Mostly factor models

Peter Hoff

Statistics, Biostatistics and the CSSS
University of Washington
Adolescent health social network

Data on 82 12th graders from a single high school:

54 boys, 28 girls

\[ \hat{\text{Pr}}(y_{i,j} = 1 \mid \text{same sex}) = 0.077 \]
\[ \hat{\text{Pr}}(y_{i,j} = 1 \mid \text{opposite sex}) = 0.056 \]
Inferential goals in the regression framework

\[ y_{i,j} \text{ measures } i \rightarrow j, \quad x_{i,j} \text{ is a vector of explanatory variables.} \]

\[
Y = \begin{pmatrix}
y_{1,1} & y_{1,2} & y_{1,3} & \text{NA} & y_{1,5} & \cdots \\
y_{2,1} & y_{2,2} & y_{2,3} & y_{2,4} & y_{2,5} & \cdots \\
y_{3,1} & \text{NA} & y_{3,3} & y_{3,4} & \text{NA} & \cdots \\
y_{4,1} & y_{4,2} & y_{4,3} & y_{4,4} & y_{4,5} & \cdots \\
\vdots & \vdots & \vdots & \vdots & \vdots & \ddots 
\end{pmatrix}
X = \begin{pmatrix}
x_{1,1} & x_{1,2} & x_{1,3} & x_{1,4} & x_{1,5} & \cdots \\
x_{2,1} & x_{2,2} & x_{2,3} & x_{2,4} & x_{2,5} & \cdots \\
x_{3,1} & x_{3,2} & x_{3,3} & x_{3,4} & x_{3,5} & \cdots \\
x_{4,1} & x_{4,2} & x_{4,3} & x_{4,4} & x_{4,5} & \cdots \\
\vdots & \vdots & \vdots & \vdots & \vdots & \ddots 
\end{pmatrix}
\]

Consider a basic (generalized) linear model

\[ y_{i,j} \sim \beta' x_{i,j} + e_{i,j} \]

A model can provide

- a measure of the association between \( X \) and \( Y \): \( \hat{\beta}, \text{se}(\hat{\beta}) \)
- imputations of missing observations: \( p(y_{1,4}|Y, X) \)
- a probabilistic description of network features: \( g(\tilde{Y}), \tilde{Y} \sim p(\tilde{Y}|Y, X) \)
Model fit

glm(formula = y ~ x, family = binomial(link = "logit"))

Coefficients:

| Estimate | Std. Error | z value | Pr(>|z|) |
|----------|------------|---------|----------|
| (Intercept) | -2.8332 | 0.1123 | -25.24 | <2e-16 *** |
| x | 0.3471 | 0.1428 | 2.43 | 0.0151 * |

This result says that a model with preferential association is a better description of the data than an i.i.d. binary model.
Nodal heterogeneity and independence assumptions
Neither of these models do well in terms of representing other features of the data - for example, transitivity:

\[ t(\mathbf{Y}) = \sum_{i < j < k} y_{i,j}y_{j,k}y_{k,i} \]
Latent variable models

Deviations from ordinary regression models can be represented as

\[ y_{i,j} \sim \beta' x_{i,j} + e_{i,j} \]

A simple “latent variable” model might include additive node effects:

\[ e_{i,j} = u_i + u_j + \epsilon_{i,j} \quad \Rightarrow \quad y_{i,j} \sim \beta' x_{i,j} + u_i + u_j + \epsilon_{i,j} \]

\{u_1, \ldots, u_n\} represent across-node heterogeneity that is additive on the scale of the regressors. Inclusion of these effects in the model can dramatically improve

- within-sample model fit (measured by \(R^2\), likelihood ratio, BIC, etc.);
- out-of-sample predictive performance (measured by cross-validation).

But this model only captures heterogeneity of outdegree/indegree, and can’t represent more complicated structure, such as clustering, transitivity, etc.
Fit of additive effects model
An eigenvalue decomposition model

\[ E = M + \mathcal{E} \]

\( M \) represents “systematic” patterns and \( \mathcal{E} \) represents “noise”. Every symmetric \( M \) has a representation of the form \( M = U\Lambda U' \) where

- \( U \) is an \( n \times n \) matrix with orthonormal columns
- \( \Lambda \) is an \( n \times n \) diagonal matrix, with elements \( \{\lambda_1, \ldots, \lambda_n\} \)

Many data analysis procedures for symmetric matrix-valued data \( Y \) are related to this decomposition. Given a model of the form

\[ Y = M + \mathcal{E} \]

where \( \mathcal{E} \) is independent noise, the ED provides

**Interpretation:** \( y_{i,j} = u_i'\Lambda u_j + \epsilon_{i,j} \), \( u_i \) and \( u_j \) are the \( i \)th, \( j \)th rows of \( U \)

**Estimation:** \( \hat{M}_R = \hat{U}_[1:R] \hat{\Lambda}[1:R,1:R] \hat{U}'_[1:R] \) if \( M \) is assumed to be of rank \( R \).
Eigenmodel fit

Parameters this model can be fit with the eigenmodel package in R:

\texttt{eigenmodel\_mcmc(Y,X,R=3)}

The latent factors are able to represent the network transitivity.
Underlying structure
Missing variables
Missing variables

The eigenmodel, without having explicit race information, captures a large degree of the racial homophily in friendship:
Factor models for multiway data

Recall the decomposition of a two-way array of rank $R$:

$$m_{i,j} = \mathbf{u}_i' \Lambda \mathbf{u}_j = \sum_{r=1}^{R} u_{i,r} u_{j,r} \lambda_r$$

Now generalize to a three-way array:

$$m_{i,j,k} = \sum_{r=1}^{R} u_{i,r} u_{j,r} w_{k,r} \lambda_r$$

- $\{u_1, \ldots, u_n\}$ represents variation among the nodes;
- $\{w_1, \ldots, w_m\}$ represents variation across the networks.

Consider the $k$th “slab” of $\mathbf{M}$, which is an $n \times n$ matrix:

$$m_{i,j,k} = \sum_{r=1}^{R} u_{i,r} u_{j,r} w_{k,r} \lambda_r$$

$$= \sum_{r=1}^{R} u_{i,r} u_{j,r} \lambda_{k,r} = \mathbf{u}_i' \Lambda_k \mathbf{u}_j \quad \text{where } \lambda_{k,r} \text{ replaces } w_{k,r} \lambda_k$$
Cold War data

Cooperation and conflict data collected on 85 countries every fifth year

How can we numerically describe variability, similarity across $Y_1, \ldots, Y_7$?
International relations data

\[ z_{i,j,t} = u_i^T \Lambda_t u_j + \epsilon_{i,j,t} \]
\[ u_i^T \Lambda_t u_j = \lambda_{1,t} u_{i,1} u_{j,1} + \lambda_{2,t} u_{i,2} u_{j,2} + \lambda_{3,t} u_{i,3} u_{j,3} \]
I struggle with the role of statistics in the analysis of some types network data.

**Static, complete network**
Data are rare and inferential goals are often unclear (to me).

**Dynamic, complete network**
Data are rare but there are some obvious statistical goals (e.g. prediction).

**Partially observed incomplete networks**
More common, modeling is underdeveloped, but often the goals are clear:
- find the links
- find the nodes (RDS)
Find the links: a concrete application?

\( \mathbf{Y} = \{y_{i,j}\} \) is unobserved, observing each \( y_{i,j} \) requires a separate experiment.

Task: find the most number of links in the fewest number of experiments.

“Real” example: Biological networks/protein-protein interaction data.

1. A set of nodes \( \{1, \ldots, n\} \) is identified;
2. A subset of pairs \( S \subset \{(i,j) : 1 \leq i < j \leq n\} \) are tested for interaction;
3. Based on \( \{y_{i,j} : (i,j) \in S\} \), identify new pairs to test.
4. Return to step 1.

Model-based approach:

1. Fit a model \( p(\mathbf{Y}_{\text{full}}|\mathbf{Y}_{\text{obs}}) \);
2. Identify pairs for which \( \mathbb{E}[y_{i,j}|\mathbf{Y}_{\text{obs}}] \) is highest.

I tried this out on Butlands 230×230 matrix of protein-protein interaction data. The model was fit using the eigenmodel package.

\[
\text{eigenmodel\_MCMC(Y.obs,R=3)}
\]
Find the links
Find the links
Find the links
Find the links
Find the links
Find the links
Find the links
Find the links
Find the links
Find the links
Find the links

Factor models for relational data

Real-world network sampling
Find the links
Find the links
Find the links

Factor models for relational data

Real-world network sampling
Find the links

Factor models for relational data
Real-world network sampling
Find the links
Find the links

[Graph showing a network with percent dyads observed on the x-axis and percent links found on the y-axis.]
Find the links
Find the links
Find the links

Factor models for relational data

Real-world network sampling
Find the links
Find the links

90-node social network of high school students
Find the nodes

Goal: Estimate $E[X]$ for some variable $X$ in a hard to reach population.
Problem: Only a few members of the population are known.

RDS procedure:
1. Measure $X$ on a known population member.
2. Have them randomly nominate a member they know to be measured.
3. Repeat.

Ideally, this produces a Markov chain on the social network of population members, with stationary probabilities $p_i \propto d_i$, the degree. $E[X]$ can be estimated with the Horwitz-Thompson estimator:

$$E[\hat{X}] = \frac{\sum_{i=1}^{n} X_i/d_i}{\sum_{i=1}^{n} 1/d_i}$$

This works if
- People select the person they nominate uniformly from their alters.
- The degree they report is the number of people they “would” nominate.
- The process is run long enough for the chain to achieve stationarity.

This approach seems problematic to me for most applications. Can a model-based procedure be implemented?