Dynamic Community Detection Using Dependent Latent Position Model

Lizhen Lin, Mengjie Wang
Department of Statistics and Data Science, University of Texas at Austin
lizhen.lin@austin.utexas.edu, wangmj2013@utexas.edu

Abstract

Community detection in network analysis has drawn more and more attention recently in many areas. We perform dynamic community detection using dependent latent position model by introducing the dependence among latent positions across time. Clustering of nodes is done via clustering the corresponding latent positions in a model-based framework. The link probability between each pair of nodes is calculated through a logit link and the latent positions are modeled using a dependent Dirichlet Process mixture model. Efficient MCMC algorithms will be developed and applications are considered for both simulated and real data sets.

Introduction

Networks are good data representation tools to show the relations between interacting nodes or pairs. They are widely used in many fields, for example, in social networks, each nodes stands for a person or a group, and each link between nodes stands for the connection between two them.

Social network data typically consist of a set of n nodes and a relational tie A_{ij}, which acts as an indicator and measured on each ordered pair of nodes $i, j = 1, \dots, n$. Here we mainly focus on the latent space model. The latent space model introduced by [2] is a stochastic model of the network in which each node has a latent position, the probability of a tie uniquely depends on the Euclidean distance between latent positions z_i and z_j, that is, $P(A_{ij}) = f(z_i, z_j)$, where f is called transition function, usually logit link function. [1] extends this latent space model to cluster the nodes in a natural way in addition to take account of transitivity and homophily by using model-based clustering idea.

Our model can be viewed as an extension of Handcock's model in the sense of assuming the latent positions comes from a dependent Dirichlet Process mixture model, which introduces dependence across time points and does not prefix the number of groups .

Dependent Latent Position Model

Let A be an n by n observed adjacency matrix of the binary network data. A_{ij}=1 if there is an edge or link between nodes i and j, otherwise $A_{ij} = 0$. Let π_{ij} be the probability of having an edge (link or edge probability) between node i and j. We assume A_{ij} follows an *independent Bernoulli model* with link probability π_{ij}. That is,

$$A_{ij} \sim \text{Ber}(\pi_{ij}).$$

To deal with the problem of *dynamic community detection*, consider n nodes with labels in $[n] = \{1, \dots, n\}$ and T time points $t = 1, \dots, T$. We have

$$A_{ij}^{(t)} \sim \text{Ber}(\pi_{ij}^{(t)}),$$

for any $t = 1, \dots, T$. For each node i, at each time t, we assign one of the K communities for the node. We label the latent position at each time t as $z_i^{(t)}$ for $i = 1, \dots, n$ and $t = 1, \dots, T$. Denote $\mathbf{z} = \{z_i^{(t)}, i = 1, \dots, n; t = 1, \dots, T\}$ and $\mathbf{A} = \{A^{(1)}, \dots, A^{(T)}\}$. Thus,

$$\pi_{ij}^{(t)} = \frac{\exp(\beta_0^T x_{ij} - \beta_1 \| z_i^{(t)} - z_j^{(t)} \|)}{1 + \exp(\beta_0^T x_{ij} - \beta_1 \| z_i^{(t)} - z_j^{(t)} \|)},$$

where $X = x_{ij}$ is the covariate vector and β is the coefficient to be estimated. Clustering is done by grouping the latent position sequence $\{z_i^{(t)}, i = 1, \dots, n; t = 1, \dots, T\}$. We also use model-based method for clustering by modeling $\{z_i^{(t)}\}$ using a dependent mixture model. The idea is to cluster nodes by clustering their latent positions at each time while allowing dependency or borrowing information across different time points.

We consider the following DDP type of mixture model, for $i = 1, \dots, n$,

$$z_i^{(t)} \sim f(z \mid t) = \int N(\mu, \sigma^2 \mathcal{I}_d) G_t(d\mu) \pi(d\sigma^2),$$

where $N(\mu, \sigma^2 \mathcal{I}_d)$ is the multivariate normal distribution (the mixing kernel) with mean $\mu \in \mathbb{R}^d$ and a diagonal covariance matrix $\sigma^2 \mathcal{I}_d$. Here $G_t(\cdot)$ is the mixing distribution for μ which in our case is a dependent Dirichlet type of process, and $\pi(d\sigma^2)$ is the mixing distribution for the variance parameter σ^2.

We are left with proposing a prior for $\{G_t(\cdot), t = 1, \dots, T\}$. To elucidate the idea of incorporating dependency across different time points, we adopt the following sticking-breaking representation:

$$G_t = \sum_{h=1}^{N(t)} w_h(t) \delta_{\theta_h(t)}(\cdot),$$

where $N(t)$ stands for the number atoms of the mixing measure G_t, $\{w_h(t), h = 1, \dots, N(t)\}$ is the weight sequence, and $\{\theta_h(t), h = 1, \dots, N(t)\}$ are the atoms of the mixing measure.

If one lets $w_l(t) = w_l$ without time dependency, then this is called the *fixed* π or *fixed weights DDP* model. Then one just draw $\{\theta_h(t), h = 1, \dots, N(t)\}$ from some *stochastic process.* Write $\{\theta_h(t), h = 1, \dots, N(t)\}$ as $\{\theta_1^{(1)}, \dots, \theta_H^{(1)}, \dots\}, \{\theta_1^{(2)}, \dots, \theta_H^{(2)}, \dots\}, \dots \{\theta_1^{(t)}, \dots, \theta_H^{(t)}, \dots\}$. One can introduce dependency by for each atom across different time points. Specially, one can assume that

$$\begin{aligned} \{\theta_1^{(1)}, \theta_1^{(2)}, \theta_1^{(3)}, \dots, \theta_1^{(T)}\} &\sim \mathcal{GP}(0, R(\cdot, \cdot)), \\ &\cdots\cdots \\ \{\theta_H^{(1)}, \theta_H^{(2)}, \theta_H^{(3)}, \dots, \theta_H^{(T)}\} &\sim \mathcal{GP}(0, R(\cdot, \cdot)), \\ &\cdots\cdots \end{aligned}$$

$\mathcal{GP}(0, \mathbb{R}(\cdot, \cdot))$ is some Gaussian process with mean 0 and covariance kernel $R(\cdot, \cdot)$. For example, one can pick the standard squared exponential covariance kernel, that is,

$$R(\theta_{t_1}, \theta_{t_2}) = \alpha \exp(-\beta (t_1 - t_2)^2).$$

Data Simulation

Given the link probability $\pi_{ij}^{(t)}$, one can simulate the dynamic binary matrices from the Bernoulli model.

To simulate the latent position sequence that incorporates dependency across different time points, we propose the following:

- At time t_1, simulated $z_i^{(t_1)} \sim \sum_{i=1}^{5} w_i N(\mu_i, \sigma_0^2)$, where the mean locations of the five normal mixtures, namely μ_i $(i = 1, \dots, t)$ are kept fixed. Note that the weights are going to vary at different time points at time t_1, one can let $w_1 = w_2 = \dots = w_5 = 1/5$, in this case we associate a five-dimensional binary vector (1,1,1,1,1) to the weights vector (1/5, 1/5, 1/5, 1/5, 1/5).
- At time t_2, we will keep the five normal mixture fixed but simulate a new weight vector by first simulating a five-dimensional binary vector by flipping the corresponding entry from time t_1 with probability for example 0.8 to keep the same entry as in time t_1. Suppose you simulation gives a binary vector of (1,1,1,0,0), then the corresponding weight vector is (1/3, 1/3, 1/3, 0, 0) which in fact only gives rises three mixture components.
- At time t_3, repeat similarly the previous step.

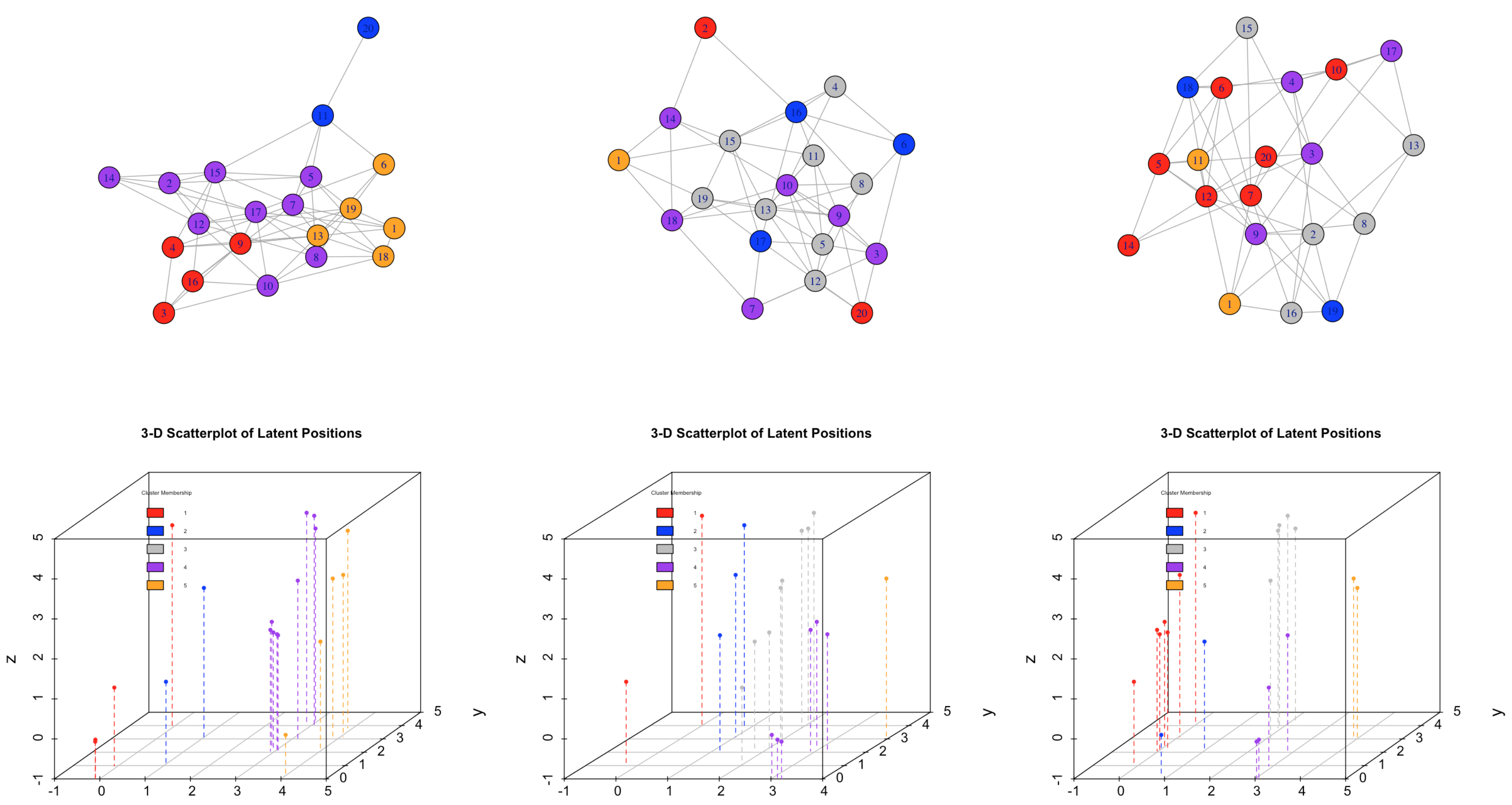

Figure 1: The first row is nodes graph and the second row is the corresponding latent position plot for simulated data (20 nodes, $t = 20, 21, 22$)

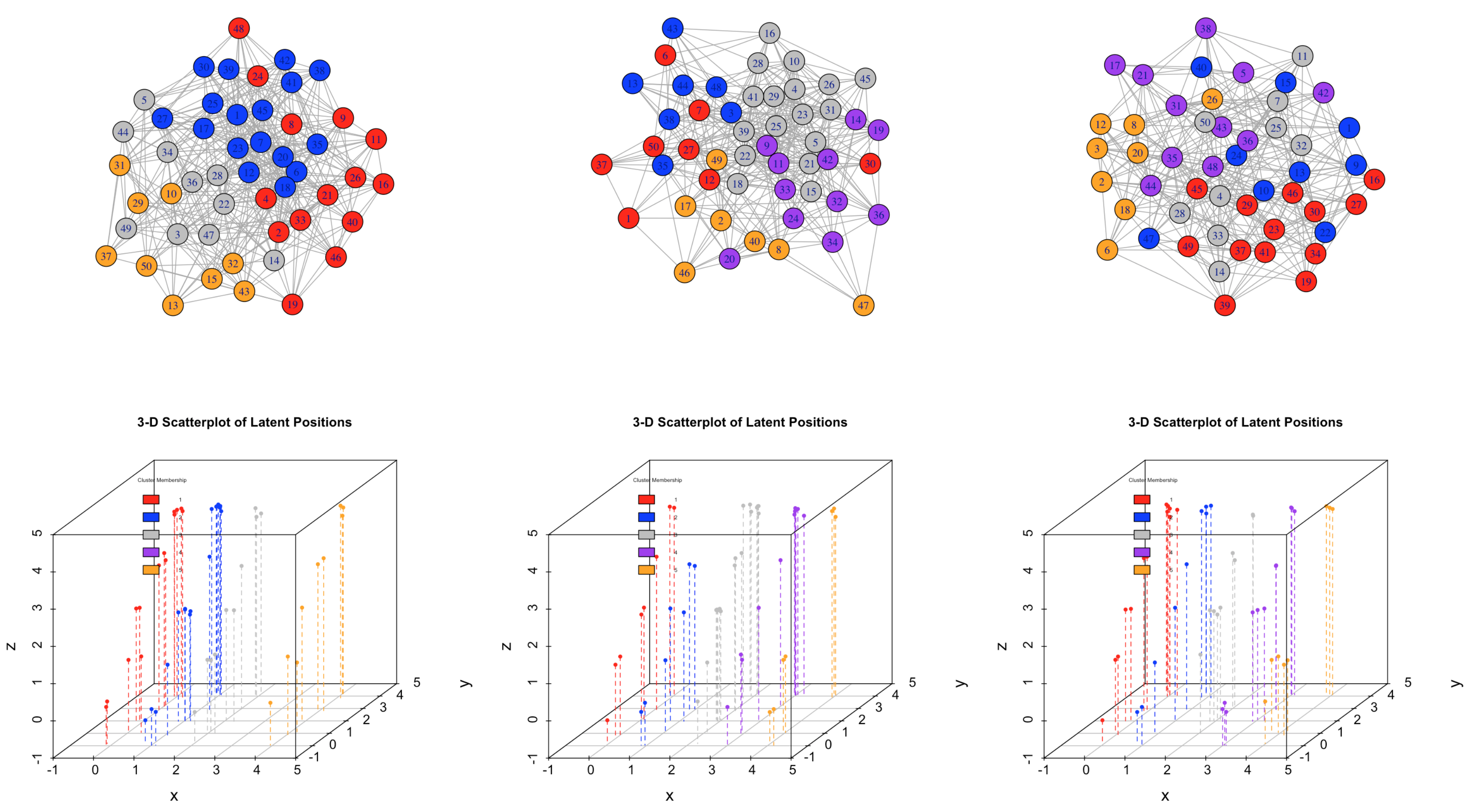

Figure 2: The first row is nodes graph and the second row is the corresponding latent position plot for simulated data (50 nodes, $t = 20, 21, 22$)

References

[1] Mark S Handcock, Adrian E Raftery, and Jeremy M Tantrum. Model-based clustering for social networks. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, 170(2):301–354, 2007.

[2] Peter D Hoff, Adrian E Raftery, and Mark S Handcock. Latent space approaches to social network analysis. *Journal of the american Statistical association*, 97(460):1090–1098, 2002.