

# Comments on binomial graph learning

## Binary graph learning model

### Version 1 (Eigenvector matrix from Graph Laplacian)

Let  $Y_{i,j}$  denote the measurement on the node  $i$  at round  $j$ , where  $j = 1, \dots, M$ , and  $i = 1, \dots, N$ .  $Y_{i,j}$  is a binomial signal that can be 1, or 0. Suppose the signals at round  $j$  denoted by  $Y[,j]$  for all  $N$  nodes are independent of the signals at round  $k$  denoted by  $Y[,k]$ , for  $i \neq k$ . Let  $p_{i,j}$  denote the probability of  $Y_{i,j} = 1$ . Our model assumes

$$\text{logit}(p_{i,j}) = \alpha_j + (\chi h)_i,$$

where  $\chi$  is the eigenvector matrix from Graph Laplacian  $L$ ,  $h$  is a vector of latent factors that governs  $p_{i,j}$  through  $\chi$ , and  $\alpha_j$  is a round specific parameter at round  $j$ .

### Version 2 (Adjacency matrix from Graph)

**Graph** We consider a weighted undirected graph  $G = (V, E)$ , with the vertices set  $V = 1, 2, \dots, N$ , and edge set  $E$ . Let  $\mathbf{A}$  denote the weighted adjacency matrix for the graph  $G$ . In the case of weighted undirected graph,  $\mathbf{A}$  is a square and symmetric matrix.

**Signals on the graph** Let  $Y_{i,j}$  denote the signal on the node  $i$  of graph  $G$  at round  $j$ , where  $j = 1, \dots, M$ , and  $i = 1, \dots, N$ . We assume that  $Y_{i,j}$  is a binary signal that can be 1, or 0. Suppose the signals at round  $j$  denoted by  $Y[,j]$  for all  $N$  nodes are independent of the signals at round (or stratum)  $k$  denoted by  $Y[,k]$ , for  $i \neq k$ , borrowing the idea of conditional logistic regression. Let  $p_{i,j}$  denote the probability of  $Y_{i,j} = 1$ . Our model assumes

$$\text{logit}(p_{i,j}) = \alpha_j + (\mathbf{A}h)_i, \tag{1}$$

where  $\mathbf{A}$  is the adjacency matrix from the graph  $G$ ,  $h$  is a vector of latent factors that governs  $p_{i,j}$  through  $\mathbf{A}$  and assumed to be a standard normal random vector, and  $\alpha_j$  is a round specific parameter at round or stratum  $j$ . In the following, we will use

## Method of Estimation

### Conditional likelihood for one stratum

Let  $O_j$  denote the set of nodes at stratum  $j$  that have observed signals of 1, and let  $Z_j$  denote the set of nodes at stratum  $j$  that have observed signals of 0. Suppose the number of nodes that have signals of 1 at stratum  $j$  is  $k_j$ . The conditional likelihood function based on model (1) for stratum  $j$  of size  $N$ , is

$$\begin{aligned} & P(Y_{lj} = 1 \text{ for } l \in O_j, Y_{mj} = 0 \text{ for } m \in Z_j | \sum_{i=1}^N Y_{ij} = k_j) \\ &= \frac{\exp(\sum_{l \in O_j} \mathbf{A}[l, ]h)}{\sum_{J \in C_{k_j}^N} \exp(\sum_{a \in J} \mathbf{A}[a, ]h)}, \end{aligned} \tag{2}$$

where  $C_{k_j}^N$  is the set of all subsets of size  $k_j$  of the set  $1, 2, \dots, N$ .

### Conditional likelihood function

The conditional likelihood function for all strata is written as

$$L(\mathbf{A}, \beta) = \prod_{j=1}^M \frac{\exp(\sum_{l \in O_j} \mathbf{A}[l, ]h)}{\sum_{J \in C_{k_j}^N} \exp(\sum_{a \in J} \mathbf{A}[a, ]h)}, \quad (3)$$

### Conditional log likelihood function

The full conditional log likelihood function is the sum of the log likelihoods for each stratum and can be written as

$$\log(L(\mathbf{A}, \beta)) = \sum_{j=1}^M \sum_{l \in O_j} \mathbf{A}[l, ]h - \sum_{j=1}^M \log \sum_{J \in C_{k_j}^N} \exp(\sum_{a \in J} \mathbf{A}[a, ]h), \quad (4)$$

### Miscellaneous

Constraints:

- Case  $\chi$ : Here I imagine the constraints are on the Laplacian

$$\begin{aligned} \text{tr}(L) &= N, \\ L_{i,j} &= L_{j,i} \leq 0, \quad i \neq j, \\ L \cdot \mathbf{1} &= \mathbf{0} \end{aligned} \quad (5)$$

- Case with adjacency matrix:

$$\begin{aligned} A_{i,j} &= 0 \text{ if } i = j, \\ A_{i,j} &= A_{j,i} \geq 0, \quad \text{if } i \neq j \end{aligned} \quad (6)$$

- Consider the quadratic form

$$x^T L x = \frac{1}{2} \sum_{(i,j) \in E} w_{i,j} (f(i) - f(j))^2$$

Because  $x$  is a binomial, this expression sums the weights of edges with incident nodes that have differing signal values. Suppose the graph has a constant signal, then  $x^T L x = 0$ . Suppose, we are in the worst case when incident nodes for each edge have a different signal; in this case  $x^T L x = \frac{1}{2} \sum_{(i,j) \in E} w_{i,j}$ . Note, we can bound  $x^T L x$  from below by  $\min w_{i,j}$  in the case were a single edge has incident nodes with differing signals. We can use these bounds to gauge the quality of model.

Sigmoid computation, Bound and Branch method of optimization

hint for factor analysis solution might be useful because there is another object function varimax.

[https://conservancy.umn.edu/bitstream/handle/11299/95957/Choi\\_umn\\_0130E\\_11451.pdf?sequence=1&isAllowed=y](https://conservancy.umn.edu/bitstream/handle/11299/95957/Choi_umn_0130E_11451.pdf?sequence=1&isAllowed=y)

[https://web.stanford.edu/~boyd/papers/pdf/max\\_sum\\_sigmoids.pdf](https://web.stanford.edu/~boyd/papers/pdf/max_sum_sigmoids.pdf)

- complete separation do not exist. observable metric ?, it is a potential new topic to be able to identify whether complete separation exists. possible discuss it in the Discussion and future work.
- While working on this one, you may also consider implementing our regressor paper.

## Model Specification

### Our goal is to estimate the Graph Laplacian

- We do not have to use the eigenvector matrix  $\chi$ . Instead, we may consider adjacency matrix  $A$  in place of  $\chi$ . This is a viable direction.

Maximum Likelihood estimation, Quasi likelihood estimation,