables; $\mathbf{W}$s represent the social structures underlying each autoregressive regime; ρ_1 and ρ_2 are the parameters of two network effect respectively; $\boldsymbol{\epsilon}$ is normally distributed disturbance term.

However, all of these autocorrelation models mentioned above have continuous dependent variable. Fujimoto and Valente (2011) developed a plausible solution by using logistics regression to investigate network influence on whether an adult use alcohol or not. The model is shown below:

$$\text{logit}(\mathbf{y}) = \mathbf{X}\boldsymbol{\beta} + \rho \mathbf{W}\mathbf{y} + \boldsymbol{\epsilon}$$

$\mathbf{y}$ is the dichotomous dependent variable; $\mathbf{X}$ are independent variables and $\boldsymbol{\beta}$ are the correspondent parameters; $\mathbf{W}$ is the social network structure and $\boldsymbol{\rho}$ are the correspondent parameters; $\boldsymbol{\epsilon}$ is the error term. This kind of method is called as "quick and dirty" (QAD) by Doreian (1982). Although supporting binary dependent variable and multiple network, this model does not satisfy the assumption of logistics regression – the observations are not independent. So the estimation results are biased.

3 Method

Although there are some network effect models available, but there are still not many models with dichotomous dependent variable. One notable work is Yang and Allenby (2003)'s. They developed a hierarchical Bayesian autoregressive mixture model. Their model can only accommodate one network effect, although this single network effect could be compounded. Yang and Allenby defined each component network be associated with an explanatory variable, the sum of component coefficients is 1. Thus, they made an assumption that all component network coefficient must be at the same side, and statistically significant at the same time. Such assumptions does not hold if the effect of any of the component networks are statistically insignificant. We thus propose a variant auto-probit model that accommodates multiple regimes of network effects for the same group of actors. We then provide two solutions for our model. The first one is an E-M method, which is in the track of maximum likelihood, and the second one is a hierarchical Bayesian method. Detailed description of both estimations are shown in Appendix A and B.

3.1 Model Specification

Assume we observe the choice of actors y_i, $(y_i = \{0, 1\}, i = 1, ..., n)$ who are in multiple networks. The choice is dichotomous, y_i=0 or 1 and is driven by a latent variable z_i. The probability that an actor make the choice is:

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

where $\mathbf{z}$ is the latent preference of consumers. If it is larger than or equal to a threshold, 0, consumers would choose $\mathbf{y}$ as 1; if it is smaller than 0, then consumers would choose $\mathbf{y}$ as 0. $\mathbf{X}$ is an $n \times m$ covariate matrix, such as $[1\ \mathbf{X}_0]$. These covariates are the exogenous characteristics of consumers. $\boldsymbol{\beta}$ is a $m \times 1$ coefficient vector associated with $\mathbf{X}$. $\boldsymbol{\theta}$ is the autocorrelation term. $\boldsymbol{\theta}$ can be described as the aggregation of multiple network structure $\mathbf{W}_i$ and coefficient ρ_i where $i = 1, ..., k$. $\mathbf{W}$s are network structures describing connections among consumers. ρs are the coefficients of $\mathbf{W}$s describing effect size of such network. The error of the model is defined as two parts, $\boldsymbol{\epsilon}$ and $\mathbf{u}$, hence it is modeled as augmented error. $\boldsymbol{\epsilon}$ is the unobservable error term of $\mathbf{z}$ and $\mathbf{u}$ is the error term of $\boldsymbol{\theta}$. The benefit of such augmented model is that the latent error term $\mathbf{u}$ accounts for the nonzero covariances in the latent variable $\mathbf{z}$, if we marginalize on $\boldsymbol{\theta}$, all the unobserved interdependency will be isolated.

The augmented error results in latent preferences with nonzero covariance:

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

where $\mathbf{Q} = I_n + \sigma^2 \left(I_n - \sum_{i=1}^{k} \rho_i \mathbf{W}_i\right)^{-1} \left(\left(I_n - \sum_{i=1}^{k} \rho_i \mathbf{W}_i\right)^{-1}\right)^{\top}$. From the specification of Q we can

could sense computationally this is a significant problem.

The network structures describing the relationships among actors are represented by matrix $\mathbf{W}$. It is important that we develop models allowing multiple competitively network effects. So far all the models can only allow one network structure. Since a group of actors could have multiple relationships, and connections could be explained by different network theories. $\mathbf{W}$s could be defined on the base of relevant theories, for example, $\mathbf{W}_i$ describes cohesion and $\mathbf{W}_j$ describes structural equivalence; or defined by different network relationship, such as $\mathbf{W}_i$ describes friendship and $\mathbf{W}_j$ describes colleagueship. The coefficient ρ_i describe the effect size of correspondent matrix $\mathbf{W}_i$. By accommodating multiple networks in an auto-probit model we can compare the effects among competing network structures for the same group of actors embedded in social networks.

3.2 Expectation-Maximization Model

We first develop a model using MLE method. Since $\mathbf{z}$ is latent, we treat it as unobservable data. E-M algorithm is one of the most used methods to solve this kind of problem. Detailed description of our solution for k regimes of network effects is in Appendix A.

Our method consists of: first estimate the value for unobserved $\mathbf{z}$ given the current parameter set $\boldsymbol{\phi}$, ($\boldsymbol{\phi} = \{\boldsymbol{\beta}, \boldsymbol{\rho}, \sigma^2\}$). Then use it to complete data set $\{\mathbf{y}, \mathbf{X}, \mathbf{z}\}$. We then use this completed data to estimate a new $\boldsymbol{\phi}$ by maximizing the expectation of the likelihood of the complete data.

We first initialize the parameters.

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

where $i = 1, ..., m$, and $j = 1, ..., k$.

We then calculate the conditional expectation of parameters in the E-step.

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

where t is the number of step, $\mathbf{Q} = \text{Var}(\mathbf{z})$, $\mathbf{Q} = I_n + \sigma^2\left(I_n - \sum_{i=1}^{k}\rho_i\mathbf{W}_i\right)^{-1}\left(\left(I_n - \sum_{i=1}^{k}\rho_i\mathbf{W}_i\right)^{-1}\right)^{\top}$, and $\check{q}_{ij}$ is an element in the matrix $\mathbf{Q}^{-1}$.

In the M-step, we maximize $Q(\boldsymbol{\phi} \mid \boldsymbol{\phi}^{(t)})$ to get $\boldsymbol{\beta}^{t+1}$, $\boldsymbol{\beta}^{t+1}$ and $\Sigma^{(t+1)}$ ($\Sigma = \sigma^2$) for the next step.

$$\boldsymbol{\beta}^{(t+1)} = \arg\max_{\boldsymbol{\beta}} Q(\boldsymbol{\beta} \mid \boldsymbol{\beta}^{(t)})$$
$$\boldsymbol{\rho}^{(t+1)} = \arg\max_{\boldsymbol{\rho}} Q(\boldsymbol{\rho} \mid \boldsymbol{\rho}^{(t)})$$
$$\Sigma^{(t+1)} = \arg\max_{\Sigma} Q(\boldsymbol{\Sigma} \mid \boldsymbol{\Sigma}^{(t)})$$

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 E-M algorithm converge to the MLE estimates (Wu, 1983).

It is worth noting that the analytical solution for all the parameters is very complicated. For example

$$\boldsymbol{\beta}^{(t+1)} = \arg\max_{\boldsymbol{\beta}} Q(\boldsymbol{\phi} \mid \boldsymbol{\phi}^{(t)})$$
$$\frac{\partial \log Q(\boldsymbol{\phi} \mid \boldsymbol{\phi}^{(t)})}{\partial \boldsymbol{\beta}} = \frac{\partial}{\partial \boldsymbol{\beta}} \left(-\frac{1}{2} (\mathbf{z} - \mathbf{X}\boldsymbol{\beta})^\top \mathbf{Q}^{-1} (\mathbf{z} - \mathbf{X}\boldsymbol{\beta}) \right)$$
$$\hat{\boldsymbol{\beta}} = \left(\mathbf{X}^\top \mathbf{Q}^{-1} \mathbf{X} \right)^{-1} \mathbf{X}^\top \mathbf{Q}^{-1} \mathbf{R}$$

where $\mathbf{R} = \mathrm{E}(\mathbf{z})$.

Although the analytical solution of $\boldsymbol{\beta}$ can still be obtained, it is very complicated. And the situation gets even more complicated for the next few parameters. Consider $\boldsymbol{\rho}$:

$$\boldsymbol{\rho}^{(t+1)} = \arg\max_{\boldsymbol{\rho}} Q(\boldsymbol{\phi} \mid \boldsymbol{\phi}^{(t)})$$

Specify $\boldsymbol{\rho} = \{\rho_1, ..., \rho_k\}$, without losing any generalizability, let us take a look at the situation of ρ_1:

$$\rho_1^{(t+1)} = \arg\max_{\rho_1} Q(\boldsymbol{\phi} \mid \boldsymbol{\phi}^{(t)})$$
$$\frac{\partial \log Q(\boldsymbol{\phi} \mid \boldsymbol{\phi}^{(t)})}{\partial \rho_1} = \frac{\partial}{\partial \rho_1} \left(-\frac{1}{2} \log \mid \mathbf{Q} \mid -\frac{1}{2} (\mathbf{z} - \mathbf{X}\boldsymbol{\beta})^\top \mathbf{Q}^{-1} (\mathbf{z} - \mathbf{X}\boldsymbol{\beta}) \right)$$
$$\frac{\partial}{\partial \rho_1} (\mathbf{z} - \mathbf{X}\boldsymbol{\beta})^\top \mathbf{Q}^{-1} (\mathbf{z} - \mathbf{X}\boldsymbol{\beta}) = \frac{\partial}{\partial \rho_1} (\mathbf{z}^\top \mathbf{Q}^{-1} \mathbf{z} - \mathbf{z}^\top \mathbf{Q}^{-1} \mathbf{X}\boldsymbol{\beta} - \boldsymbol{\beta}^\top \mathbf{X}^\top \mathbf{Q}^{-1} \mathbf{z} + \boldsymbol{\beta}^\top \mathbf{X}^\top \mathbf{Q}^{-1} \mathbf{X}\boldsymbol{\beta})$$

Establish an analytical solution for this parameter is possible only for very simple specification. Given multiple *rho*s, an analytical solution is impossible. So we have to use numerical method to solve it.

We finally come to σ^2. Let $\sigma^2 = \Sigma$

$$\Sigma^{(t+1)} = \arg\max_{\Sigma} Q(\boldsymbol{\phi} \mid \boldsymbol{\phi}^{(t)})$$
$$\frac{\partial \log L}{\partial \Sigma} = \frac{\partial}{\partial \Sigma} \left(-\frac{1}{2} \log \mid \mathbf{Q} \mid -\frac{1}{2} (\mathbf{z} - \mathbf{X}\boldsymbol{\beta})^\top \mathbf{Q}^{-1} (\mathbf{z} - \mathbf{X}\boldsymbol{\beta}) \right) \tag{1}$$

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

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

The second term is:

$$\begin{aligned} &\frac{\partial}{\partial \Sigma} (\mathbf{z} - \mathbf{X}\boldsymbol{\beta})^{\top} \mathbf{Q}^{-1} (\mathbf{z} - \mathbf{X}\boldsymbol{\beta}) \\ &= \frac{\partial}{\partial \Sigma} (\mathbf{z} - \mathbf{X}\boldsymbol{\beta})^{\top} \left(I_n + \Sigma \left(I_n - \sum_{i=1}^{k} \rho_i W_i \right)^{-1} \left(\left(I_n - \sum_{i=1}^{k} \rho_i W_i \right)^{-1} \right)^{\top} \right)^{-1} (\mathbf{z} - \mathbf{X}\boldsymbol{\beta}) \end{aligned}$$

This is again not solvable analytically, and numerical method are needed to get the estimators for all parameters. Although could be solved by numerical method, E-M still cannot be applied in this situation. This is because the mode of σ^2, the error term of the autocorrelation term θ, is at 0 (see Figure 1), so the estimated value of it by maximum likelihood is at 0, and makes the variance-covariance matrix singular. Thus we have to find another solution.

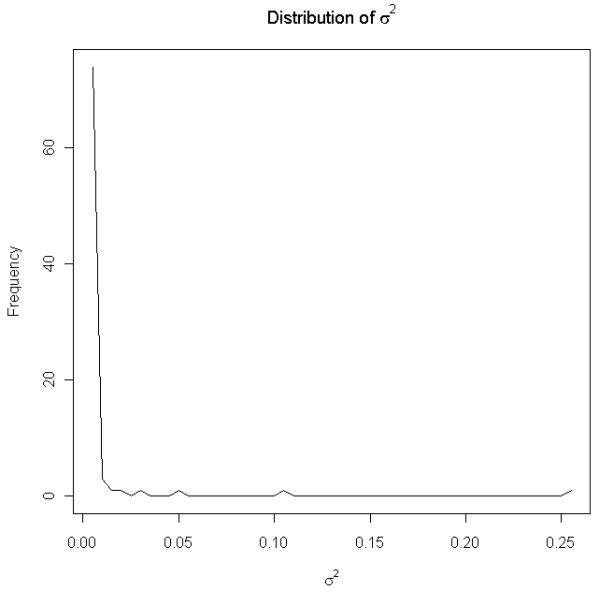


Figure 1: Distribution of σ^2, variance of $\boldsymbol{\theta}$, estimated by E-M solution

3.3 Hierarchical Bayesian Solution

We then turn to Bayesian methods. Since the observed choice of consumer's is decided by his/her unobserved preference, such problem has a hierarchical structure, so it is natural to think of using hierarchical Bayesian method. The model specification is the same as for the E-M solution. $\mathbf{y}$ is

the observed dichotomous variable and modeled by probit function, $\mathbf{z}$ is the latent variable. The estimation (MCMC method) is done by sequentially generating draw from a series of distributions. MCMC requires specification of prior distributions for the model parameters and derivation of the full conditional distribution of parameters. The full conditional distributions of all the parameters we need to estimate are presented in the Rotational Conditional Maximization and/or Sampling (RCMS) table (Thomas, 2009) below. The prior distributions of the model parameters are specified as follows. We

Table 1: RCMS table for hierarchical Bayesian solution

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

generally adopted the priors proposed by Smith and LeSage (2004).

$$\begin{aligned}\boldsymbol{\beta} &\sim \text{Normal}(\boldsymbol{\nu}_\beta, \boldsymbol{\Omega}_\beta)\\ \sigma^2 &\sim \text{Gamma}(a, b)\\ \boldsymbol{\rho} &\sim \text{Normal}(\boldsymbol{\nu}_\rho, \boldsymbol{\Omega}_\rho)\end{aligned}$$

$\boldsymbol{\beta}$ follows normal distribution with mean $\boldsymbol{\nu}_\beta$ and variance $\boldsymbol{\Omega}_\beta$. σ^2 follows inverted gamma distribution, where parameters a and b. ρ follows a normal distribution. We then use Markov chain Monte Carlo (MCMC) to generate draws of conditional posterior distributions for the parameters. Detailed description of the implementation of my method, including the conditional distribution of all parameters, is given in B.

3.4 Validation of Bayesian Software

One challenge of Bayesian method is getting an error-free solution. Usually Bayesian solution has high complexityh, and lacking of software causes many researchers to develop their own, hence increases the chance of error. Unfortunately most of the 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. Compared with the available software of Bayesian method, its validation does not have many literature available. We use posterior quantiles method (Cook et al., 2006) to validate our software. The basic idea is, we draw parameter θ from its prior distribution $p(\Theta)$; then generate data distribution $p(y \mid \theta)$. If the software is correctly coded, the 95% credible interval should contain the true parameter with probability 95%. The details are described as follow.

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.

Define the quantile as:

$$q(\theta_0) = P(\Theta < \theta_0)$$

where θ_0 is the true value drawn from prior distribution; Θ is a series of draw from posterior distribution. and the estimated quantile is:

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

where θ is a series of draw from posterior distribution generated by the software to-be-tested; N is the number of draws in MCMC. The quantile is the probability of 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 approaches Uniform$(0, 1)$, when $N \to \infty$ (Cook et al., 2006). The whole process up to now is defined as one replication. If run a number of replications, we expect to observe a uniformly distribution $\hat{q}(\theta_0)$ around θ_0, meaning posterior should be randomly distributed around the true value..

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

$$\begin{aligned}\mathbf{z} &= \mathbf{X}_1\beta_1 + \mathbf{X}_2\beta_2 + \boldsymbol{\theta} + \boldsymbol{\epsilon} \\ \boldsymbol{\theta} &= \rho_1\mathbf{W}_1\boldsymbol{\theta} + \rho_2\mathbf{W}_2\boldsymbol{\theta} + \mathbf{u}\end{aligned} \tag{2}$$

The data is generated using normal distribution:

$$\mathbf{X} \sim \text{Normal}(1, 4)$$

We then specified a conjugate prior distribution for each parameter, and use Metropolis-Hasting sampling to simulate the posterior distributions.

$$\begin{aligned}\boldsymbol{\beta} &\sim \text{Normal}(0, 1) \\ \sigma^2 &\sim \text{InvGamma}(5, 10) \\ \boldsymbol{\rho} &\sim \text{Normal}(0.05, 0.05^2)\end{aligned}$$

We performed a simulation of 10 replication to validate our hierarchical Bayesian MCMC software. The generated sample size for $\mathbf{X}$ is 50, so the size of the network structure $\mathbf{W}$ is 50 by 50. In each replication we generated 20000 draws from the posterior distribution of all the parameters in $\boldsymbol{\phi}$ ($\boldsymbol{\phi} = \{\beta_1, \beta_2, \rho_1, \rho_2, \sigma^2\}$), and kept one from every 20 draws. So finally we have 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 2. The X-axis is the total replications of the five parameters – 50. The Y-axis is the number of draws larger than true parameters in each replication. The 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 confirmed that our Bayesian software is correctly written.

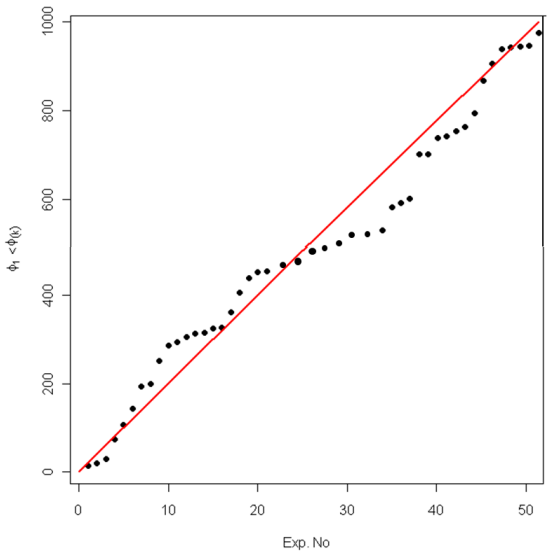


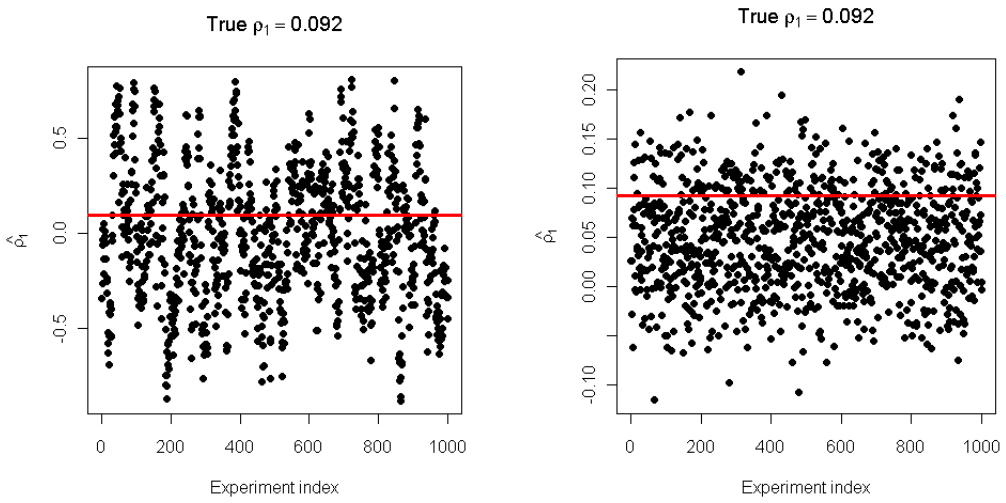
Figure 2: Distribution of sorted quantiles of parameters, $\beta_1, \beta_2, \rho_1, \rho_2, \sigma^2$, 10 replications of posterior quatiles experiments

4 Experiments

Before use real data to validate the accuracy of parameter estimates of our Bayesian solution, we want to make sure our solution generates robust estimation from the posterior distribution, thus we want to diagnose our MCMC. We first study the sensitivity of prior distribution of parameters, because the solution will not have a proper posterior distribution without a proper distribution even the code is correctly written. We first choose a flat prior distribution for ρ, $\rho \sim \text{Normal}(0, 100)$. As shown in Figure 3(a), the posterior draws of ρ have strong autocorrelation. We then choose a narrow prior distribution for ρ, $\rho \sim \text{Normal}(0.05, 0.05^2)$, the posterior draws for ρ are shown in Figure 3(b), and we could find the draws tend to be random. So posterior distribution of ρ is sensitive to its prior distribution. In order to generate random estimates, we choose $\text{Normal}(0.05, 0.05^2)$ as the prior distribution for ρ.

Second, since ρs are generated by using sequential draws, there is autocorrelations exist between consecutive draws, sometimes the autocorrelation is still high even the lag between two draws is large. The autocorrelation plot for ρ_1 (Figure 4) shows two ρ_1 draws have strong correlation even the lag is larger than 300, so thinning for the draws is necessary.

We use Yang and Allenby (2003)'s Japanese car data to validate the accuracy of parameter estimates of our Bayesian solution. Such data consists of 857 actors' midsize car purchase information. The dependent variable is whether an actor purchased a Japanese or not, where 1 stands for purchased and 0 otherwise. All the car models in the data are substitutable and roughly have similar prices. Researchers are interested in whether the preferences of Japanese car among actors are interdependent



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

Figure 3: Prior sensitivity for parameter ρ, hierarchical Bayesian solution

or not. The interdependence in the network are measured by geographical location:

$$W_{ij} = \begin{cases} 1, & \text{if } i \text{ and } j \text{ have the same zip code;} \\ 0, & \text{otherwise.} \end{cases}$$

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.

The comparison of the coefficient estimates from Yang and Allenby's code and our Bayesian solution is shown in Figure 5. In order to make a peer comparison, we set all the network effects except the first one as $\mathbf{0}_{n,n}$ matrix. Our $\mathbf{W}_1$ has the same definition as Yang and Allenby's $\mathbf{W}$. For the third method, we add one more network structure $\mathbf{W}_2$, the structure equivalence of two consumers. We use Euclidean distance to measure structural equivalence. In a directed network with non-weighted edges the Euclidean distance between two nodes i and j is the sum of squared common neighbors between the nodes that i and j connect to respectively, and from all nodes to i and j respectively. The distance is shown in the equation (3).

$$d_{ij} = \sqrt{\sum_{k=1, k\neq i,j}^{N} (A_{ik} - A_{jk})^2} \tag{3}$$

where

$$A_{ik} = \begin{cases} 1 & \text{if node } i \text{ and } k \text{ are neighbors} \\ 0 & \text{otherwise} \end{cases}$$

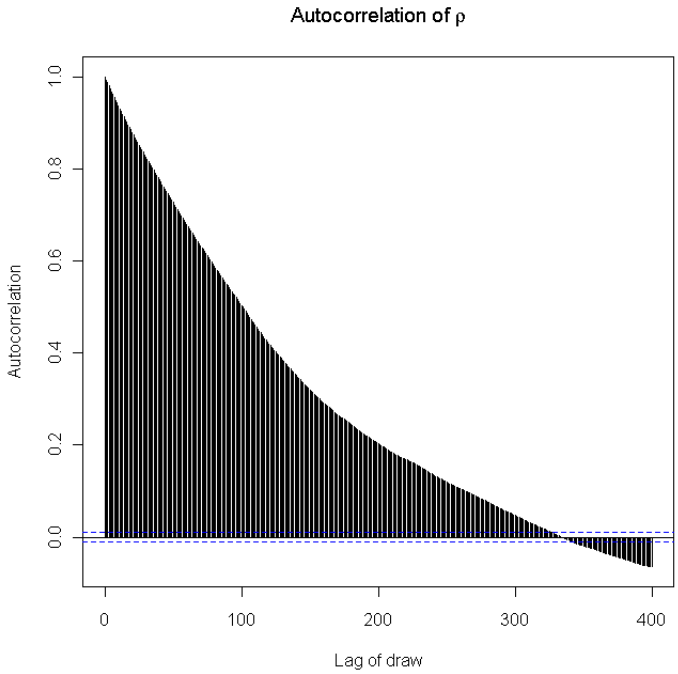


Figure 4: Autocorrelation plot of ρ

The larger d between node i and j, the less structurally equivalent they are. We get the inverse of d_{ij} plus one in order to construct a measure with a positive relationship with role equivalence:

$$s_{ij} = \frac{1}{d_{ij} + 1}$$

The comparison is shown in Figure 5. Each box contains the estimates of one parameter from three methods. The left one is from Yang and Allenby's, the middle one is from NAP with 1 network, and the right one is from NAP with 2 networks. All the coefficient estimates, $\hat{\boldsymbol{\beta}}_i$, $\hat{\boldsymbol{\rho}}_2$, and $\hat{\sigma}^2$ of the three methods have similar mean, standard deviation and credible interval. Such results confirm again that NAP returns correct estimates of parameters in the model. One thing interesting here is the effect size of the second network, structural equivalence, has a significant negative effect. Which suggests a diminishing cluster effect, when the number of people in the cluster gets bigger, the influence is not proportionally bigger. When the the structural equivalence between two customers is large, meaning they are in the same community (zip code), and the size, i.e. number of customers, of such community is large, so they have more common neighbors, thus more same scalar component in the vector.

5 Conclusion

We introduced an auto-probit model to study binary choice of a group of actors that have multiple network relationships among them. We specified the model in both E-M and hierarchical Bayesian

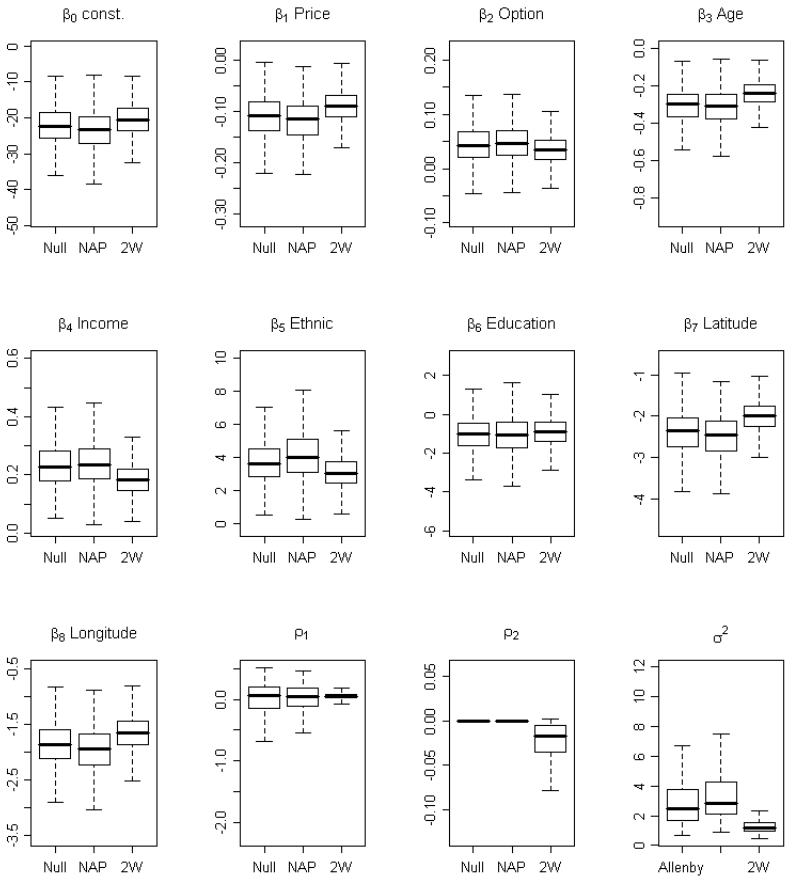


Figure 5: Coefficient estimates comparison

methods, and developed estimation solutions for both of them. We found E-M solution cannot estimate the parameters thus only hierarchical Bayesian solution can be used here. We also validated our Bayesian solution by using posterior quantiles methods 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 effect, cohesion and structural equivalence, by using real data. Experiments showed all three returned same estimates, thus confirmed our softweare return correct parameter estimates. Our future plan includes first, run our software on more benchmark data with better defined network structure. We also want to run more experiments with simulated populations to evaluate the properties of the solution. For example, let $\mathbf{W}$ have different features, such as network with randomly distributed edges, clustered edges, and skewed distributed edges etc. Second, assume $\mathbf{W}\boldsymbol{\theta}$ has strong effect, we will vary ρ's true value from small number to large number, and observe whether our solution can capture the variation. Third, we want to compare our program with QAD, because although people know parameter estimates returned by QAP is biased, we do not know how different they are from the true value. Finally we also want to study how multicollinearities between $\mathbf{X}$s, and between $\mathbf{X}$ and $\mathbf{W}\boldsymbol{\theta}$ affect estimated results.

Acknowledgement

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

A E-M solution implementation

A.1 Deduction

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

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

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

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

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

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

$$\begin{aligned} p(\mathbf{y}|\mathbf{z}) p(\mathbf{z}|\boldsymbol{\beta}, \boldsymbol{\rho}, \sigma^2) &= p(\mathbf{y}, \mathbf{z}|\boldsymbol{\beta}, \boldsymbol{\rho}, \sigma^2) \\ &= p(\mathbf{z}|\mathbf{y}; \boldsymbol{\beta}, \boldsymbol{\rho}, \sigma^2) p(\mathbf{y}) \end{aligned} \tag{4}$$

The right side of equation (4) are two distributions we already have, as shown below. Please be aware, although $\mathbf{z}$ follow normal distribution, $\mathbf{z}|\mathbf{y}$ follows truncated normal distribution.

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

We now show how the algorithm works by using a simple example. Assume we only consider parameter $\boldsymbol{\beta}$

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

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

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

Then we include more parameters, $\boldsymbol{\rho}$ and σ^2, and Var($\mathbf{z}$) = $\mathbf{Q}$:

$$
\begin{aligned}
\mathrm{E}[\mathbf{z}]^{(t+1)} &= \mathrm{E}[\mathbf{z}|\mathbf{y},\boldsymbol{\beta}^{(t)}] = f(\boldsymbol{\beta}^{(t)},\mathbf{y}) \\
\log L(\boldsymbol{\beta},\boldsymbol{\rho},\sigma^2|\mathbf{z}) &= \log p(\mathbf{z}|\boldsymbol{\beta},\boldsymbol{\rho},\sigma^2) \\
&= \log\prod_{i=1}^{n} p(z_i|\boldsymbol{\beta},\boldsymbol{\rho},\sigma^2) \\
&= \sum_{i=1}^{n}\log\frac{1}{\sqrt{2\pi|\mathbf{Q}|}} - \frac{1}{2}(\mathbf{z}-\mathbf{X}\boldsymbol{\beta})^\top\mathbf{Q}^{-1}(\mathbf{z}-\mathbf{X}\boldsymbol{\beta}) \\
&= \sum_{i=1}^{n}\log\frac{1}{\sqrt{2\pi|\mathbf{Q}|}} - \left(\frac{1}{2}\mathbf{z}^\top\mathbf{Q}^{-1}\mathbf{z} - \mathbf{z}^\top\mathbf{Q}^{-1}\mathbf{X}\boldsymbol{\beta} - \mathbf{X}^\top\boldsymbol{\beta}\mathbf{Q}^{-1}\mathbf{z} + \mathbf{X}^\top\boldsymbol{\beta}\mathbf{Q}^{-1}\mathbf{X}\boldsymbol{\beta}\right) \qquad (5)
\end{aligned}
$$

If we decompose the matrices above as vector product, then:

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

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

$$\mathrm{E}[\mathbf{z}|\boldsymbol{\theta}, \mathbf{y}] = \int \mathbf{z} p(\mathbf{z}) d\mathbf{z} = \mathbf{R}$$

A.2 Expectation step

E-M algorithm consists of two steps, the first one is expectation.

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

where $\boldsymbol{\phi}$ is the parameter set, and t is the number of steps.

A.3 Maximization step

The second step of E-M algorithm is maximization.

$$\begin{aligned}
\boldsymbol{\beta}^{(t+1)} &= \arg\max_{\boldsymbol{\beta}} Q(\boldsymbol{\phi}|\boldsymbol{\phi}^{(t)}) \\
&= \arg\max_{\boldsymbol{\beta}} \sum_{i=1}^{n} \log \frac{1}{\sqrt{2\pi|\mathbf{Q}|}} - \frac{1}{2}(\mathbf{z} - \mathbf{X}\boldsymbol{\beta})^\top \mathbf{Q}^{-1}(\mathbf{z} - \mathbf{X}\boldsymbol{\beta}) \qquad (6)
\end{aligned}$$

If we directly use analytical method to solve the Equation (6) above, then:

$$\begin{aligned}
\frac{\partial \log L}{\partial \boldsymbol{\beta}} &= \frac{\partial}{\partial \boldsymbol{\beta}}\left(-\frac{1}{2}(\mathbf{z} - \mathbf{X}\boldsymbol{\beta})^\top \mathbf{Q}^{-1}(\mathbf{z} - \mathbf{X}\boldsymbol{\beta})\right) \\
\frac{\partial}{\partial \boldsymbol{\beta}}(\mathbf{z} - \mathbf{X}\boldsymbol{\beta})^\top \mathbf{Q}^{-1}(\mathbf{z} - \mathbf{X}\boldsymbol{\beta}) &= \frac{\partial}{\partial \boldsymbol{\beta}}(\mathbf{z}^\top \mathbf{Q}^{-1}\mathbf{z} - \mathbf{z}^\top \mathbf{Q}^{-1}\mathbf{X}\boldsymbol{\beta} - \boldsymbol{\beta}^\top \mathbf{X}^\top \mathbf{Q}^{-1}\mathbf{z} + \boldsymbol{\beta}^\top \mathbf{X}^\top \mathbf{Q}^{-1}\mathbf{X}\boldsymbol{\beta}) \\
&= -\mathbf{z}^\top \mathbf{Q}^{-1}\mathbf{X} - \mathbf{X}^\top \mathbf{Q}^{-1}\mathbf{z} + \mathbf{X}^\top \mathbf{Q}^{-1}\mathbf{X}\boldsymbol{\beta} \qquad (7)
\end{aligned}$$

Set Equation (7) as 0, then:

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

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

Assume $\boldsymbol{\rho} = \{\rho_1, ..., \rho_k\}$, without losing any generalizabiliy, let us take a look at the situation of ρ_1:

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

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

Impossible to get the analytical solution for ρ_i.

Let $\sigma^2 = \Sigma$

$$\begin{aligned}
\Sigma^{(t+1)} &= \arg\max_{\Sigma} Q(\boldsymbol{\phi}|\boldsymbol{\phi}^{(t)}) \\
\frac{\partial \log L}{\partial \Sigma} &= \frac{\partial}{\partial \Sigma}\left(-\frac{1}{2}\log|\mathbf{Q}| - \frac{1}{2}(\mathbf{z}-\mathbf{X}\boldsymbol{\beta})^\top \mathbf{Q}^{-1}(\mathbf{z}-\mathbf{X}\boldsymbol{\beta})\right) \qquad (8)
\end{aligned}$$

The first term at the the right hand side of equation above is:

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

The second term is:

$$\begin{aligned}
&\frac{\partial}{\partial \Sigma}(\mathbf{z}-\mathbf{X}\boldsymbol{\beta})^\top \mathbf{Q}^{-1}(\mathbf{z}-\mathbf{X}\boldsymbol{\beta}) \\
&= \frac{\partial}{\partial \Sigma}(\mathbf{z}-\mathbf{X}\boldsymbol{\beta})^\top\left(I_n + \Sigma\left(I_n - \sum_{i=1}^{k}\rho_i W_i\right)^{-1}\left(\left(I_n - \sum_{i=1}^{k}\rho_i W_i\right)^{-1}\right)^\top\right)^{-1}(\mathbf{z}-\mathbf{X}\boldsymbol{\beta})
\end{aligned}$$

This is again not solvable by using analytical method.

B Markov chain Monte Carlo estimation

The Markov chain Monte Carlo method generate chain of draws from the conditional posterior distributions of parameters. Our solution consists of steps as follows.

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

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

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 $\boldsymbol{\beta}$, $\boldsymbol{\beta} \sim \text{Normal}(\boldsymbol{\nu}_\beta, \boldsymbol{\Omega}_\beta)$

1. define $\boldsymbol{\beta}_0$, where

$$\boldsymbol{\beta}_0 = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

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

$$\mathbf{D}^{-1} = \begin{bmatrix} \frac{1}{400} & 0 & \dots & 0 \\ 0 & \frac{1}{400} & \dots & 0 \\ \vdots & \vdots & \dots & \vdots \\ 0 & 0 & \dots & \frac{1}{400} \end{bmatrix} = \begin{bmatrix} \sigma_0^2 & 0 & \dots & 0 \\ 0 & \sigma_0^2 & \dots & 0 \\ \vdots & \vdots & \dots & \vdots \\ 0 & 0 & \dots & \sigma_0^2 \end{bmatrix}$$

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

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

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

$\therefore \boldsymbol{\beta} \sim \text{Normal}\left(\mathbf{X}^{-1}(\mathbf{z} - \boldsymbol{\theta}),\ (\mathbf{X}^\top \mathbf{X})^{-1}\right)$
We need to have an offset for the initial variance, so $\boldsymbol{\Omega}_\beta = \left(\mathbf{D}^{-1} + \mathbf{X}^\top \mathbf{X}\right)^{-1}$

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

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

1. First, define $\mathbf{B} = I_n - \rho_1 \mathbf{W}_1 - \rho_2 \mathbf{W}_2$

$$\begin{aligned} \boldsymbol{\theta} &= \rho_1 \mathbf{W}_1 \boldsymbol{\theta} + \rho_2 \mathbf{W}_2 \boldsymbol{\theta} + \mathbf{u} \\ \boldsymbol{\theta} - \rho_1 \mathbf{W}_1 \boldsymbol{\theta} - \rho_2 \mathbf{W}_2 \boldsymbol{\theta} &= \mathbf{u} \\ (I_n - \rho_1 \mathbf{W}_1 - \rho_2 \mathbf{W}_2)\boldsymbol{\theta} &= \mathbf{u} \\ \mathbf{B}\boldsymbol{\theta} &= \mathbf{u} \end{aligned}$$

2. Then $\mathbf{\Omega}_\theta = \left(I_n + \dfrac{\mathbf{B}^\top\mathbf{B}}{\sigma^2}\right)^{-1}$

$$\begin{aligned}\mathbf{B}\boldsymbol{\theta} &= \mathbf{u}\\ \boldsymbol{\theta} &= \mathbf{B}^{-1}\mathbf{u}\end{aligned}$$

Since $\text{Var}(\mathbf{u}) = \sigma^2 I_n$

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

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

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

$$\begin{aligned}\mathbf{z} &= \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\theta} + \boldsymbol{\epsilon}\\ \mathbf{z} &= \mathbf{X}\boldsymbol{\beta} + (I_n - \rho_1\mathbf{W}_1 - \rho_2\mathbf{W}_2)^{-1}\mathbf{u} + \boldsymbol{\epsilon}\\ \boldsymbol{\theta} &= (\mathbf{z} - \mathbf{X}\boldsymbol{\beta}) - \boldsymbol{\epsilon}7\end{aligned}$$

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

$$\begin{aligned}a &= s_0 + \frac{n}{2}\\ b &= \frac{2}{\boldsymbol{\theta}^\top\mathbf{B}^\top\mathbf{B}\boldsymbol{\theta} + \dfrac{2}{q_0}}\end{aligned}$$

where $s_0 = 5$, n is the size of data.

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

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

where $\Delta_i \sim \text{Normal}(0, 0.01)$.

The accepting probability α is obtained by:

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

C Solution diagnose

We run MCMC experiment to confirm there is no autocorrelation among draws of each parameter. In this experiment, we set the length of 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 for the 1000 draws after burn-in and thinning are listed in the Figure 6 below.

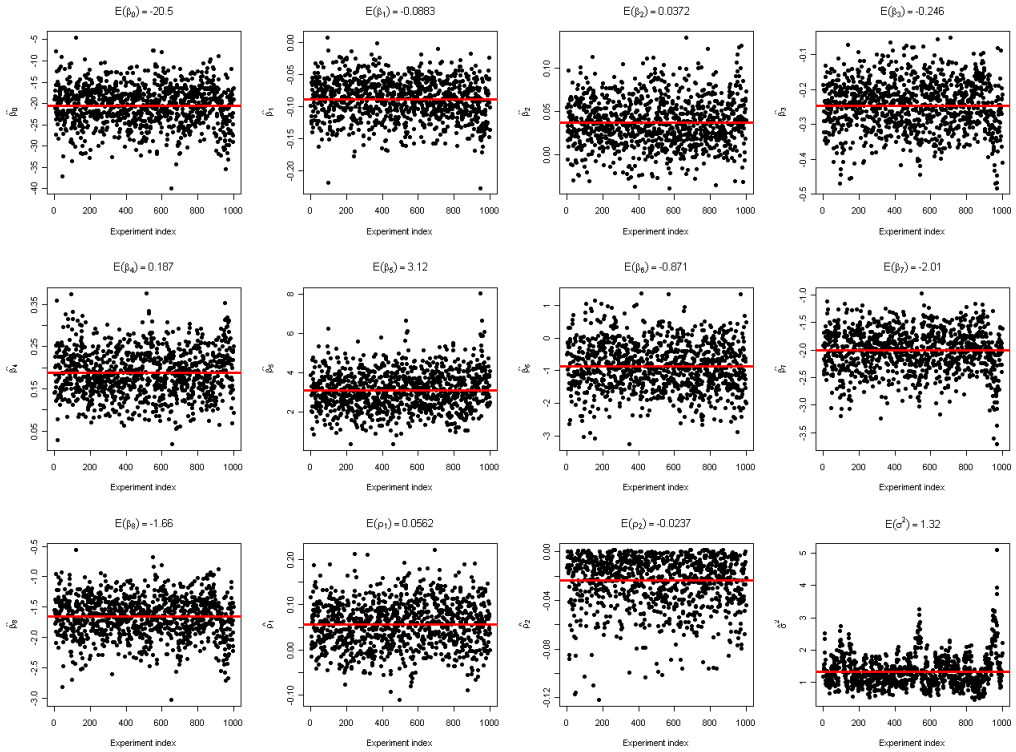


Figure 6: Trace plot of a two-network auto-probit model

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 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 plot below 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 red area is otherwise. If 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 correlation between they (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 non-correlation.

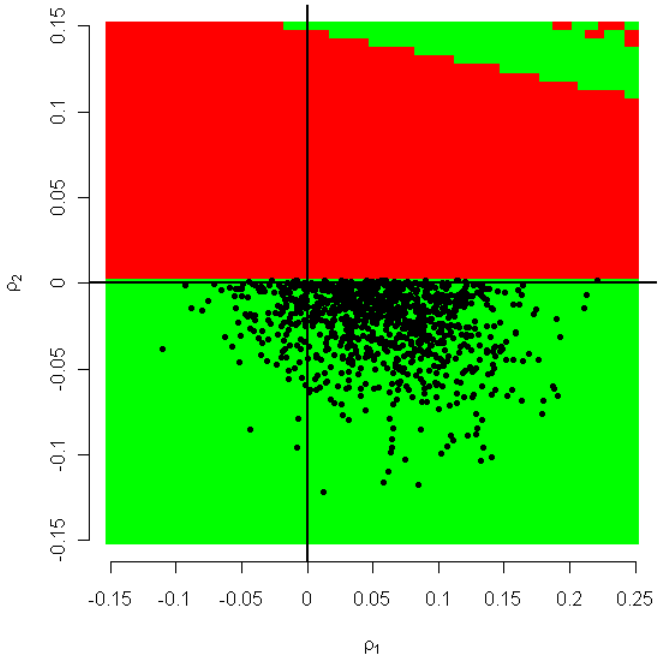


Figure 7: Scatter plot of ρ_1 and ρ_2 on valid region for invertible $\mathbf{B}$,

References

Allenby, G. M. and Rossi, P. E. (1998). Marketing models of consumer heterogeneity. *Journal of Econometrics*, 89(1-2):57–78.

Anselin, L. (1988). Spatial econometrics: Methods and models. Studies in Operational Regional Science. Springer, 1st edition.

Bass, F. M. (1969). A new product growth for model consumer durables. *Management Science*, 15(5):215–227.

Bernheim, B. D. (1994). A theory of conformity. *Journal of Political Economy*, 102(5):841–77.

Burt, R. S. (1987). Social contagion and innovation: Cohesion versus structural equivalence. *American Journal of Sociology*, 92(6):1287.

Case, A. C. (1991). Spatial patterns in household demand. *Econometrica*, 59(4):953–965.

Cook, S. R., Gelman, A., and Rubin, D. B. (2006). Validation of software for bayesian models using posterior quantiles. *Journal of Computational and Graphical Statistics*, 15:675–692.

Cressie, N. A. C. (1993). Statistics for spatial data. Probability and Statistics series. Wiley-Interscience, revised edition.

Doreian, P. (1980). Linear models with spatially distributed data: Spatial disturbances or spatial effects. *Sociological Methods and Research*, 9(1):29–60.

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

Doreian, P. (1989). *Two Regimes of Network Effects Autocorrelation.* The Small World. Ablex Publishing.

Duncan, O. D., Haller, A. O., and Portes, A. (1968). Peer influences on aspirations: A reinterpretation. *The American Journal of Sociology*, 74(2):119–137.

Fujimoto, K. and Valente, Thomas, W. (2011). Network influence on adolescent alcohol use: Relational, positional, and affiliation-based peer influence.

Kamakura, W. A. and Russell, G. J. (1989). A probabilistic choice model for market segmentation and elasticity structure. *Journal of Marketing Research*, 26(4):379–390.

Ord, K. (1975). Estimation methods for models of spatial interaction. *Journal of the American Statistical Association*, 70(349):120–126.

Rogers, E. M. (1962). *Diffusion of Innovations.* Free Press.

Smirnov, O. A. (2005). Computation of the information matrix for models with spatial interaction on a lattice. *Journal of Computational and Graphical Statistics*, 14(4):910–927.

Smith, T. E. and LeSage, J. P. (2004). A bayesian probit model with spatial dependencies. In Pace, K. R. and LeSage, J. P., editors, *Advances in Econometrics: Volume 18: Spatial and Spatiotemporal Econometrics*, pages 127–160. Elsevier.

Thomas, A. C. (2009). *Hierarchical Models for Relational Data*. Phd dissertation, Harvard University, Cambridge, MA.

Valente, T. W. (2005). Models and methods for innovation diffusion. In Carrington, P. J., Scott, J., and Wasserman, S., editors, *Models and Methods in Social Network Analysis*. Cambridge University Press.

Wu, C. F. J. (1983). On the convergence properties of the em algorithm. *The Annals of Statistics*, 11(1):95–103.

Yang, S. and Allenby, G. M. (2003). Modeling interdependent consumer preferences. *Journal of Marketing Research*, XL:282–294.