主文献研读论文汇报

✓ *A block model for node popularity in networks with community structure*

- 作者： Sengupta Srijan & Chen Yuguo
- 期刊：JRSSB
- 发表时间：2018年

✓ *Estimation and clustering in popularity adjusted block model*

- 作者： Noroozi Majid, Ramchandra Rimal, and Marianna Pensky
- 期刊：JRSSB
- 发表时间：2021年

罗伊琳 2020103670 2021/12/22

■ 论文1：Sengupta, S., & Chen, Y. (2018). A block model for node popularity in networks with community structure. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 80(2), 365-386.

◻ 主要内容

论文***提出了流行度调整的块模型（popularity-adjusted block model，PABM）***，更好地对节点在每个社区中的流行度进行建模，在PABM下建立了作为社区检测的最大化目标的**似然模块度**，并证明了似然模块度和节点流行度及模型参数估计的一致性。

PABM是随机块模型（stochastic block model，SBM）的一个拓展模型，实际上就是设定节点i和节点j之间的连接概率为：

$$i\text{到}j\text{所在社区的流行度参数}\times j\text{到}i\text{所在社区的流行度参数}$$

并基于泊松分布进行统计推断。对该“似然模块度”的优化可以用变分方法，Kernighan-Lin型算法，伪似然算法等方法进行，本文使用的是Le等（2016）[1]提出的EPs算法求解。

[1] Le,C. M., Levina, E. andVershynin, R. (2016) Optimization via low-rank approximation for community detection in networks. Ann. Statist., 44, 373–400.

◻ Review：SBM & DCBM

◆ 基于模型的社区检测/网络聚类

➢ 设无向网络邻接矩阵的元素服从贝努利分布，p_{ij}为节点i和j的连接概率 $A_{ij} \sim \mathrm{Ber}(p_{ij})$,

- SBM假设节点i和节点j的连接只与它们所在的社区c_i和c_j有关，即

$$p_{ij} = P_{c_i c_j},$$

- DCBM则考虑了节点本身的度，在SBM基础上增加了度参数，节点i和节点j的连接不仅和它们所在的社区c_i和c_j有关，还跟它们本身的度有关，即

$$p_{ij} = \theta_i \omega_{c_i c_j} \theta_j,$$

其中θ_i和θ_j为度系数，ω_{rs}为社区r和s的连接系数。DCBM存在可识别性问题，因此加上限制（每个社区里的节点度系数之和为1）： $\sum_{i \in \mathcal{N}_a} \theta_i = 1, \forall a = 1, \ldots, K,$

◻ 节点在社区中的流行度

- 节点在社区中的流行度是和网络社区结构紧密相关的一个特征，节点i在社区r中的流行度定义为<u>节点到该社区中所有节点的边数（的期望）</u>。
- 注意到，对于同一社区，不同节点的流行度不同；同一节点在不同社区里的流行度也不同。可能节点i在社区r中的流行度比节点j大，但在社区s中相反。

◻ SBM和DCBM无法很好地刻画节点流行度

- SBM为了让同一社区的节点具有相同的行为，会对节点流行度产生限制；
- DCBM对每个节点引入了度参数，会一致地放大或降低节点的流行度，因此如果同一社区内的节点i和j，在一个社区里i比j流行度更高，那么在所有社区里i的流行度都会比j高。
- SBM和DCBM里计算节点流行度可以发现，<u>SBM里同一社区的节点流行度是相同的</u>，而<u>DCBM中同一社区节点，流行度和度存在比例关系</u>，度越大的节点流行度就越大。

Table 1. Illustrative nodes for political blogs, with popularities fited by the DCBM in parentheses

Name	*Observed (fitted by DCBM)*			
	Community	*Liberal popularity*	*Conservative popularity*	*Degree*
andrewsullivan.com	Conservative	58 (10)	85 (133)	143 (142)
blogsforbush.com	Conservative	5 (21)	296 (278)	301 (299)
democraticunderground.com	Liberal	59 (85)	34 (7)	93 (93)
liberaloasis.com	Liberal	169 (157)	2 (13)	171 (170)

- 行表示四个样本博客，列分别表示从属社区，分别在两个社区的流行度，节点度。括号外是观测值，括号内是DCBM拟合值。
- 第一行和第二行（属于同一社区）：2的节点度约为1的两倍，DCBM拟合得到的流行度，在两个社区里都是2约为1的两倍，但实际上在liberal里1流行度远大于2，在conservative里则相反。

Table 2. Illustrative nodes for British MPs†

Name	*Observed (fitted by DCBM)*			
	Community	*Conservative popularity*	*Labour popularity*	*Degree*
Zac Goldsmith	Conservative	46 (62)	25 (8)	71 (70)
Matt Hancock	Conservative	68 (62)	3 (8)	71 (70)
Seema Malhotra	Labour	0 (4)	88 (84)	88 (88)
Ian Austin	Labour	11 (3)	76 (83)	87 (87)

†Identities were looked up by using tweeterid.com.

- 第一行和第二行（属于同一社区）：两个节点实际上在两个社区里的流行度相差较大，但由于二者的度相同，DCBM可以很好地拟合节点度，但得到两个节点在两个社区的流行度也是相同的。与实际情况不符。

◻ 流行度调整的随机块模型（Popularity-adjusted block model, PABM）

- 在PABM中，节点i和j分别从属于社区c_i和c_j，这两个节点的连接取决于“i在c_j中的流行度”和“j在c_i中的流行度”，即

$$p_{ij} = \lambda_{ic_j}\lambda_{jc_i},$$

其中λ_{ir}表示节点i在社区r中的流行度参数。

- PABM也需要给定一个限制条件以解决可识别性问题，令r中所有节点到s的流行度参数之和与s中所有节点到r的流行度参数之和相同，即：

$$\Lambda_{rs} = \Lambda_{sr} \qquad \Lambda_{rs} := \sum_{j \in \mathcal{N}_r} \lambda_{js}.$$

- 设M_{ir}为节点i到社区r所有社区的连接数，在PABM下，节点流行度计算为

$$\mu_{ir} = E[M_{ir}] = \sum_{j \in N_r} p_{ij} = \lambda_{ir}\Lambda_{rc_i}$$

◻ PABM下的似然模块度

- 这里考虑似然函数时，不用伯努利分布而改用**泊松分布**作为节点连接的假设（即考虑多边情况）。DCBM提出的时候使用的也是泊松分布的假设，Zhao等（2012）说明了使用泊松分布在实际理论证明中带来了很大的便利，而产生的误差成本是很小的。
- 计算似然函数：

$$L=\left\{\prod_{i<j}\frac{(\lambda_{ic_j}\lambda_{jc_i})^{A_{ij}}}{A_{ij}!}\exp(-\lambda_{ic_j}\lambda_{jc_i})\right\}\prod_i\frac{(\frac{1}{2}\lambda_{ic_i}^2)^{A_{ii}/2}}{(A_{ii}/2)!}\exp\left(-\frac{1}{2}\lambda_{ic_i}^2\right).$$

- 对数似然：

$$\begin{aligned}l&=\sum_{i<j}A_{ij}\log(\lambda_{ic_j})+\sum_{i<j}A_{ij}\log(\lambda_{jc_i})+\sum_i A_{ii}\log(\lambda_{ic_i})-\left(\sum_{i<j}\lambda_{ic_j}\lambda_{jc_i}+\frac{1}{2}\sum_i\lambda_{ic_i}^2\right)\\&=\sum_i\sum_r M_{ir}\log(\lambda_{ir})-\frac{1}{2}\sum_i\sum_j\lambda_{ic_j}\lambda_{jc_i},\end{aligned}$$

- 对数似然对λ_{ir}求导=0，得到 $\hat{\lambda}_{ir}=\frac{M_{ir}}{\sqrt{O_{rc_i}}}.$

这里M_{ir}为节点i到社区r所有节点的连接数，O_{rs}表示社区r和s之间的连接数。

◻ PABM下的似然模块度

- 把λ_{ir}带入对数似然函数中进行化简，最后得到只关于节点分配c的轮廓似然函数，由于最后只需要求解使得该轮廓似然最大的节点分配，和之前模块度函数的本质是类似的，它也可以看作是某种模块度，称为“似然模块度”：

$$Q(c) = 2\sum_i \sum_r M_{ir} \log\left(\frac{M_{ir}}{\sqrt{O_{rc_i}}}\right) = 2\sum_i \sum_r M_{ir} \log(M_{ir}) - \sum_r \sum_s O_{rs} \log(O_{rs}).$$

- 在候选的节点分配策略中，选择使得似然模块度最大的节点分配，之后代入到参数估计的公式中，可得到流行度的参数估计结果。

一致性结果

- 引入稀疏性参数，考虑PABM 模块度的scaled version

$$Q(e)=\frac{2}{n^2\rho_n}\sum_i\sum_r M_{ir}\log\left(\frac{M_{ir}}{\sqrt{O_{e_i r}}}\right)=\frac{1}{n^2\rho_n}\left\{2\sum_i\sum_r M_{ir}\log(M_{ir})-\sum_r\sum_s O_{rs}\log(O_{rs})\right\}.$$

- 似然模块度的population version

$$\tilde{Q}(e)=\frac{2}{n^2\rho_n}\sum_i\sum_r \mu_{ir}(e)\log\left\{\frac{\mu_{ir}(e)}{\sqrt{o_{re_i}(e)}}\right\},$$

- 假设条件：

Assumption 1 (number of communities). The number of communities K is fixed and known. The true assignment c as well as all candidate assignments e have exactly K non-empty communities.

Assumption 2 (sparsity parameter). $\rho_n=\omega\{\log(n)/\sqrt{n}\}$, i.e. $n\rho_n^2/\log^2(n)\to\infty$ as $n\to\infty$.

Assumption 3 (identifiability). For any two communities $1\leqslant a,b\leqslant K$, $\Lambda_{ab}=\Lambda_{ba}$, where Λ_{ab} is defined in equation (6).

Assumption 4 (detectability). For any two distinct communities $1\leqslant a\neq b\leqslant K$ and any two nodes $j_1\in\mathcal{N}_a$, $j_2\in\mathcal{N}_b$, the set $\{p_{ij_1}/p_{ij_2}\}_{i=1}^n$ assumes at least $K+1$ distinct values.

一致性结果

- 社区检测的一致性：对样本似然模块度求最大值对应的节点分配，就会和真实分配情况很接近。

Theorem 1 (community detection). Under assumptions 1–4,

$$\xi_n(\hat{c}) \xrightarrow{P} 0$$

where $\hat{c} = \arg\max_{e} Q(e)$, $\xi_n(e) = \min_{e' \in \Pi(e)} \frac{1}{n} \sum_{i=1}^{n} I(e'_i \neq c_i)$,

- 参数估计的一致性：在求解最大似然模块度得到节点分配 $\hat{c}$ 的基础上，带入公式得到流行度参数和流行度的估计，以一定速度依概率收敛到真实参数值。

Theorem 2 (parameter estimation). Under assumptions 1–6,

$$\Delta_n(\hat{c}) \xrightarrow{P} 0$$

where $\hat{c}$ is defined in equation (11) and Δ_n is defined in equation (13).

where $\Delta_n(\hat{c}) = \frac{1}{\sqrt{n}} \|\hat{\lambda}_{n\times K} - \lambda_{n\times K}\|_F$.

◻ Simulation

- 对于该目标函数的求解文中提到可以使用变分方法，Kernighan-Lin型算法，伪似然算法等。本文采用的Le等（2016）[1]提出的EPs算法。
- EPs算法本质上是寻找候选分配集合，缩小优化范围。以K=2为例，找到邻接矩阵的前2个主特征向量张成的空间，将$[-1,1]^n$投影到空间里，投影构成的凸包的极值点所对应的标签子集即为候选分配集合。

In this paper, we use the so-called EPs algorithm, which is a state of the art low dimensional optimization algorithm proposed by Le *et al.* (2016). Briefly, for $K=2$ the algorithm computes the two leading eigenvectors of the adjacency matrix A and finds the candidate assignments that are associated with the EPs of the projection of the cube $[-1,1]^n$ onto the space that is spanned by the two leading eigenvectors of A. Let $\mathcal{B}_{\text{can}}$ be the set of all such candidate assignments. The modularity function Q (or Q_{DC}) is evaluated on all assignments $b \in \mathcal{B}_{\text{can}}$, and the best assignment is defined as the maximizer of Q (or Q_{DC}) over $\mathcal{B}_{\text{can}}$, i.e. $\hat{c} := \arg\max_{b \in \mathcal{B}_{\text{can}}} Q(b)$ for the PABM, and $\hat{c} := \arg\max_{b \in \mathcal{B}_{\text{can}}} Q_{\text{DC}}(b)$ for the DCBM. Some advantages of EPs over the competing

[1] Le,C. M., Levina, E. andVershynin, R. (2016) Optimization via low-rank approximation for community detection in networks. Ann. Statist., 44, 373–400.

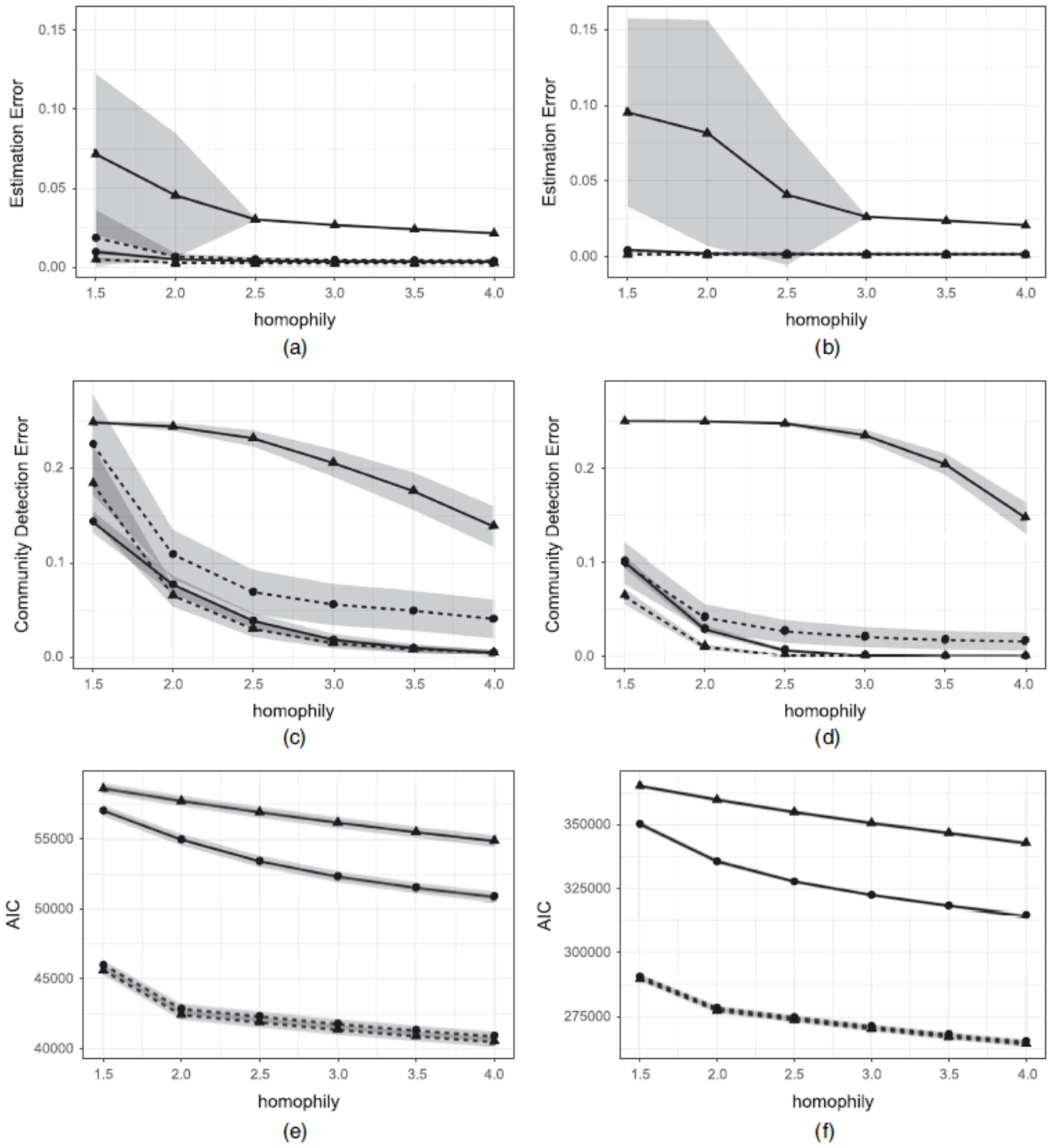
◻ Simulation

- 分别在PABM和DCBM的设定下生成网络，分别在PABM和DCBM的似然模块度下进行网络社区检测和节点流行度的估计。（二分类$K = 2$）
  - PABM设定：$\lambda_{ir} = \alpha\sqrt{h/(1+h)}\ when\ r = c_i, and\ \lambda_{ir} = \beta\sqrt{1/(1+h)}\ when\ r \neq c_i$，第一节类节点$\alpha = 0.8\ and\ \beta = 0.2$，第二类节点$\alpha = 0.2\ and\ \beta = 0.8$，$h$是homophily factor，$h$越大则网络社区结构越强。
  - DCBM设定： $P(\theta_i = 1.6) = P(\theta_i = 0.4) = 0.5$ $\omega = \frac{0.2}{h+1}\begin{pmatrix} h & 1 \\ 1 & h \end{pmatrix}$
  - 节点数：$n = 400\ (n_1 = 200\ and\ n_2 = 200)$，$n = 1000\ (n_1 = 500\ and\ n_2 = 500)$
- 比较：节点流行度的估计表现；社区检测准确性的表现
- 流行度估计误差 $\frac{1}{n\mathcal{E}}\sum_{i=1}^{n}\sum_{r=1}^{K}\{\hat{\mu}_{ir}(\hat{c}) - \mu_{ir}(c)\}^2,$

◻ Simulation

- ✓ **实线**：由PABM生成网络；**虚线**：由DCBM生成网络
- ✓ **圆点**：求解PABM似然模块度；**三角点**：求解DCBM似然模块度
- ✓ **左列**：$n = 400$；**右列**：$n = 1000$
- ✓ **上**：流行度估计误差；**中**：社区检测误差；**下**：AIC

- ➢ 对于由PABM生成的网络，基于DCBM模块度的表现很差，体现了DCBM无法很好刻画节点流行度。PABM模块度一致地比DCBM模块度好。
- ➢ 对于由DCBM生成的网络，PABM模块度增加了很多不必要的参数，会增加噪音影响效果，但流行度估计和AIC表现来看，PABM只是略差于DCBM，还是comparable的。



◻ Data analysis

- ✓ The political blogs network：美国政治博客网络，节点博客之间通过超链接相连
- ✓ The British MPs Twitter network：英国议员推特网络，节点议员之间通过转发推特相连
- ✓ The DBLP network：计算机类的文献检索数据库系统，节点作者通过参加了同一个会议相连

➢ goodness-of-fit measures

$$F_1 = \frac{1}{2E} \sum_{i=1}^{n} \sum_{r=1}^{K} \{\hat{\mu}_{ir}(\hat{c}) - M_{ir}(c)\}^2 ,$$

$$F_2 = \frac{1}{2E} \sum_{i=1}^{n} \sum_{r=1}^{K} \{\hat{\mu}_{ir}(c) - M_{ir}(c)\}^2 ,$$

Where F_1 measures the overall goodness of fit originating from **community detection and model fit**, whereas F_2 measures the community-coreected goodness of fit that originates **purely from the model fit**.

◻ Data analysis

Table 3. Community detection error rates†

Network	*Nodes*	*Error from unregularized EPs (%)*		*Error from regularized EPs (%)*	
		PABM	*DCBM*	*PABM*	*DCBM*
Political blogs	1222	4.99 (61)	5.07 (62)	4.99 (61)	5.40 (66)
British MPs	329	0.30 (1)	0.61 (2)	0.00 (0)	0.61 (2)
DBLP	2203	2.81 (62)	4.77 (105)	2.81 (62)	5.17 (114)

†The numbers of misclustered nodes are given in parentheses.

Table 4. Goodness-of-fit measures for node popularity

Network	*F_1 from unregularized EPs*		*F_1 from regularized EPs*		*F_2*	
	PABM	*DCBM*	*PABM*	*DCBM*	*PABM*	*DCBM*
Political blogs	0.06	1.157	0.057	1.155	0.002	1.883
British MPs	0.002	0.282	0.002	0.282	0.002	0.284
DBLP	2.255	52.430	2.255	52.430	0.000	61.425

- PABM在社区检测误差和拟合优度上表现都要优于DCBM，尤其是拟合优度（考虑了节点流行度）

Table 5. Illustrative nodes for political blogs (regularized EPs)

Name	*Observed (fitted by PABM)*			
	Community	*Liberal Popularity*	*Conservative Popularity*	*Degree*
andrewsullivan.com	Conservative	58 (59)	85 (84)	143 (143)
blogsforbush.com	Conservative	5 (6)	296 (292)	301 (298)
democraticunderground.com	Liberal	59 (62)	34 (31)	93 (93)
liberaloasis.com	Liberal	169 (169)	2 (1)	171 (170)

Table 6. Illustrative nodes for British MPs†

Name	*Observed (fitted by PABM)*			
	Community	*Conservative popularity*	*Labour popularity*	*Degree*
Zac Goldsmith	Conservative	46 (46)	25 (25)	71 (71)
Matt Hancock	Conservative	68 (67)	3 (3)	71 (70)
Seema Malhotra	Labour	0 (0)	88 (88)	88 (88)
Ian Austin	Labour	11 (11)	76 (76)	87 (87)

†Identities of the nodes of this network were looked up by using tweeterid.com.

- PABM不仅能很好地刻画度，还能准确地表现实际流行度。（与最开头DCBM拟合结果对比）

■ 论文2： Noroozi, M., Rimal, R., & Pensky, M. (2021). Estimation and clustering in popularity adjusted block model. Journal of the Royal Statistical Society: Series B (Statistical Methodology).

◻ 主要内容

本文从另一个角度对Sengupta和Chen(2018)[1]提出的PABM进行了拓展理解，对PABM的公式进行了转换，将概率矩阵写成了**分块矩阵**的形式，每个块都是一个秩一矩阵；利用**Bregman散度**提出了PABM社区检测的一般目标，在F范数下可以写出已知社区数K的目标函数，并且通过加入对K的惩罚项可以处理K未知的情况；根据一定假设下该分块矩阵的性质，可以将**稀疏子空间聚类（SSC）算法**运用到目标函数的求解中。

[1] Sengupta, S., & Chen, Y. (2018). A block model for node popularity in networks with community structure. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 80(2), 365-386.

◻ 从分块角度理解PABM

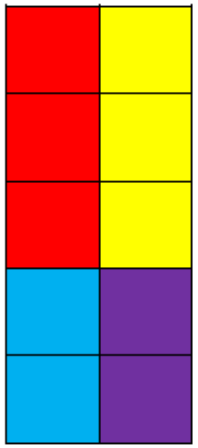
- 对于节点间连接概率$P_{i,j} = V_{i,c_j}V_{j,c_i}$的PABM，为了更好地理解这个模型，在已知社区数$K$和节点分配矩阵$Z$（$Z \in R^{n\times K}$）时，考虑对概率矩阵$P$按照节点分配形式进行重排，得到$P(Z,K)$，前$n_1$行和列表示第一个社区里的节点，以此类推，最后$n_K$行和列表示第$K$个社区的节点。
- $P(Z,K)$是一个分块矩阵，(k,l)块$P^{(k,l)}(Z,K)$是$n_k \times n_l$的矩阵，每个元素$P_{i,j}^{(k,l)} = V_{i_k,l}V_{j_l,k}$，容易看出，$P^{(k,l)}(Z,K)$是一个秩一矩阵，可以表示成一组唯一的一维向量乘积。

◆ 记$\Lambda^{(k,l)}$为n_k维向量，分量$\Lambda_i^{(k,l)} = V_{i_k,l}$为社区$k$的第$i$个元素在社区$l$中的流行度参数，则有如下关系

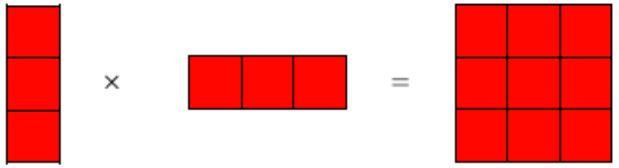
$$P^{(k,l)}(Z,K) = \Lambda^{(k,l)}\,[\Lambda^{(l,k)}]^T.$$

从分块角度理解PABM

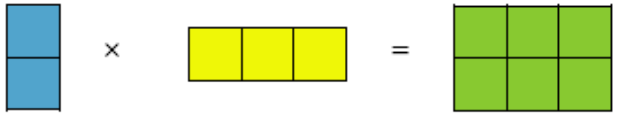
$\Lambda =$



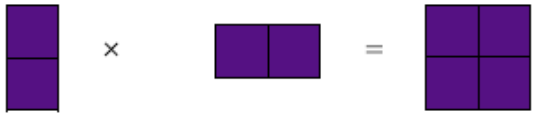
$P^{(1,1)}(Z,K) = \Lambda^{(1,1)}\left[\Lambda^{(1,1)}\right]^T =$



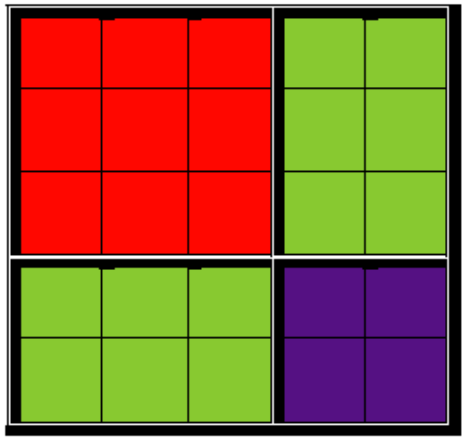
$P^{(2,1)}(Z,K) = \Lambda^{(2,1)}\left[\Lambda^{(1,2)}\right]^T =$



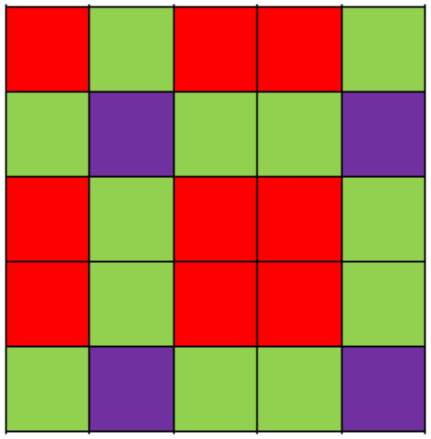
$P^{(2,2)}(Z,K) = \Lambda^{(2,2)}\left[\Lambda^{(2,2)}\right]^T =$



$P(Z,K) =$



$P =$



◻ 从分块角度理解PABM

◆ 整个重排后的概率矩阵就有如下的分块形式

$$P(Z,K)=\begin{bmatrix} \Lambda^{(1,1)}(\Lambda^{(1,1)})^T & \Lambda^{(1,2)}(\Lambda^{(2,1)})^T & \cdots & \Lambda^{(1,K)}(\Lambda^{(K,1)})^T \\ \Lambda^{(2,1)}(\Lambda^{(1,2)})^T & \Lambda^{(2,2)}(\Lambda^{(2,2)})^T & \cdots & \Lambda^{(2,K)}(\Lambda^{(K,2)})^T \\ \vdots & \vdots & \cdots & \vdots \\ \Lambda^{(K,1)}(\Lambda^{(1,K)})^T & \Lambda^{(K,2)}(\Lambda^{(2,K)})^T & \cdots & \Lambda^{(K,K)}(\Lambda^{(K,K)})^T \end{bmatrix}$$

- 这说明矩阵$P(Z,K)$是由很多个秩一块矩阵组成的。本文最主要的落脚点就是认识到<u>PABM的概率矩阵是由一些秩一矩阵组合而成</u>，这是Sengupta和Chen(2018)没有意识到的。这可以帮助我们从另一个角度对PABM进行估计和聚类，建立基于F范数最小化的聚类目标。
- 之后也发现在可检测条件下，<u>任一社区对应的概率矩阵的列都在一个K维的子空间里</u>，并且这个社区的子空间与其他社区所在子空间不同。本文基于这个结论引入了SSC进行社区检测。

估计和聚类的过程推导

- 当真实社区数K_*已知时，给定节点分配Z，可以用一个置换矩阵$\mathcal{P}_{Z,K_*}$对邻接矩阵A进行重排，即$A(Z,K_*)=\mathcal{P}_{Z,K_*}^T A\mathcal{P}_{Z,K_*}$，回忆概率矩阵分块

$$P^{(k,l)}(Z,K)=\Lambda^{(k,l)}\,[\Lambda^{(l,k)}]^T.$$

- 很直观地，如果K_*已知，想要得到Z_*和P_*的估计，我们可以在分块意义下最小化$A(Z,K_*)$和$\Lambda^{(k,l)}\left[\Lambda^{(l\,k)}\right]^T$构成的概率矩阵之间的某个散度度量。
- 考虑<u>Bregman散度</u>（F是连续可微严格凸函数）： $D_F(x,y)=F(x)-F(y)-\langle\nabla F(y),x-y\rangle$
- 目标函数就可以一般化为最小化$A(Z,K_*)$和$P(Z,K_*)$之间的Bregman散度

- 不难发现，取 $F(x)=\sum_i(x_i\ln x_i-x_i)$. 时，就可以推导得到Sengupta和Chen(2018)得到的基于泊松分布的似然模块度。

估计和聚类的过程推导

- 这里考虑取$F(x) = ||x||^2$，对应矩阵的F范数，得到给定K下的优化目标

$$(\hat{\Lambda}, \hat{Z}) \in \underset{\Lambda, Z}{\operatorname{argmin}} \left\{ \sum_{k,l=1}^{K} \| A^{(k,l)}(Z,K) - \Lambda^{(k,l)} [\Lambda^{(l,k)}]^T \|_F^2 \right\} \quad \text{s.t.} \quad A(Z,K) = \mathcal{P}_{Z,K}^T A \mathcal{P}_{Z,K}$$

- 这个目标对Λ的求解是需要之前Sengupta和Chen(2018)提出的可识别性条件的，但是我们知道了它的每个块都是一个秩一矩阵，就可以转化为一个求秩一矩阵$\Theta^{(k,l)}$的目标。另外针对K未知的情况，再加上一个对K的惩罚项$\mathrm{Pen}(\mathrm{n}, \mathrm{K})$，得到完整的目标函数为

$$(\hat{\Theta}, \hat{Z}, \hat{K}) \in \underset{\Theta, Z, K}{\operatorname{argmin}} \left\{ \sum_{k,l=1}^{K} \| A^{(k,l)}(Z,K) - \Theta^{(k,l)} \|_F^2 + \mathrm{Pen}(n,K) \right\}$$
$$\text{s.t.} \quad A(Z,K) = \mathcal{P}_{Z,K}^T A \mathcal{P}_{Z,K}, \quad \operatorname{rank}(\Theta^{(k,l)}) = 1; \quad k,l = 1,2,\cdots,K.$$

估计和聚类的过程推导

- 假设$\hat{Z}$和$\hat{K}$已知，用$A^{(k,l)}(\hat{Z},\hat{K})$的秩一估计作为$\hat{\Theta}^{(k,l)}$，即

$$\hat{\Theta}^{(k,l)}(\hat{Z},\hat{K}) = \Pi_{\hat{u},\hat{v}}\left(A^{(k,l)}(\hat{Z},\hat{K})\right) = \hat{\sigma}_1^{(k,l)}\hat{u}^{(k,l)}(\hat{Z},\hat{K})\left(\hat{v}^{(k,l)}(\hat{Z},\hat{K})\right)^T,$$

- 上式代入原目标，假设$\hat{K}$已知时，求解Z，即

$$\hat{Z}_K \in \underset{Z\in\mathcal{M}_{n,K}}{\operatorname{argmin}}\left\{\sum_{k,l=1}^{K}\|A^{(k,l)}(Z,K) - \Pi_{\hat{u},\hat{v}}\left(A^{(k,l)}(Z,K)\right)\|_F^2\right\} \tag{12}$$

- 再代回原目标，得到一个只含K的函数，对K进行搜索找到最优社区数

$$\hat{K} \in \underset{K}{\operatorname{argmin}}\left\{\sum_{k,l=1}^{K}\|A^{(k,l)}(\hat{Z}_K,K) - \Pi_{\hat{u},\hat{v}}\left(A^{(k,l)}(\hat{Z}_K,K)\right)\|_F^2 + \mathrm{Pen}(n,K)\right\}. \tag{13}$$

- 选择惩罚项为

$$\mathrm{Pen}(n,K) = H_1 nK + H_2 K^2 \ln n + H_3 n \ln K,$$

◻ 稀疏子空间聚类（Sparse Subspace Clustering, SSC）

- 子空间聚类就是为对处于不同子空间的点的分离而设计的，令$\{\mathrm{X}_j \in R^D\}_{j=1}^n$是一组点，来自$K$个线性或仿射子空间$\{S_i\}_{i=1}^K$，每个子空间维度是$d_i = \dim(S_i)$，以线性子空间为例，即

$$S_i = \{\boldsymbol{x} \in \mathbb{R}^D : \boldsymbol{x} = \boldsymbol{U}_i \boldsymbol{y}\}, \quad i = 1, \ldots, K$$

其中U_i为S_i的基，y为x在S_i中的低维表示。

- 子空间聚类的目的就是找到子空间数目K，每个子空间的维数$\{d_i\}_{i=1}^k$和基$\{U_i\}_{i=1}^k$，以及每个点的分配。子空间聚类算法包括：代数方法，迭代方法，基于谱聚类的方法。这里考虑的是<u>基于谱聚类</u>的算法。
- 谱聚类算法依靠的是构建一个亲和（相似）矩阵(affinity matrix)，里面的元素是两点间的某种距离度量。比如SBM直接用邻接矩阵A，DCBM则通过对A进行一些规范化处理。
- 但是子空间聚类就不太适用，因为有可能两个点离得很近，但是在两个子空间里。

◻ 稀疏子空间聚类（Sparse Subspace Clustering, SSC）

- 一个解决的办法就是构建亲和矩阵的时候利用"点的自表示"，希望这个点能够更可能被在它自己子空间里的其他点表示，而不是来自其他子空间的点。
- 构建方法包括低秩表示(low-rank representation)和稀疏子空间聚类(SSC)。
- 本文用的是SSC，可以利用到之前得到的结论：给定K时，P_*实际上是处在K个子空间的并集里，每个子空间至多是K维的。
- 如果P_*已知，权重矩阵W就可以在将每个点写成所有其他点的稀疏线性组合的条件下，通过最小化非零分量的数量（0范数）得到，即

$$\min_{W_j} \|W_j\|_0 \quad \text{s.t } (P_*)_j = \sum_{k \neq j} W_{kj} (P_*)_k$$

- 在数据含噪声的情况下，上面的条件太过于精确，可以引入调整参数$\gamma > 0$，得到

$$\widehat{W}_j \in \underset{W_j}{\operatorname{argmin}} \left\{ \|W_j\|_0 + \gamma \|A_j - AW_j\|_2^2 \quad \text{s.t.} \quad W_{jj} = 0 \right\}, \quad j = 1, \ldots, n,$$

◻ 稀疏子空间聚类（Sparse Subspace Clustering, SSC）

◆ 前面的优化目标可以改写成

$$\widehat{W}_j \in \underset{W_j}{\operatorname{argmin}} \left\{ \|A_j - AW_j\|_2^2 \quad \text{s.t.} \quad \|W_j\|_0 \le L, \quad W_{jj} = 0 \right\}, \quad j = 1, \ldots, n, \tag{32}$$

对于PABM的求解，取L =K。

➢ 利用<u>正交匹配追踪(orthogonal matching pursuit, OMP)算法</u>可以解决(32)这个问题。

➢ 给定$\widehat{W}$，就可以根据权重矩阵的对称形式得到SSC的亲和（相似）矩阵

$$S = |\widehat{W}| + |\widehat{W}^T| \tag{33}$$

◆ <u>**SSC的基本流程**</u>为两步：1. 估计权重矩阵W（用OMP算法求解(32)）

2. 在亲和矩阵S上运用谱聚类（33）

◻ Simulations

- 在Sengupta和Chen(2018)里的模拟中，只考虑了二分类问题，并且设定的人工网络虽然是从PABM中生成的，但是太过于平衡了，其实谱聚类也能得到较准确的结果。因此这里考虑生成连接概率多样化且符合PABM的人工网络，并且考虑K取多种值。
- 具体生成过程这里不展开。

◆ 对PABM的求解步骤：1. 用OMP算法优化(32)，得到权重矩阵W的估计；

（<u>全程K是给定的</u>）　2. 对相似矩阵S应用谱聚类算法，得到聚类矩阵Z的估计；

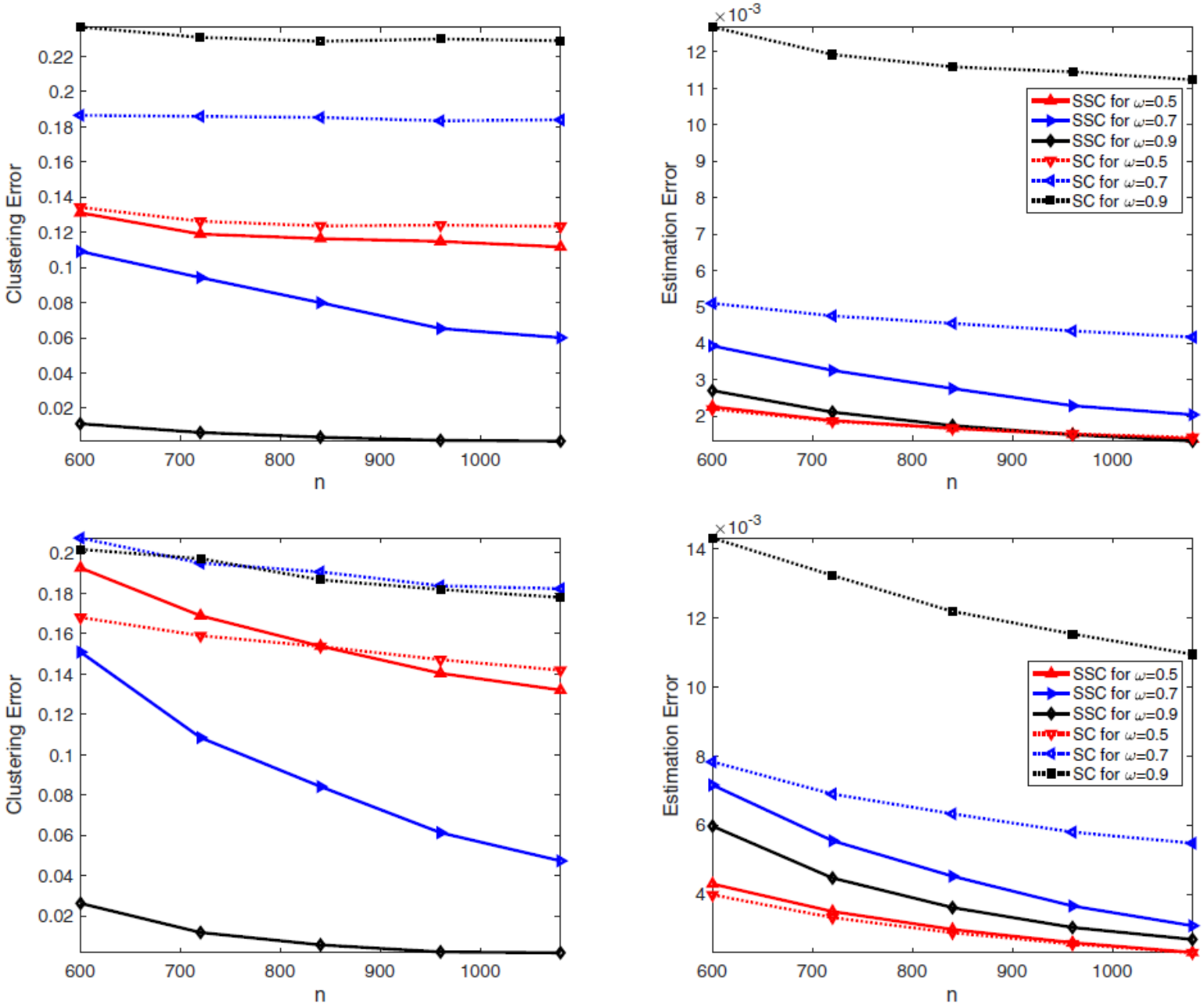
3. 给定$\hat{Z}$，可以得到A分块形式，根据每个块的秩一估计得到$\widehat{\Theta}$;

4. 最后把$\widehat{\Theta}$用置换矩阵还原到原来的样本顺序，得到$\hat{P}$.

- 尝试不同的K取值，找到目标函数最小的那个作为最优社区数。

◻ Simulations

✓ 随着ω的增加，SSC比SC的效果更准确。因为SSC更加适合处理更加多样化的连接概率。



- 左边是聚类估计误差，右边是概率矩阵估计误差；
- 实线为SSC，虚线为SC；
- ω 越大表示生成的网络异质性（不均匀）越强；
- $K = 4$（上），$K = 8$（下）。

◻ Simulations

✓ 出现频率最高的都是真实的社区数，说明选择K的方法是合理的。

- 每个模型设定下生成50个网络，统计网络选择的最优社区数出现的频率。

TABLE 1 The relative of the estimators $\hat{K}$ of K_* for K_* ranging from 3 to 6, $n = 420$ and $n = 840$ and $\omega = 0.5$, 0.7 and 0.9

		n = 420			n = 840		
K_*	$\hat{K}$	$\omega = 0.5$	$\omega = 0.7$	$\omega = 0.9$	$\omega = 0.5$	$\omega=0.7$	$\omega = 0.9$
3	2	0	0	0	0	0	0
	3	**0.76**	**0.80**	**0.90**	**0.52**	**0.60**	**0.80**
	4	0.24	0.16	0.10	0.36	0.26	0.16
	5	0	0.04	0	0.12	0.14	0.02
	6	0	0	0	0	0	0.02
4	2	0	0	0	0	0	0
	3	0.06	0.14	0	0.02	0.02	0
	4	**0.64**	**0.66**	**0.96**	**0.56**	**0.64**	**0.76**
	5	0.28	0.16	0.04	0.30	0.26	0.22
	6	0.02	0.04	0	0.12	0.08	0.02
5	2	0	0.02	0	0	0	0
	3	0.02	0	0.02	0	0	0
	4	0.14	0.16	0.04	0.04	0.04	0
	5	**0.64**	**0.66**	**0.82**	**0.78**	**0.68**	**0.90**
	6	0.20	0.16	0.12	0.18	0.28	0.10
6	2	0	0.04	0	0	0	0
	3	0.06	0.18	0.02	0	0	0
	4	0.18	0.22	0.02	0	0	0
	5	0.28	0.22	0.08	0.12	0.16	0.10
	6	**0.48**	**0.34**	**0.88**	**0.88**	**0.84**	**0.90**

The probabilities for the true values of K are given in bold.

◻ Real data examples

- 社交网络往往具有比较强的同配行为，这种现象可能和人类倾向于建立强关联的趋势有关。
- Sengupta和Chen(2018)采用的政治博客、英国推特等网络都比较接近分块对角的邻接矩阵，实际上SC在这些网络上的表现也会很好。
- PABM能够对于更加多样的网络提供精确的描述，尤其是出现在生物科学中的网络。
- 这里考虑了两个网络：蝴蝶相似性网络和人脑功能网络
  - 蝴蝶相似性网络：节点为几百种蝴蝶，边是通过细粒度图得到的表型相似性的0-1值，总共有4个种群大类。
  - 人脑功能网络：节点为人脑划分出的几百个区域，边是各区域之间的某种功能性连接，根据以往的研究有6个功能大类。

◻ Real data examples

➢ 在真实数据上分别采用SSC，SC和加权k-median算法，和真实结果进行比较，SSC的调整兰德系数(ARI)最高。

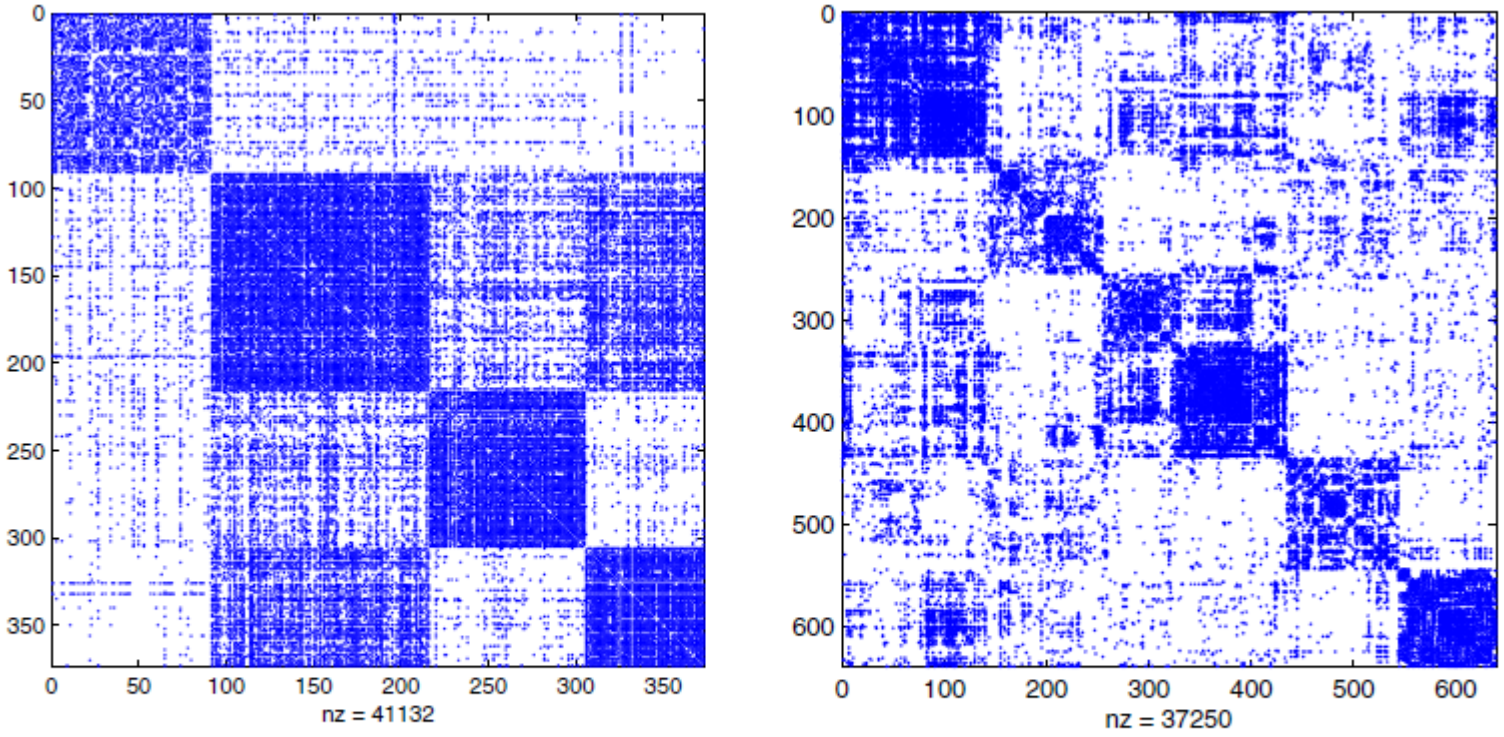


FIGURE 5 Adjacency matrices of the butterfly similarity network with 41,132 nonzero entries and 4 clusters (left) and the brain network with 37,250 nonzero entries and 6 clusters (right)

Fig5. 估计得到的邻接矩阵的分块形式
（左：蝴蝶网络，右：人脑网络）