Multiple-Regime Binary Autocorrelation Models for Social Networks

Bin Zhang
Heinz College, iLab
Carnegie Mellon University

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

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

David Krackhardt
Heinz College, iLab
Carnegie Mellon University

Ramayya Krishnan
Heinz College, iLab
Carnegie Mellon University

August 25, 2011

Abstract

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. Another is that these decisions are likely to be binary in nature and more difficult to model with conventional approaches, both conceptually and computationally. To that end, we present a hierarchical, multiple network-regime auto-probit model (m-NAP) for this class of data and propose two algorithms for fitting it, based on Expectation-Maximization (E-M) and Markov Chain Monte Carlo (MCMC). We investigate the behaviors of the parameter estimates on various sensitivity conditions, such as the impact of the prior distribution and the nature of the structure of the network, and demonstrate on several examples of correlated binary data in networks.

1 Introduction

The prevalence and widespread adoption of online social networks have made the analysis of social networks, and behaviors of individuals embedded in these networks, an important topic of study in information systems (Brancheau and Wetherbe, 1990; Chatterjee and Eliashberg, 1990; Premkumar and Nilakanta, 1994; Agarwal et al., 2008; Oinas-Kukkonen et al., 2010). While past investigations into behaviour 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, potentially billions of nodes and trillions or ties. Analyzing the behavior of these individuals, such as their purchasing or technology adoption tendencies, requires 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 individuals choose to adopt a product or technology adoption based solely on their own attributes, such as age, education, and income (Kamakura and Russell, 1989; Allenby and Rossi, 1998), 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 the tendency for two people to form a friendship and for each of them to exhibit this adoption behavior (**??**); either social property will increase the ability to predict a person's adoption behavior beyond their observed characteristics.

Either of these explanations would produce outcomes that, when viewed statically, 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; 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 model for binary outcomes beginning with the probit framework, which allows 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 first use the Expectation-Maximization algorithm (EM) to find a maximum likelihood estimator for the model parameters, then use Markov Chain Monte Carlo, 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 distribution. Preliminary experiments show that the E-M solution to this model is degenerate, and cannot produce a usable variance-covariance matrix for parameter estimates, and so the MCMC method is preferred. Our software is also validated by using the posterior quantiles method (Cook et al., 2006). We ensure that the parameter estimates from the model are correct by testing first on simulated data, before moving on to real examples of network-correlated behavior.

The rest of the paper is organized as follows. We discuss the literature on the network effects model in Section 2. Our two estimation algorithms for the multi-network autoprobit, 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 paper in Section 5.

2 Literature

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, which is 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 where the outcomes of neighboring individuals are explicitly linked, such as the simultaneous autoregressive model (SAR). The general methods of SAR are described in Anselin (1988) and Cressie (1993); we consider simultaneous autoregression on the residuals, of the form

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

where $\mathbf{y}$ is a vector of observed outcomes, in this case consumer choice; $\mathbf{X}$ is a vector of explanatory variables; and the initial error term ϵ_i follows a normal distribution. Some well accepted MLE solutions are provided by Ord (1975), Doreian (1980, 1982), and Smirnov (2005).

Most of the current 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, such as that of friends and that of colleagues. So if a research requires investigation of which effect out of multiple networks plays the most significant role in consumers' decision, 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. 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 two regimes of network effects autocorrelation model for continuous outcomes. Such a method allows us to investigate effects of two network effects on consumers' choices, so long as these choices reflect the type of data required. The network autocorrelation model takes both interdependence of actors and their attributes such as demographics into consideration; these interdependencies are each described by a weight matrix W_i. 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; $\mathbf{X}$ is a vector of explanatory variables; $\mathbf{W}$s represent the social structures underlying each autoregressive regime.

As this model can only have a continuous dependent variable, Fujimoto and Valente (2011) developed a plausible solution for binary outcomes by directly inserting an autocorrelation term $\mathbf{Wy}$ into the right hand side of a logistic regression:

$$\begin{aligned} y_i &\sim \text{Be}(p_i) \\ \log(\frac{p_i}{1-p_i}) &= \mathbf{X}\boldsymbol{\beta} + \rho \sum_j \mathbf{W}_{ij}\mathbf{y}_j \end{aligned}$$

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.

Yang and Allenby (2003) developed a hierarchical Bayesian autoregressive mixture model to analyze the effect of multiple network effects 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, and also be statistically significant or insignificant together. Such assumptions do not hold if the effect of any but not all of the component networks is statistically insignificant.

3 Method

We propose a variant of the auto-probit model that accommodates multiple regimes of network effects 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 one is a Markov Chain Monte Carlo routine that treats the model as Bayesian. Detailed descriptions of both estimations are shown in Appendix A and B.

3.1 Model Specification

The actors are assumed to have different types of network connections between them, where W_i is the i^{th} network in question. $\mathbf{y}$ is the vector of observed binary choices, and is an indicator function of the latent preference of consumers $\mathbf{z}$. If $\mathbf{z}$ is larger than a threshold 0, consumers choose $\mathbf{y}$ as 1; if $\mathbf{z}$ is smaller than 0, then consumers would choose $\mathbf{y}$ as 0.

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

$\mathbf{z}$ could be represented as a function of both exogenous covariates $\mathbf{X}$ and autocorrelation term $\boldsymbol{\theta}$. $\mathbf{X}$ is an $n \times m$ covariate matrix, such as $[1\ \mathbf{X}_0]$. These covariates could be the exogenous characteristics of consumers. $\boldsymbol{\beta}$ is a $m \times 1$ coefficient vector associated with $\mathbf{X}$. $\boldsymbol{\theta}$ is the autocorrelation term, which is responsible for those nonzero covariances in the $\mathbf{z}$. $\boldsymbol{\theta}$ 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 and relationships among consumers. Our model allows multiple competing network effects $\mathbf{W}$. Each $\mathbf{W}_i$ could be defined on the base of relevant theories; for example, $\mathbf{W}_1$ describes homophily, $\mathbf{W}_2$ describes social influence and $\mathbf{W}_3$ describes structural equivalence; or defined by different network relationship, such as $\mathbf{W}_1$ describes friendship, $\mathbf{W}_2$ describes colleagueship, and $\mathbf{W}_3$ describes mutual group membership. The coefficient ρ_i describe the effect size of correspondent network $\mathbf{W}_i$. 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.

The error of the model is modeled as augmented error. It consists of two parts, $\boldsymbol{\epsilon}$ and $\mathbf{u}$. $\boldsymbol{\epsilon}$ 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, consequently the calculation of the likelihood function will also be simplified.

The augmented error results in $\mathbf{z}$, given parameters $\boldsymbol{\beta}$, ρ and σ^2, a normal distribution with mean $\mathbf{X}\boldsymbol{\beta}$ and variance $\mathbf{Q}$.

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

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 $\mathbf{Q}$ we can could sense computationally this is a significant problem.

3.2 Expectation-Maximization Solution

We first develop an approach by maximizing the likelihood of the model using E-M. Since $\mathbf{z}$ is latent, we treat it as unobservable data, for which the E-M algorithm is one of the most used methods. Detailed description of our solution for k regimes of network effects is in Appendix A.

The method consists of two steps: first, estimate the expected value of functions of the unobserved $\mathbf{z}$ given the current parameter set $\boldsymbol{\phi}$, ($\boldsymbol{\phi} = \{\boldsymbol{\beta}, \boldsymbol{\rho}, \sigma^2\}$). Second, 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 need to be estimated.

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

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

$$\begin{aligned}
\boldsymbol{\beta}^{(t+1)} &= \arg\max_{\boldsymbol{\beta}} Q(\boldsymbol{\beta} \mid \boldsymbol{\beta}^{(t)}); \\
\boldsymbol{\rho}^{(t+1)} &= \arg\max_{\boldsymbol{\rho}} Q(\boldsymbol{\rho} \mid \boldsymbol{\rho}^{(t)}); \\
\Sigma^{(t+1)} &= \arg\max_{\Sigma} Q(\boldsymbol{\Sigma} \mid \boldsymbol{\Sigma}^{(t)})
\end{aligned}$$

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 parameter σ^2, the variance of autocorrelation term $\boldsymbol{\theta}$. Let $\sigma^2 = \Sigma$

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

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 not solvable analytically, and numerical methods are needed to get the estimators for all parameters. As it happens, in its current form the E-M algorithm produces a degenerate solution. 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 produces a singular variance-covariance matrix estimate using the Hessian approximation. Thus we have to find another solution.

3.3 Full 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 a hierarchical

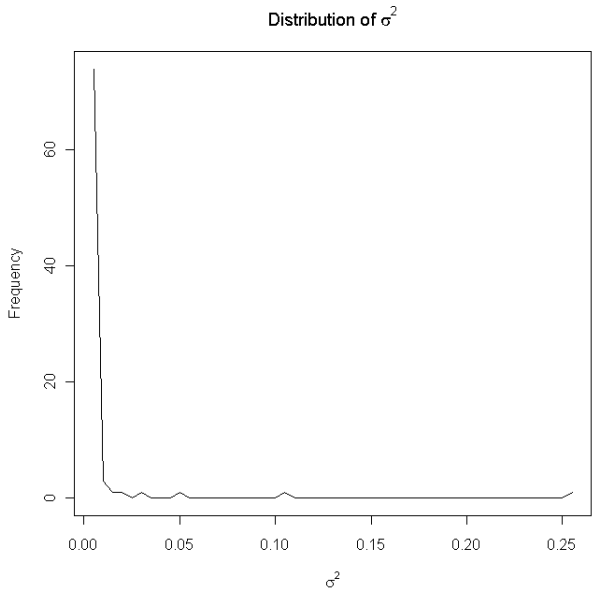


Figure 1: Distribution of σ^2, variance of $\boldsymbol{\theta}$, estimated by E-M solution

Bayesian method. In addition to the model specification above, the prior distributions for each of the highest-level parameters in the model are also need to be specified. $\mathbf{y}$ is the observed dichotomous choice and calculated by the latent preference $\mathbf{z}$. The estimation (MCMC method) is done by sequentially generating draws from a series of full conditional distributions, which are derived from the joint distribution; 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. Given the observed

Table 1: RCMS table for hierarchical Bayesian solution

Parameter	**Density**	**Draw Type**
$\mathbf{z}$	$\text{TrunNormal}_n(\mathbf{X}\boldsymbol{\beta}+\boldsymbol{\theta}, I_n)$	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	$\text{InvGamma}(a, b)$	Parallel
ρ_i	Metropolis step	Sequential

choice of consumer, the latent variable $\mathbf{z}$ can be generated from a truncated normal distribution with a mean of $\mathbf{X}\boldsymbol{\beta}+\boldsymbol{\theta}$ with unit error. The prior distributions of the parameters (shown in Table 1 are generally adopted from the priors proposed by Smith and LeSage (2004). $\boldsymbol{\beta}$ follows normal distribution with mean $\boldsymbol{\nu}_\beta$ and variance $\boldsymbol{\Omega}_\beta$. σ^2 follows inverse gamma distribution with 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 in 5 steps. Detailed description of my method, including the conditional distribution of all parameters, is given in B.

3.4 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 use 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 parameters in a principled way. First, we draw the parameters 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 prior distribution; $\hat{\theta}$ 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:

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

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

distributions.

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

We performed a simulation of 10 replications 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, 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 2. The X-axis

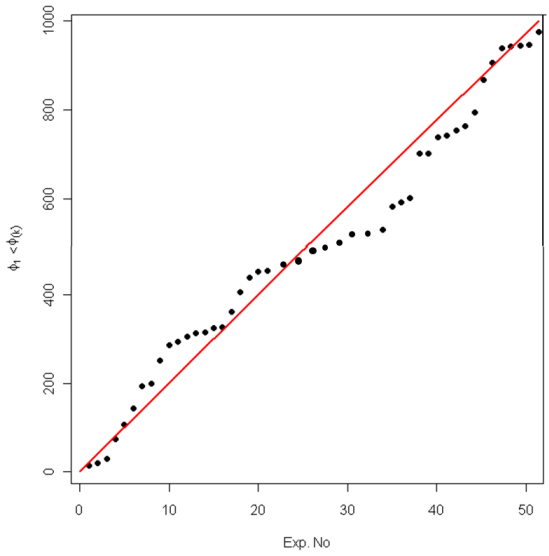
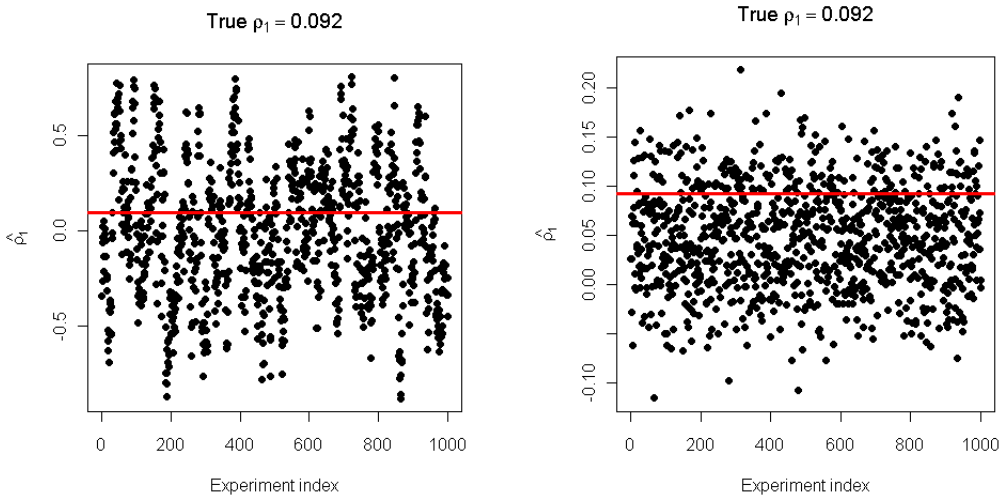


Figure 2: Distribution of sorted quantiles of parameters, $\beta_1, \beta_2, \rho_1, \rho_2, \sigma^2$, 10 replications of posterior quatiles experiments

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, hence we can apply our software to experiments and return correct estimates.

4 Experiments

We next 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 non-ideal conditions. We first choose a prior distribution for ρ with high variance, $\rho \sim \text{Normal}(0, 100)$. As shown in Figure 3(a), the posterior draws of ρ have strong autocorrelation. To compare, we 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 the autocorrelation is considerably smaller, if not zero. So posterior distribution of ρ is sensitive to its prior distribution.



(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

With such high autocorrelation between sequential draws, the effective sample size is extremely small. We therefore use a high degree of thinning to produce uncorrelated draws from the posterior.

We use Yang and Allenby (2003)'s Japanese car data to study 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 or not. The interdependence in the network are 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.

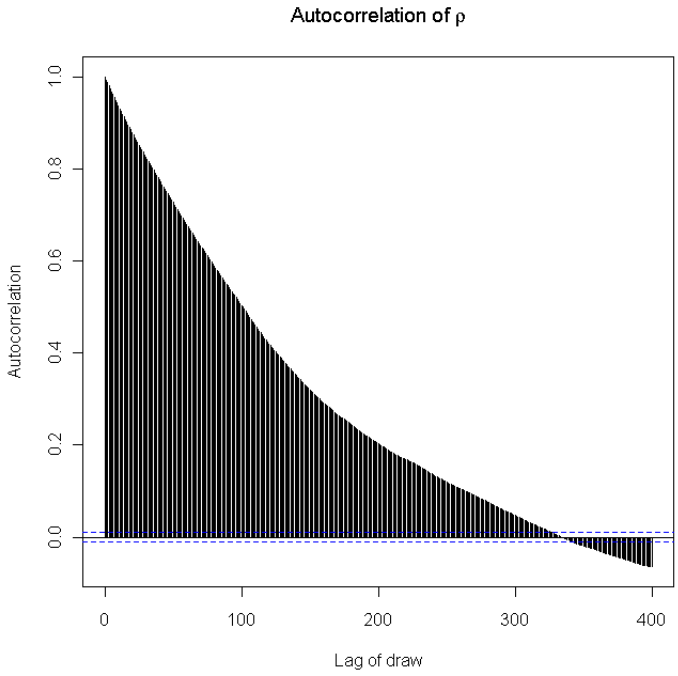


Figure 4: Autocorrelation plot of ρ

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 proper 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 below:

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

where $A_{ik} = 1$ if node i and k are neighbors, and 0 otherwise. 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.

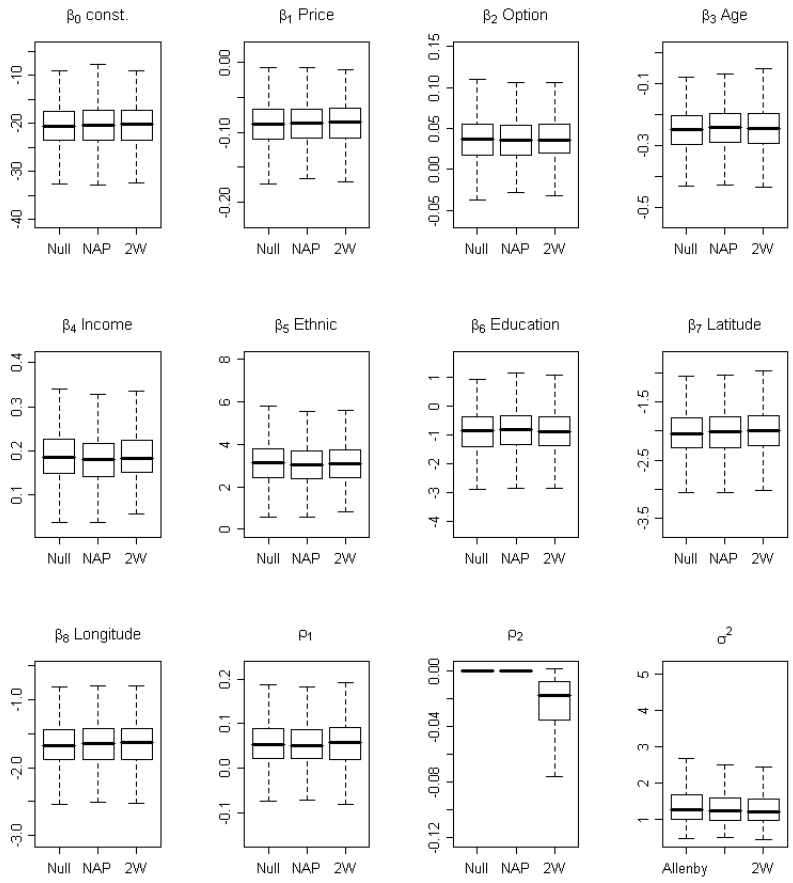


Figure 5: Coefficient estimates comparison

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 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 identical estimates, thus confirmed our software returns correct parameter estimates.

We intend to 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.

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 small number to large number, and observe whether our solution can capture the variation.

Additionally, 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.

References

Agarwal, R., Gupta, A. K., and Kraut, R. (2008). Editorial overview – the interplay between digital and social networks. *Information Systems Research*, 19(3):243–252.

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.

Brancheau, C. J. and Wetherbe, C. J. (1990). The adoption of spreadsheet software: Testing innovation diffusion theory in the context of end-user computing. *Information Systems Research*, 1(2):115–143.

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

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.

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.

Manski, C. F. (2000). Economic analysis of social interactions. *Journal of Economic Perspectives*, 14(3):115–136.

Oinas-Kukkonen, H., Lyytinen, K., and Yoo, Y. (2010). Social networks and information systems: Ongoing and future research streams. *Journal of the Association for Information Systems*, 11(2).

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

Premkumar, G., R. K. and Nilakanta, S. (1994). Implementation of electronic data interchange: an innovation diffusion perspective. *Journal of Management Information Systems - Special section: Strategic and competitive information systems archive*, 11(2).

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.

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.

Appendices

A E-M solution implementation

A.1 Deduction

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

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

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:

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

The right side of equation (2) are two distributions we already have, as shown below.

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

Consider parameter $\boldsymbol{\beta}$ only,

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

Assume Var(**z**)=1,

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

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

$$\begin{aligned}
\mathrm{E}[\mathbf{z}]^{(t+1)} &= \mathrm{E}[\mathbf{z}|\mathbf{y}, \boldsymbol{\beta}^{(t)}] = f(\boldsymbol{\beta}^{(t)}, \mathbf{y}) \\
\log L(\boldsymbol{\beta}, \boldsymbol{\rho}, \sigma^2|\mathbf{z}) &= \log p(\mathbf{z}|\boldsymbol{\beta}, \boldsymbol{\rho}, \sigma^2) \\
&= \log \prod_{i=1}^{n} p(z_i|\boldsymbol{\beta}, \boldsymbol{\rho}, \sigma^2) \\
&= \sum_{i=1}^{n} \log \frac{1}{\sqrt{2\pi|\mathbf{Q}|}} - \frac{1}{2}(\mathbf{z} - \mathbf{X}\boldsymbol{\beta})^\top \mathbf{Q}^{-1}(\mathbf{z} - \mathbf{X}\boldsymbol{\beta}) \\
&= \sum_{i=1}^{n} \log \frac{1}{\sqrt{2\pi|\mathbf{Q}|}} - \left(\frac{1}{2}\mathbf{z}^\top \mathbf{Q}^{-1}\mathbf{z} - \mathbf{z}^\top \mathbf{Q}^{-1}\mathbf{X}\boldsymbol{\beta} - \mathbf{X}^\top \boldsymbol{\beta}\mathbf{Q}^{-1}\mathbf{z} + \mathbf{X}^\top \boldsymbol{\beta}\mathbf{Q}^{-1}\mathbf{X}\boldsymbol{\beta}\right) \qquad (3)
\end{aligned}$$

If decompose the matrices above as vector product, then:

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

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

A.2 Expectation step

In the expectation step, get the expected log-likelihood of parameters.

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

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

A.3 Maximization step

In the maximization step, get the parameter estimates maximizing the expected log-likelihood. First, estimate $\boldsymbol{\beta}$

$$\begin{aligned}\boldsymbol{\beta}^{(t+1)} &= \arg\max_{\boldsymbol{\beta}} Q(\boldsymbol{\phi}|\boldsymbol{\phi}^{(t)}) \\ &= \arg\max_{\boldsymbol{\beta}} \sum_{i=1}^{n} \log \frac{1}{\sqrt{2\pi|\mathbf{Q}|}} - \frac{1}{2}(\mathbf{z}-\mathbf{X}\boldsymbol{\beta})^\top \mathbf{Q}^{-1}(\mathbf{z}-\mathbf{X}\boldsymbol{\beta}) \end{aligned} \tag{4}$$

If directly apply analytical method to solve the Equation (4) above, then:

$$\begin{aligned}\frac{\partial \log L}{\partial \boldsymbol{\beta}} &= \frac{\partial}{\partial \boldsymbol{\beta}} \left(-\frac{1}{2}(\mathbf{z}-\mathbf{X}\boldsymbol{\beta})^\top \mathbf{Q}^{-1}(\mathbf{z}-\mathbf{X}\boldsymbol{\beta}) \right) \\ \frac{\partial}{\partial \boldsymbol{\beta}}(\mathbf{z}-\mathbf{X}\boldsymbol{\beta})^\top \mathbf{Q}^{-1}(\mathbf{z}-\mathbf{X}\boldsymbol{\beta}) &= \frac{\partial}{\partial \boldsymbol{\beta}}(\mathbf{z}^\top \mathbf{Q}^{-1}\mathbf{z} - \mathbf{z}^\top \mathbf{Q}^{-1}\mathbf{X}\boldsymbol{\beta} - \boldsymbol{\beta}^\top \mathbf{X}^\top \mathbf{Q}^{-1}\mathbf{z} + \boldsymbol{\beta}^\top \mathbf{X}^\top \mathbf{Q}^{-1}\mathbf{X}\boldsymbol{\beta}) \\ &= -\mathbf{z}^\top \mathbf{Q}^{-1}\mathbf{X} - \mathbf{X}^\top \mathbf{Q}^{-1}\mathbf{z} + \mathbf{X}^\top \mathbf{Q}^{-1}\mathbf{X}\boldsymbol{\beta} \end{aligned} \tag{5}$$

Set Equation (5) as 0, then:

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

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

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

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

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

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

It is impossible to get the analytical solution for ρ_i.

Third, estimate parameter σ^2. Let $\sigma^2 = \Sigma$

$$\Sigma^{(t+1)} = \arg\max_{\Sigma} Q(\boldsymbol{\phi}|\boldsymbol{\phi}^{(t)})$$
$$\frac{\partial \log L}{\partial \Sigma} = \frac{\partial}{\partial \Sigma}\left(-\frac{1}{2}\log|\mathbf{Q}| - \frac{1}{2}(\mathbf{z}-\mathbf{X}\boldsymbol{\beta})^\top \mathbf{Q}^{-1}(\mathbf{z}-\mathbf{X}\boldsymbol{\beta})\right) \tag{6}$$

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:

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

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.

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

Set σ_0^2 as $\frac{1}{400}$, a small number close to 0, compared with 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:

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

   $\therefore \boldsymbol{\beta} \sim \text{Normal}\left(\mathbf{X}^{-1}(\mathbf{z} - \boldsymbol{\theta}),\ (\mathbf{X}^\top \mathbf{X})^{-1}\right)$
   Based on law of initial values, $\boldsymbol{\Omega}_\beta = \left(\mathbf{D}^{-1} + \mathbf{X}^\top \mathbf{X}\right)^{-1}$

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 $\mathbf{B} = I_n - \sum_i \rho_i \mathbf{W}_i$

$$\begin{aligned} \boldsymbol{\theta} &= \sum_i \rho_i \mathbf{W}_i + \mathbf{u} \\ (I_n - \sum_i \rho_i \mathbf{W}_i)\boldsymbol{\theta} &= \mathbf{u} \\ \mathbf{B}\boldsymbol{\theta} &= \mathbf{u} \\ \boldsymbol{\theta} &= \mathbf{B}^{-1}\mathbf{u} \end{aligned}$$

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

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

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

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

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

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

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 for $\mathbf{W}$, ρ_i, using Metropolis-Hasting sampling with a random walk chain.

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

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

The accepting probability α is obtained by:

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

C Solution diagnostic

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.

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

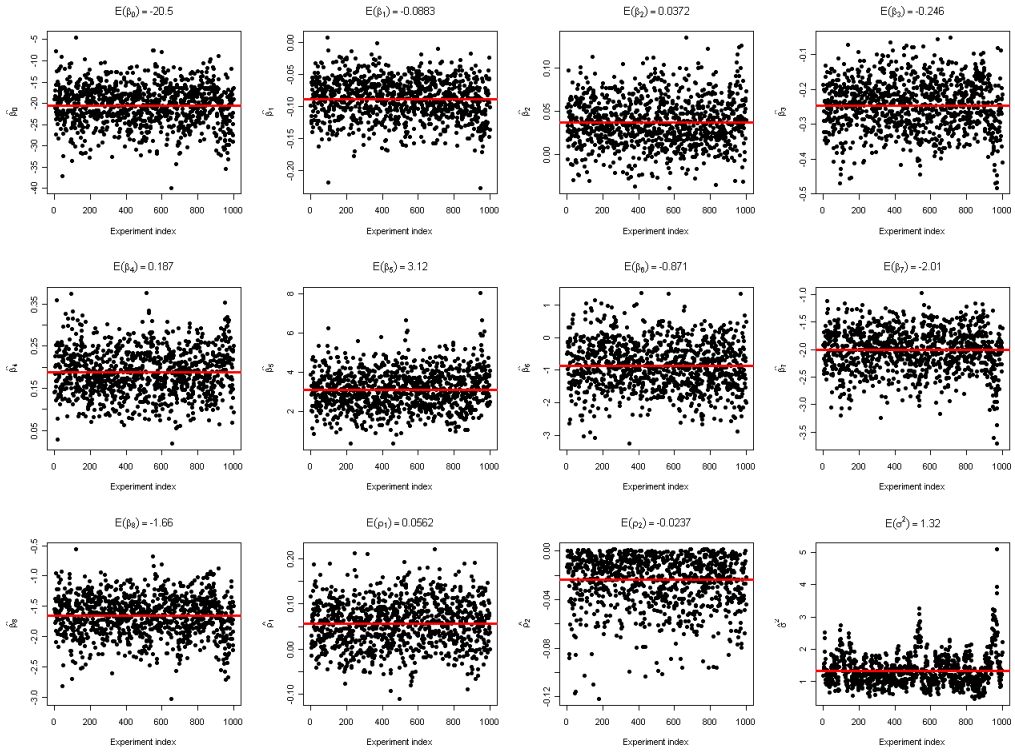


Figure 6: Trace plot of a two-network auto-probit model

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.

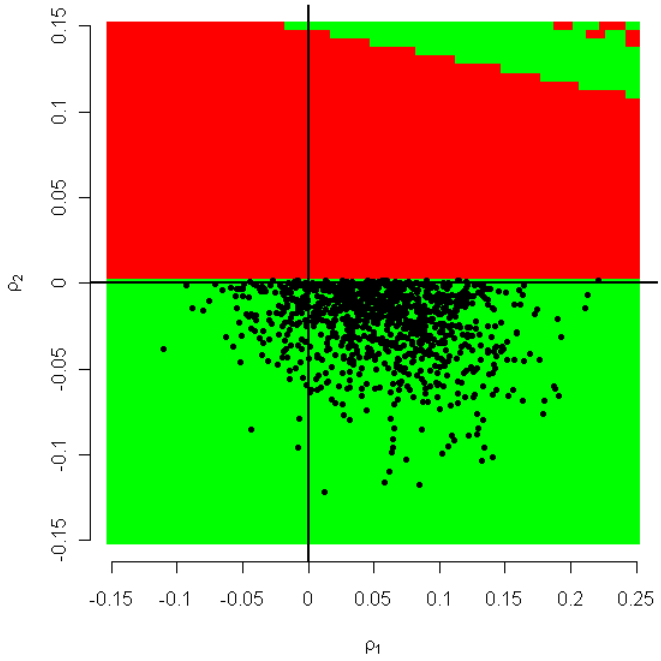


Figure 7: Scatter plot of ρ_1 and ρ_2 on valid region for invertible $\mathbf{B}$,