Sparse Matrix-Variate *t* Process Blockmodels

Zenglin Xu
Dept. of Computer Science
Purdue University
West Lafayette, IN 47907
xu218@purdue.edu

Feng Yan
Dept. of Computer Science
Purdue University
West Lafayette, IN 47907
yan12@purdue.edu

Yuan Qi
Depts. of Computer Science and Statistics
Purdue University
West Lafayette, IN 47907
alanqi@cs.purdue.edu

Abstract

We consider the problem of modeling network interactions and identifying latent groups of network nodes. This problem is challenging due to the facts i) that the network nodes are interdependent instead of independent, ii) that the network data are very noisy (e.g., missing edges), and iii) that the network interactions are often sparse. To address these challenges, we propose a Sparse Matrix-variate t process Blockmodel (SMTB). In particular, we generalize a matrix-variate t distribution to a t process on matrices with nonlinear covariance functions. Due to this generalization, our model can estimate latent memberships for individual network nodes. This separates our model from previous t distribution based relational models. Also, we introduce sparse prior distributions on the latent membership parameters to select group assignments for individual nodes. To learn the model efficiently from data, we develop a variational method. When compared with several state-of-the-art models, including the predictive matrix-variate t models and mixed membership stochastic blockmodels, our model achieved improved prediction accuracy on real world network datasets.

Introduction

A critical task in relational learning is to model interactions among objects in a network, such as proteins in an interaction network and people in a social network, and to identify *latent* groups in the network. This task is encountered for many real-world applications. For example, we might want to discover common research interests from groups of researchers who are co-authors of many papers, or predict the functions of a protein based on a latent group it belongs to.

This task, however, presents new modeling challenges. First, we cannot use classical independence or exchangeability assumptions made in machine learning and statistics for relational data analysis; the objects are interdependent via interactions or links between them, necessitating new models that capture relations among objects. Second, the relationships among objects may be quite complicated. A simple linear (or bilinear) model may not be sufficient to model the complex relationships. Third, the network data are often sparse; since the nodes of a network are often far from being fully connected, an adjacent matrix representing the network structure contains many zeros. This sparsity imposes additional difficulty for modeling.

Copyright © 2011, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

To address these challenge, we propose a Sparse Matrix-variate t process Blockmodel (SMTB). A t distribution is known to enhance sparsity and has been used in many sparse Bayesian models, such as variational relevance vector machine (Bishop and Tipping 2000) and sparse probabilistic projection (Archambeau and Bach 2009). Recently matrix-variate t distributions on matrices have been used to model relational data (e.g., (Yu, Tresp, and Yu 2007; Zhu, Yu, and Gong 2008)). We extend the work in (Zhu, Yu, and Gong 2008) in two ways: i) While (Zhu, Yu, and Gong 2008) matrix-variate t distribution model (MVTM) has high prediction accuracy in term of modeling interactions between nodes, it cannot reveal latent groups of nodes in a network. By contrast, we use nonlinear covariance functions in our model so that we generalize the matrix-variate t distributions to a stochastic process on matrices. This generalization allows us to estimate latent memberships for individual network nodes. ii) Also, we introduce sparse prior distributions on the latent membership parameters, such that the model selects group assignments for individual nodes. In particular, we use an exponential prior distribution that not only forces the latent membership parameters to be nonnegative but also serves as a sparsity regularizer. Furthermore, we present an efficient method to learn the new model efficiently from data. When compared with several state-of-the-art models, including the predictive matrix-variate t models (MVTM) (Zhu, Yu, and Gong 2008) and mixed membership stochastic blockmodels (MMSB) (Airoldi et al. 2008), our model achieved improved prediction accuracy on real world network datasets.

The rest of the paper is organized as follows. In Section 2, we present the proposed sparse matrix-variate t process blockmodel. In Section 3, we describe related work. Section 4 presents experimental results, followed by the conclusions in Section 5.

Sparse Matrix-variate t Process Blockmodels

First we introduce our notations. We denote a constant by c and an identity matrix by $\mathbf{I}$. We use a n by n interaction matrix $\mathbf{Y}$ to represent the *noisy* binary relationships between n network nodes. We denote the index set of observed interactions by $\mathbb{O}$. We use a n by n latent interaction matrix $\mathbf{X}$ to

represent the noiseless version of $\mathbf{Y}$. We represent the d by 1 membership vector for node i as $\mathbf{u}_i$, where d is the number of latent clusters. All the membership vectors are put together in the matrix $\mathbf{U} = (\mathbf{u}_1, \ldots, \mathbf{u}_n) \in \mathbb{R}^{d \times n}$. Given the partially observed matrix $\mathbf{Y}_{\mathbb{O}}$, our objective is to predict missing interactions in $\mathbf{Y}$ and estimate $\mathbf{U}$ to identify latent groups of networks nodes.

Matrix-variate t process models

In the relational setting, we assume that latent matrix $\mathbf{X}$ takes the form:

$$\mathbf{X} = \mathbf{U}^\top \mathbf{W} \mathbf{U}, \tag{1}$$

where $\mathbf{W} \in \mathbb{R}^{d \times d}$ denotes the interactions among groups and the membership matrix $\mathbf{U}$ represents the assignment of individual nodes to latent groups. If $\mathbf{W}$ is an identity matrix, $\mathbf{X}$ becomes the direct product of $\mathbf{U}^\top$ and $\mathbf{U}$ and this model reduces to classical matrix factorization.

Since interactions tend to be sparse, we hope $\mathbf{X}$ can be modeled as a sparse matrix. To this end, we use a matrix-variate t distribution (Gupta and Nagar 2000) on $\mathbf{W}$, i.e., $\mathbf{W} \sim \mathcal{T}_{d,d}(\mathbf{W}; \rho, \mathbf{0}, \Omega, \Upsilon)$, where ρ is the degree of freedom, and Ω and Υ define the column-wise and row-wise covariance matrix respectively. We then have $\mathbf{X} \sim \mathcal{T}_{d,d}(\mathbf{U}^\top \mathbf{W} \mathbf{U}; \rho, \mathbf{0}, \mathbf{U}^\top \Omega \mathbf{U}, \mathbf{U}^\top \Upsilon \mathbf{U})$.

Now we set $\Omega = \mathbf{I}$ and $\Upsilon = \mathbf{I}$. Replacing $\mathbf{U}$ by a mapping $\phi(\mathbf{U})$, we obtain $\phi(\mathbf{U})^\top \Omega \phi(\mathbf{U}) = \mathbf{K}(\mathbf{U}, \mathbf{U})$ as the covariance matrix for columns of $\mathbf{X}$. Using another mapping for $\mathbf{U}$, we obtain $\mathbf{G}(\mathbf{U}, \mathbf{U})$ as the covariance matrix for rows of $\mathbf{X}$ (different mappings allow us to obtain model the column-wise and row-wise relationships differently). As a result, $\mathbf{X}$ follows a matrix-variate t process. The matrix-variate t process is a nonparametric Bayesian model on matrices. Formally, we have the following definition:

Definition 1 (Matrix-variate t process) . *A matrix-variate t process is a stochastic process whose projection on any finite matrix follows a matrix-variate t distribution.*

Specifically, the t process on $\mathbf{X}$ has the following form:

$$\mathbf{X} \sim \mathcal{TP}_{n,n}(\mathbf{X}; \rho, \mathbf{0}, \mathbf{K}, \mathbf{G}), \tag{2}$$

i.e.,

$$\begin{aligned} p(\mathbf{X}) &= \frac{\Gamma_n[\frac{1}{2}(\rho + 2n - 1)]}{\pi^{\frac{1}{2}n^2} \Gamma_n[\frac{1}{2}(\rho + n - 1)]} |\mathbf{K}|^{-\frac{1}{2}n} |\mathbf{G}|^{-\frac{1}{2}n} \\ &\quad |\mathbf{I}_n + \mathbf{K}^{-1} \mathbf{X} \mathbf{G}^{-1} \mathbf{X}^\top|^{-\frac{1}{2}(\rho + 2n - 1)}, \end{aligned} \tag{3}$$

Noise Model

We consider a Gaussian distribution to model the noise between the observable measurement $\mathbf{Y}_{\mathbb{O}}$ and the latent variation $\mathbf{X}$. We then have

$$\mathbf{Y}_{ij} = \mathbf{X}_{ij} + \epsilon_{ij},$$

where $\epsilon_{ij} \sim \mathcal{N}(\mathbf{0}, \sigma^2)$ and the density of $\mathbf{X}$ is defined in Eq.(2). Therefore, the log probability of the noise model is

$$\begin{aligned} \ln P(\mathbf{Y}_{\mathbb{O}} | \mathbf{X}) &\equiv \ell(\mathbf{Y}_{\mathbb{O}}, \mathbf{X}) \\ &= -\frac{1}{2\sigma^2} \sum_{(i,j) \in \mathbb{O}} (\mathbf{Y}_{i,j} - \mathbf{X}_{i,j})^2 + \text{c}. \end{aligned} \tag{4}$$

Variational approximation

Our task is to estimate the parameter $\mathbf{U}$. Ideally we want to maximize the evidence, i.e., $P(\mathbf{Y}_{\mathbb{O}} | \mathbf{U}) = \int P(\mathbf{Y}_{\mathbb{O}} | \mathbf{X}) P(\mathbf{X} | \mathbf{U}) d\mathbf{X}$ over $\mathbf{U}$. However, the computation of the evidence is intractable since we cannot marginalize out the latent variable $\mathbf{X}$ parameter in this integration.

One can use a Markov Chain Monte Carlo method to sample the parameter. However, due to the large size of $\mathbf{Y}_{\mathbb{O}}$, a sampling method could be very slow. In this work, we employ a variational approximation method in an expanded model.

Specifically, we first expand the original t process prior:

$$\begin{pmatrix} \mathbf{X} & \mathbf{R} \\ \mathbf{L} & \mathbf{Z} \end{pmatrix} \sim \mathcal{TP}_{r,r}(:; \rho, \mathbf{0}, \begin{pmatrix} \mathbf{K} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_m \end{pmatrix}, \begin{pmatrix} \mathbf{G} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_m \end{pmatrix}),$$

where $r = m + n$.

Then we will use the following properties of a joint t distribution.

Theorem 1

$$P(\mathbf{X}) = \mathcal{TP}_{n,n}(\mathbf{X}; \rho, \mathbf{0}, \mathbf{K}, \mathbf{G}) \tag{5}$$

$$P(\mathbf{Z}) = \mathcal{T}_{m,m}(\mathbf{Z}; \rho, \mathbf{0}, \mathbf{I}_m, \mathbf{I}_m) \tag{6}$$

$$P(\mathbf{X} | \mathbf{Z}, \mathbf{R}, \mathbf{L}) = \mathcal{TP}_{n,n}(\mathbf{X}; \rho + n + m, \boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\Psi}), \tag{7}$$

$$P(\mathbf{R} | \mathbf{Z}) = \mathcal{T}_{n,m}(\mathbf{R}; \rho + m, \mathbf{0}, \mathbf{K}, \mathbf{Z}^\top \mathbf{Z} + \mathbf{I}_m), \tag{8}$$

$$P(\mathbf{L} | \mathbf{Z}) = \mathcal{T}_{m,n}(\mathbf{L}; \rho + m, \mathbf{0}, \mathbf{Z}^\top \mathbf{Z} + \mathbf{I}_m, \mathbf{G}), \tag{9}$$

where $\boldsymbol{\mu} = \mathbf{R}^\top \mathbf{Z} (\mathbf{Z}^\top \mathbf{Z} + \mathbf{I}_m)^{-1} \mathbf{R}$, $\boldsymbol{\Sigma} = \mathbf{K} + \mathbf{R}^\top (\mathbf{Z}^\top \mathbf{Z} + \mathbf{I}_m)^{-1} \mathbf{R}$ and $\boldsymbol{\Psi} = \mathbf{G} + \mathbf{L}^\top (\mathbf{Z}^\top \mathbf{Z} + \mathbf{I}_m)^{-1} \mathbf{L}$. We denote by $\Theta = \{\mathbf{Z}, \mathbf{R}, \mathbf{L}\}$ as the free variables.

Theorem 1 suggests that we can employ the conditional distributions over Θ to approximate a distribution on $\mathbf{X}$. The approximation could be efficient since $m \leq n$.

Now we use this idea to approximate the joint log-likelihood of the expanded model:

$$\begin{aligned} & \ln P(\mathbf{Y} | \mathbf{U}) \\ = \; & \ln \int P(\mathbf{Y} | \mathbf{X}) P(\mathbf{X} | \mathbf{U}, \Theta) P(\Theta) d\mathbf{X} d\Theta \\ \approx \; & \ln P(\Theta_{\text{MAP}}) + \ln \int P(\mathbf{Y} | \mathbf{X}) P(\mathbf{X} | \mathbf{U}, \Theta_{\text{MAP}}) d\mathbf{X} \\ \geq \; & \ln P(\Theta_{\text{MAP}}) + \int \ln P(\mathbf{Y} | \mathbf{X}) P(\mathbf{X} | \mathbf{U}, \Theta_{\text{MAP}}) d\mathbf{X}. \end{aligned} \tag{10}$$

Note that we use a Maximum-a-Posteriori (MAP) approximation to obtain the second equation above. The inequality in the third equation holds because of the concavity of the logarithmic transformation.

Based on the definition of matrix-variate t distributions, we can easily obtain

$$\begin{aligned} \ln P(\Theta) &= \ln P(\mathbf{Z}) + \ln P(\mathbf{R} | \mathbf{Z}) \\ &= -s_1 \ln |\mathbf{I}_m + \mathbf{Z}^\top \mathbf{Z}| - s_2 \ln |\mathbf{K}| - s_2 \ln |\mathbf{G}| \\ &\quad -s_3 \ln |\mathbf{I}_n + \mathbf{K}^{-1} \mathbf{R} (\mathbf{Z}^\top \mathbf{Z} + \mathbf{I}_m)^{-1} \mathbf{R}^\top| \\ &\quad -s_3 \ln |\mathbf{I}_n + (\mathbf{Z}^\top \mathbf{Z} + \mathbf{I}_m)^{-1} \mathbf{R} \mathbf{G}^{-1} \mathbf{R}^\top| + \text{c}, \end{aligned} \tag{11}$$

where $s_1 = \frac{\rho + 2m - 1}{2}$, $s_2 = \frac{n}{2}$, $s_3 = -\frac{\rho + n + m - 1}{2}$ are all constants.

Based on the distribution $P(\mathbf{X}|\Theta) = \mathcal{TP}_{n,n}(\mathbf{X}; \rho + n + m, \boldsymbol{\mu}, \boldsymbol{\Sigma}, \boldsymbol{\Psi})$, we have the following proposition to calculate the mean and variance of $\text{Vec}(\mathbf{X})$.

Proposition 1 *The mean and variance of the vector* $\text{Vec}(\mathbf{X})$ *are given by:*

$$\mathbb{E}(\text{Vec}(\mathbf{X})|\Theta) = \text{Vec}(\boldsymbol{\mu}), \quad (12)$$

$$\text{Cov}\,(\text{Vec}(\mathbf{X})|\Theta) = \frac{1}{\rho + n + m - 2}\boldsymbol{\Sigma} \otimes \boldsymbol{\Psi}, \quad (13)$$

where $\otimes$ *denotes the Kronecker product.*

The result directly comes from (Gupta and Nagar 2000).

Then we can compute the second term in Eq. (10) as follows:

$$\int \ln P(\mathbf{Y}|\mathbf{X}) P(\mathbf{X}|\mathbf{U}, \Theta_{\text{MAP}}) d\mathbf{X}$$
$$= \text{E}\left[-\ell(\mathbf{Y}_{\mathbb{O}}, \boldsymbol{\mu}) + s_4 \sum_{(i,j)\in\mathbb{O}} \boldsymbol{\Sigma}_{i,i}\boldsymbol{\Psi}_{j,j} + \text{c}\right], \quad (14)$$

where $s_4 = \frac{1}{2\sigma^2(\rho+m+n-2)}$ is a constant.

We can further parameterize the above equation by defining $\mathbf{Q} = \mathbf{R}(\mathbf{I}_m + \mathbf{Z}^\top\mathbf{Z})^{-1/2} \in \mathbb{R}^{n\times m}$ and $\mathbf{P} = \mathbf{L}^\top(\mathbf{I}_m + \mathbf{Z}^\top\mathbf{Z})^{-1/2} \in \mathbb{R}^{n\times m}$. We can then have the following minimization problem:

$$\min_{\mathbf{Q},\mathbf{Z},\mathbf{P},\mathbf{U}} \quad f(\mathbf{Q}, \mathbf{Z}, \mathbf{P}, \mathbf{U}) \quad (15)$$

where $f(\mathbf{Q}, \mathbf{Z}, \mathbf{P}, \mathbf{U})$ is defined as

$$\begin{aligned} f(\mathbf{Q}, \mathbf{Z}, \mathbf{P}, \mathbf{U}) = \quad & -\ell(\mathbf{Y}_{\mathbb{O}}, \mathbf{Q}\mathbf{Z}\mathbf{P}^\top) \\ & + s_1 \ln|\mathbf{I}_m + \mathbf{Z}^\top\mathbf{Z}| + s_2 \ln|\mathbf{K}| + s_2 \ln|\mathbf{G}| \\ & + s_3 \ln|\mathbf{I}_n + \mathbf{K}^{-1}\mathbf{Q}\mathbf{Q}^\top| + s_3 \ln|\mathbf{I}_n + \mathbf{G}^{-1}\mathbf{P}\mathbf{P}^\top| \\ & + s_4 \sum_{(i,j)\in\mathbb{O}} \left(\mathbf{K} + \mathbf{Q}\mathbf{Q}^\top\right)_{i,i} \left(\mathbf{G} + \mathbf{P}\mathbf{P}^\top\right)_{j,j} \end{aligned} \quad (16)$$

In the above, $\mathbf{K}$ and $\mathbf{G}$ define the covariance functions of $\mathbf{U}$ by which the nonlinear interaction between $\mathbf{U}$ is modeled. For symmetric data, we simply set $\mathbf{G}$ equal to $\mathbf{K}$.

Sparse prior

To make $\mathbf{U}$ sparse, we impose an exponential prior on $\mathbf{U}$. This is equivalent to adding a L_1 regularizer, i.e.,

$$\min_{\mathbf{Q},\mathbf{Z},\mathbf{P},\mathbf{U}} \quad f(\mathbf{Q}, \mathbf{Z}, \mathbf{P}, \mathbf{U}) + \lambda|\mathbf{U}|_1, \quad (17)$$

where λ is a hyperparameter and we set its value based on cross-validation.

Optimization and prediction

We use a projected gradient descent method to optimize the cost function (17). The optimization results will provide the estimates of latent memberships $\mathbf{U}$.

Furthermore, using the estimates of $\mathbf{P}$, $\mathbf{Q}$ and $\mathbf{Z}$, we can use the conditional mean of $\mathbf{X}$, i.e., $\mathbb{E}(\mathbf{X}) = \mathbf{Q}\mathbf{Z}\mathbf{P}^\top$ to predict unobserved interactions.

It is important to note that by exploring the structure of kronecker products, we avoid the cost of $\mathcal{O}(n^6)$ from a naïve derivation. Instead, the time complexity of the above optimization is mainly dominated by solving a linear system of n variables (n is the number of nodes). Depending on matrix condition numbers, effective linear system solvers give us a cost from $\mathcal{O}(n)$ to $\mathcal{O}(n^3)$. We can easily handle networks of thousands of nodes.

Related Work

Modeling the interaction among nodes in social and biological networks has become an active research area in recent years. One popular approach is the stochastic block model and its variations and extensions, e.g., (Snijders and Nowicki 1997; Wang and Wong 1987; Kemp, Griffiths, and Tenenbaum 2004; Xu et al. 2006; Airoldi et al. 2008; Hoff 2007). This type of approaches assigns each node in a network to one or multiple latent clusters. Our model belongs to this type of approaches too. What separates ours from the previous ones is the nonparametric Bayesian t process modeling, which allows us to capture complex nonlinear network interactions. Also, due to the models' nonparametric nature, the model complexity is adaptive with the amount of data available.

Another type of network (or relational) models focuses on the latent similarity between two nodes and instead of modeling their latent cluster memberships. Such approaches include the latent distance model (Hoff et al. 2001) and matrix-variate t model (Zhu, Yu, and Gong 2008). Although these approaches may achieve accurate predictions for missing interactions, they cannot reveal latent cluster structures, limiting their applications in practice. As described in Section 1, our model is closely related to the work by (Zhu, Yu, and Gong 2008); we generalizes it to the nonparametric model and uses sparse priors to learn latent memberships for network nodes.

Experiment

In this section, we illustrate how our new model, SMTB, works on synthetic data and compare it with alternative methods on several real world network datasets.

Experiment on Synthetic Data

First, we test SMTB on a synthetic dataset to answer the following two questions:

1. Is SMTB robust to noise?
2. Can SMTB output block structures?

To generate the synthetic data, we first randomly sample a 40×40 clean interaction matrix, representing a network with four 10-node cliques. In each clique the nodes are fully connected (so the corresponding sub-matrix is dense), as shown in Figure 1(a). We then randomly remove some elements from the clean interaction matrix and add Gaussian noises to the remaining elements. We use this noisy matrix as our observation $\mathbf{Y}$ (Figure 1(b)). Given $\mathbf{Y}$ we run SMTB to obtain the latent interaction matrix $\mathbf{X}$, as an estimate for the original interaction matrix. The result is shown in Figure 1(c). Clearly, the model identifies the block structure embedded

in the noisy observation $\mathbf{Y}$ and recovers the latent structure to a reasonable accuracy. We also measure the mean square errors (MSE) based on the exact interaction matrix in 1(a). The MSE value of the noisy matrix is 0.269 and that of the estimated $\mathbf{X}$ is only 0.131, demonstrating the power of SMTB in filtering out the network noise and recovering the latent structure.

Furthermore, we plot the estimated membership matrix $\mathbf{U}$ in Figure 1(d). Note that $\mathbf{u}_i$ indicates which latent group node i should belong to. As shown in Figure 1(d), the estimated memberships are consistent with the original block structure in Figure 1(a).

Experiment on Real-world Datasets

We use three real-world datasets to test SMTB. It should be noted that the number of edges in a network is in the quadratic order of the number of nodes, and the prediction will be made on each edge. The large number of edges makes the estimation problem computationally challenging.

The used network datasets are summarized in the following:

- The first dataset represents friendship ties among 90 12^{th}-graders from the National Longitudinal Study of Adolescent Health [1]. The data is represented by a symmetric matrix corresponding to an undirected graph. $Y_{ij} = 1$ means identity nodes i and j are friends. This dataset is named as "Friends".
- The second dataset is a protein-protein interaction data of $E.coli$ (Butland et al. 2005). There are 230 proteins, where $Y_{ij} = 1$ means the i^{th} protein interacts with the j^{th} protein. This dataset is named as "E.coli".
- The third dataset is a protein-protein interaction dataset, which consists of 283 yeast proteins from the third class of the data produced by (Bu et al. 2003). $Y_{ij} = 1$ means the i^{th} protein is likely to function with the j^{th} protein. This data is represented by an asymmetric matrix. Note that by using different column- and row-wise covariance functions, SMTB can be applied to model asymmetric networks. This dataset is named as "Yeast".

On these datasets, we compare our model, SMTB, with the following competitive ones:

- Non-negative Matrix Factorization (NMF) (Lee and Seung 1999). NMF factorizes an interaction matrix to low-dimensional representations with non-negativity constraints. NMF has been successfully applied to a wide range of application and is used as a baseline method here.
- Mixed membership stochastic blockmodels (MMSBs) (Airoldi et al. 2008). MMSB is a state-of-the-art approach for network modeling.
- Predictive matrix-variate t models (MVTMs) (Zhu, Yu, and Gong 2008). MVTM is another advanced model for relational data and closely related to our model.

For nonnegative matrix factorization, we adopt an implementation in the statistics toolbox of Matlab 2009.

[1] www.cpc.unc.edu/projects/addhealth

Data	NMF	MVTM	MMSB	SMTB
	Friends			
d=3	66.10	65.31	72.17	**76.11**
d=5	70.02	67.51	72.03	**74.94**
	$E.coli$			
d=3	75.30	78.89	80.83	**87.40**
d=5	77.15	82.09	83.58	**87.83**
	Yeast			
d=3	89.16	89.85	83.19	**92.58**
d=5	91.07	82.09	81.60	**93.24**

Table 1: The AUC values averaged over 10 runs. We vary the number of the latent groups for all the models. The highest average AUC value for each setting is highlighted.

For the mixed membership stochastic blockmodel, we use the default setting of the software downloaded from the authors' website previously. For the predictive matrix-variate t model, we adopt the code kindly provided by the authors [2]. For both this model and SMTB, we fix the degree of freedom ρ to 10. For SMTB, we use the Gaussian covariance function (i.e., the RBF kernel function). The kernel width is selected from $[0.01, 0.05, 0.1, 0.15, 0.20, 0.25, 0.30, 0.35, 0.40, 0.45, 0.50]$ by five-fold cross validation.

Since for these datasets we do not know the true latent groups, we use the prediction accuracy on hold out edges to compare all these models (actually one cannot even use MVTM to identify latent groups). Specifically for each of these datasets, we randomly choose 80% of the matrix elements (edges) for training and use the remaining for testing. The experiment is repeated 10 times. We evaluate all the models by Receiver Operating Characteristic (ROC) curves and Area Under Curve (AUC) values averaged over 10 runs.

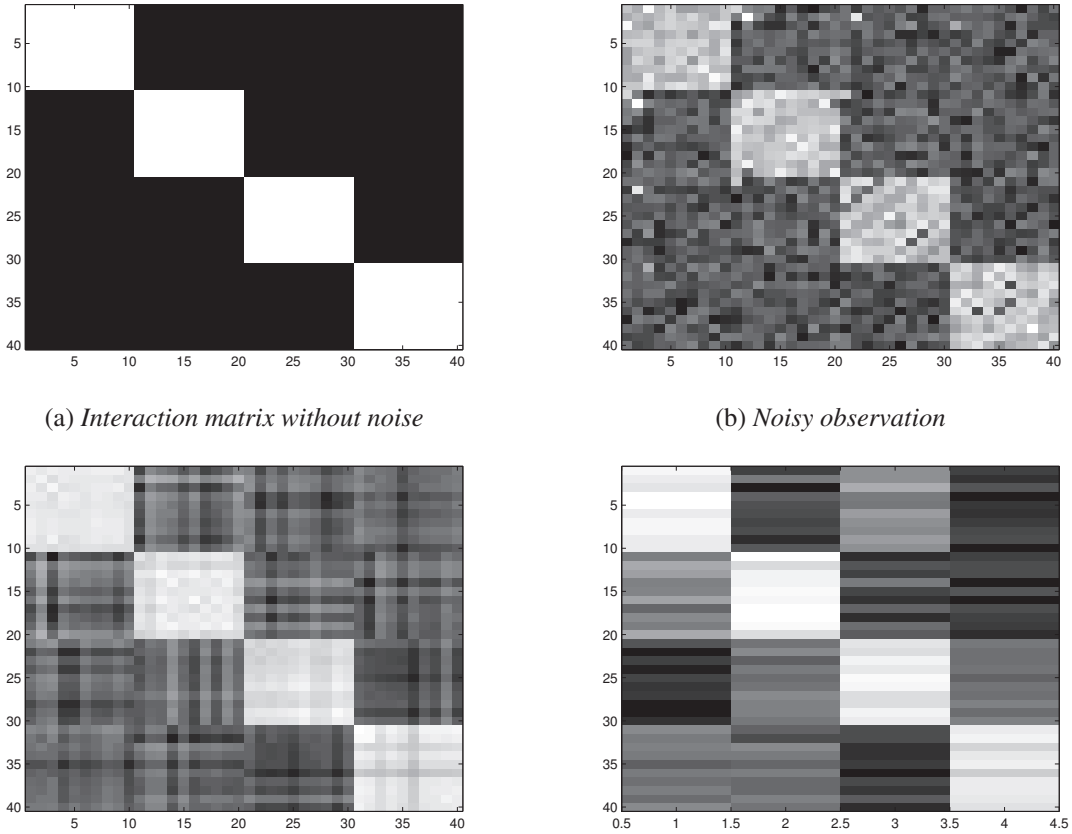
Figure 2 shows the ROC curves of all the models. The higher a ROC curve, the better the predictive performance. We change the number of latent clusters (i.e., the length of $\mathbf{u}_i$) from 3 to 5. As shown in Figure 2 SMTB consistently achieves better performance than the other models. Among these models, NMF achieves the lowest accuracy, probably caused by its simple modeling of relational data; NMF simply treats an interaction matrix as a regular matrix without exploring the underlying structure of network data. The performance of MMSB is often better than MVTM but slightly worse than SMTB. A special case appears for the yeast dataset, on which both SMTB and MVTM outperform MMSB. Since this data is very sparse, we expect that the t distributions and processes used by MVTM and SMTB help them achieve higher accuracy.

For a detailed comparison, we report the average AUC values in Table . SMTB consistently outperforms all the other models in terms of the average AUC.

Conclusions and Future Work

In this paper, we have presented a new model, SMTB, for modeling interactions of network nodes and discover-

[2] http://www.nec-labs.com/~zsh/files/MVTM-1.18.zip



(a) *Interaction matrix without noise*

(b) *Noisy observation*

(c) *Estimated latent interaction matrix*

(d) *Estimated membership matrix*

Figure 1: Illustration of SMTB estimation on synthetic data. As shown in (c), SMTB significantly reduces the noise in the observation (b). Also SMTB reveals the correct node memberships shown in (d), consistent with the block structure in the clean (unknown) interaction matrix (a).

ing latent groups in a network. Our results on real network datasets demonstrate SMTB outperforms several the state-of-art models.

As to the future plan, we will explore other likelihood functions (e.g., probit functions) to better model binary interactions or more complex relationships between network nodes.

Acknowledgement

The authors would thank Shenghuo Zhu for discussing MVTM and providing its source code. We also thank Syed Abbas Z. Naqvi for discussing the model and proofreading of this paper. This work was supported by NSF CAREER award IIS-1054903 and NSF IIS-0916443.

References

Airoldi, E. M.; Blei, D. M.; Fienberg, S. E.; and Xing, E. P. 2008. Mixed membership stochastic blockmodels. *J. Mach. Learn. Res.* 9:1981–2014.

Archambeau, C., and Bach, F. 2009. Sparse probabilistic projections. In *Advances in Neural Information Processing Systems 21*.

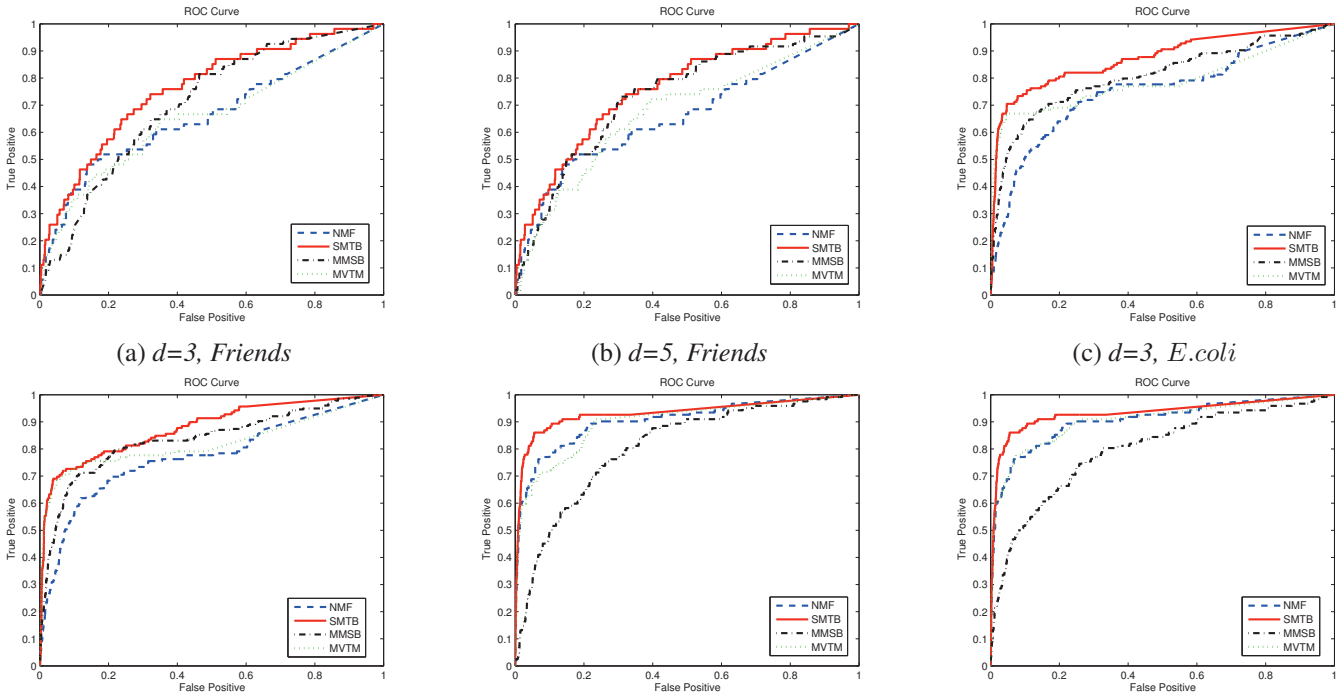
Bishop, C., and Tipping, M. E. 2000. Variational relevance vector machines. In *16th UAI*.

Bu, D.; Zhao, Y.; Cai, L.; Xue, H.; Zhu, X.; Lu, H.; Zhang, J.; Sun, S.; Ling, L.; Zhang, N.; Li, G.; and Chen, R. 2003. Topological structure analysis of the protein-protein interaction network in budding yeast. *Nucl. Acids Res.* 31(9):2443–2450.

Butland, G.; Peregrin-Alvarez, J. M.; Li, J.; Yang, W.; Yang, X.; Canadien, V.; Starostine, A.; Richards, D.; Beattie, B.; Krogan, N.; Davey, M.; Parkinson, J.; Greenblatt, J.; and Emili, A. 2005. Interaction network containing conserved and essential protein complexes in Escherichia coli. *Nature* 433(7025):531–537.

Gupta, A. K., and Nagar, D. K. 2000. *Matrix variate distributions*. CRC Press.

Hoff, P. D.; Raftery, A. E.; Handcock, M. S.; and H, M. S. 2001. Latent space approaches to social network analysis. *Journal of the American Statistical Association* 97:1090–1098.



(a) *d=3, Friends* (b) *d=5, Friends* (c) *d=3, E.coli*

(d) *d=5, E.coli* (e) *d=3, Yeast* (f) *d=5, Yeast*

Figure 2: ROC curves of Non-negative matrix factorization (NMF), mixed membership stochastic blockmodels (MMSBs), and predictive matrix-variate t models on three network datasets (Friends, $E.coli$ and Yeast). We randomly hold out 20% of each network data to test all the models.

Hoff, P. 2007. Modeling homophily and stochastic equivalence in symmetric relational data. In *Advances in Neural Information Processing Systems 20*.

Kemp, C.; Griffiths, T. L.; and Tenenbaum, J. B. 2004. Discovering latent classes in relational data. Technical Report AI Memo 2004-019, MIT.

Lee, D. D., and Seung, H. S. 1999. Learning the parts of objects by non-negative matrix factorization. *Nature* 401(6755):788–791.

Snijders, T. A., and Nowicki, K. 1997. Estimation and prediction for stochastic blockmodels for graphs with latent block structure. *Journal of Classification* 14(1):75–100.

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

Xu, Z.; Tresp, V.; Yu, K.; and Kriegel, H.-P. 2006. Infinite hidden relational models. In *Proceedings of the 22nd Conference on Uncertainty in Artificial Intelligence (UAI 2006)*.

Yu, S.; Tresp, V.; and Yu, K. 2007. Robust multi-task learning with t-processes. In *ICML'07: Proceedings of the Twenty-Fourth International Conference on Machine Learning*, 1103–1110.

Zhu, S.; Yu, K.; and Gong, Y. 2008. Predictive matrix-variate t models. In *Advances in Neural Information Processing Systems 20 (NIPS 2007)*, 1721–1728.