Community detection in networks with node features

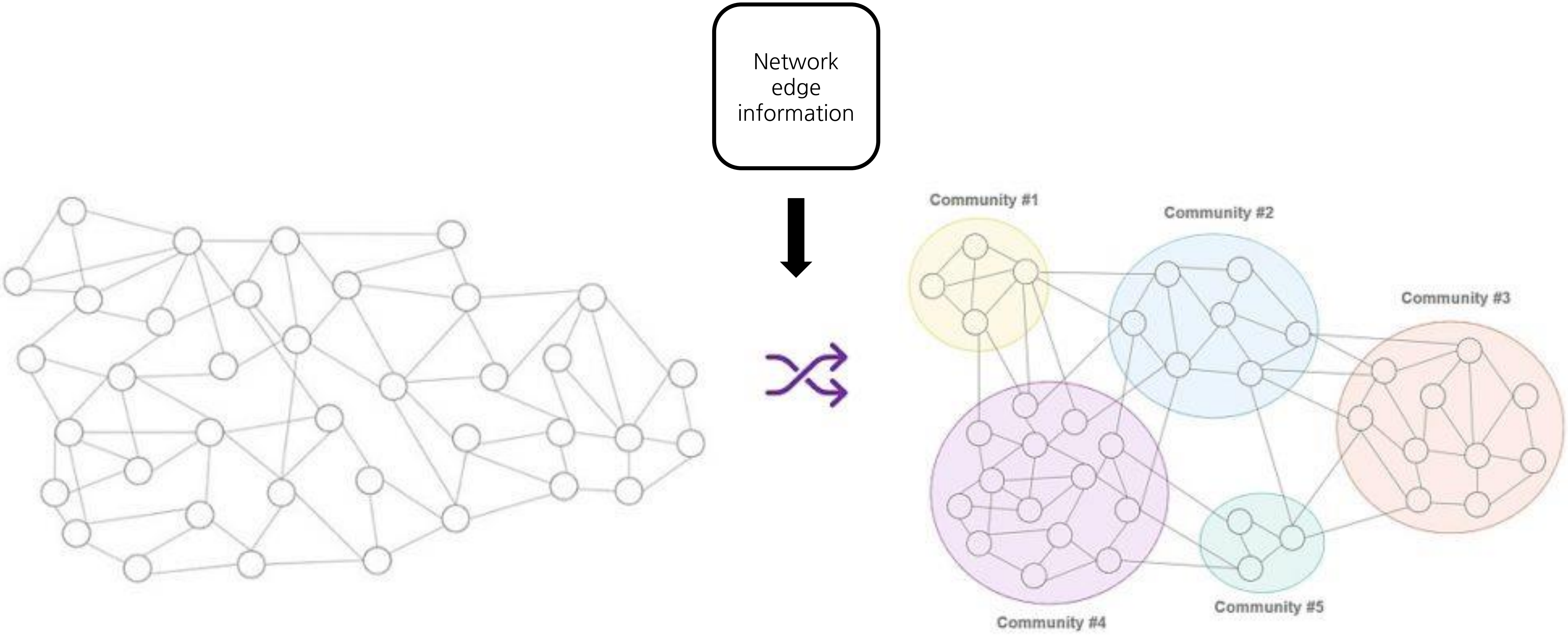
Mose Park

Department of Statistical Data Science
University of Seoul

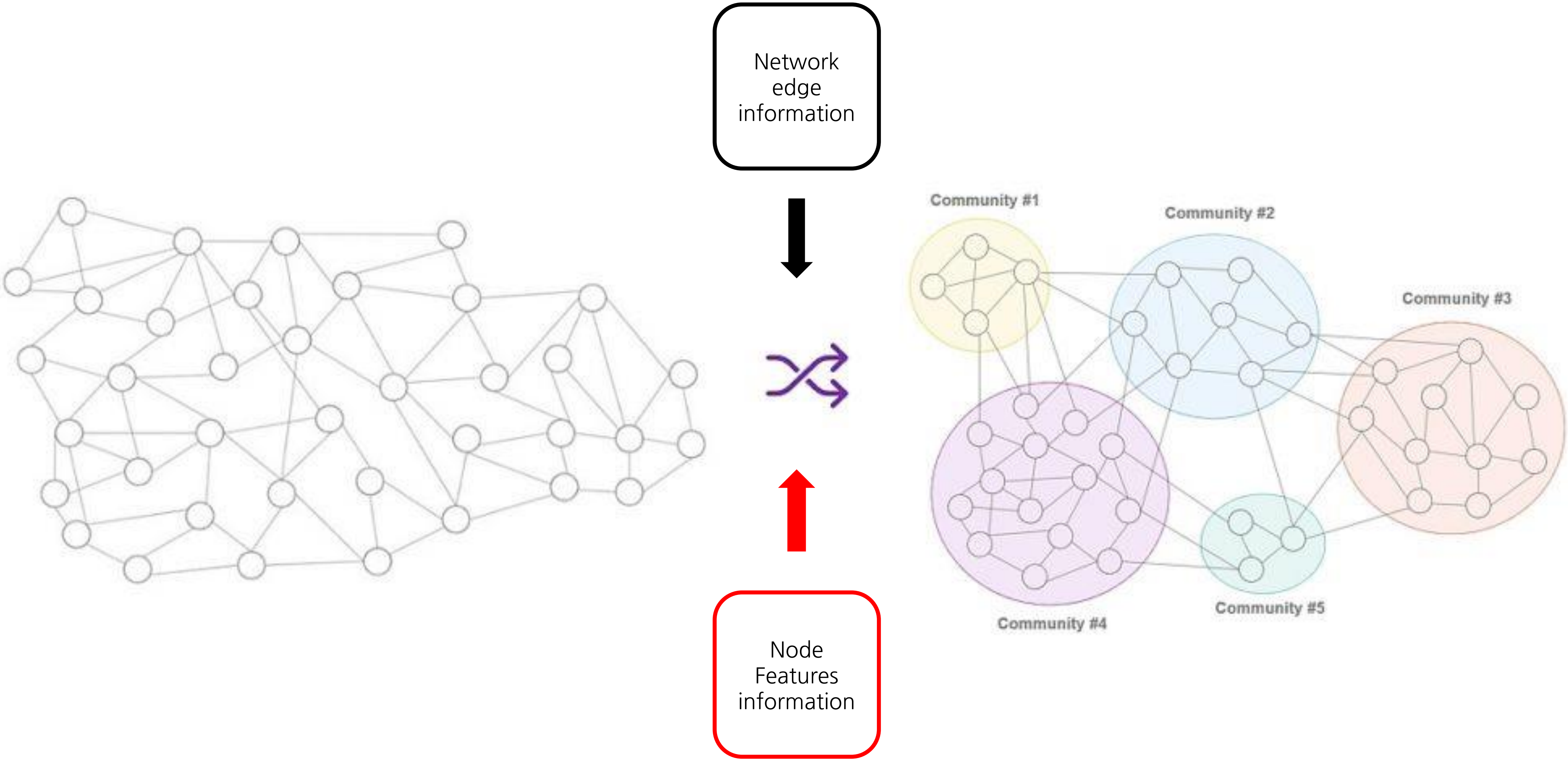
January 5, 2024

1. Build Up

Problem

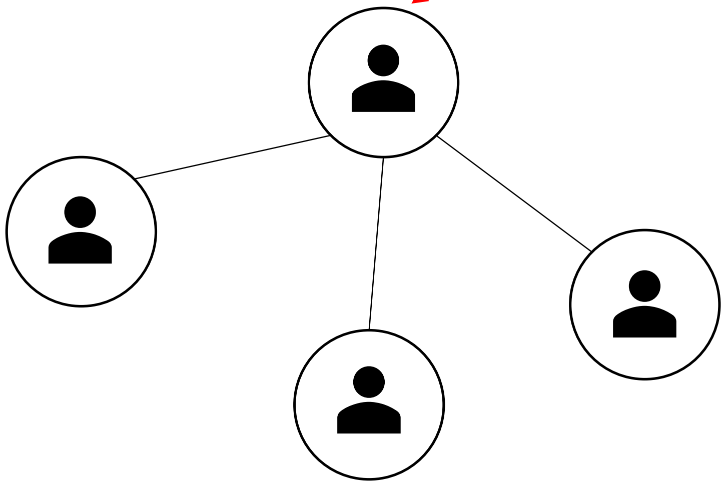

Network edge information
Community #1
Community #2
Community #3
Community #4
Community #5

Solution


Network edge information
Node Features information
Community #1
Community #2
Community #3
Community #4
Community #5

Node features, attributes

성별, 직업, 취미 등
커뮤니티 형성에 영향



Model Example

Based on probabilistic

- SBM
- Latent Factor Model

Based on node features

- CESNA
- BAGC

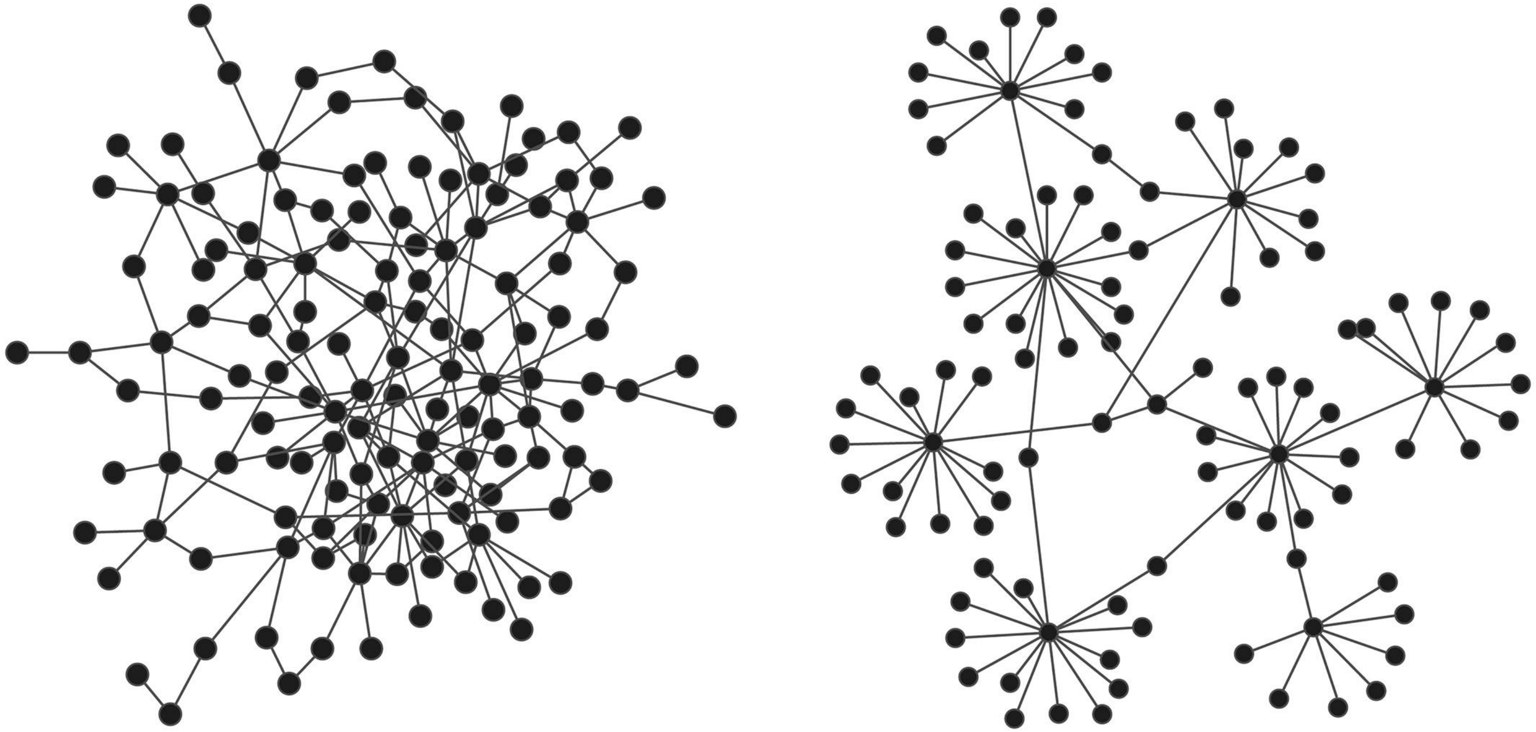
Based on network structure

- Modularity
- Spectral Clustering etc.

3154 intro 하단

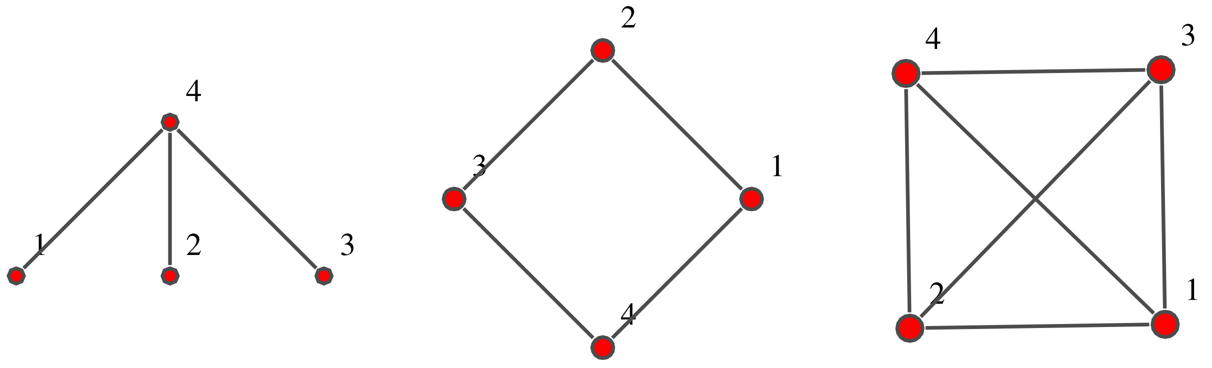
2. Joint Community Detection Criterion

Assortative



(A) Assortative (B) Disassortative

Adjacency matrix



$$\begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix} \qquad \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix} \qquad \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}$$

Community detection criterion

$$R(e;\ \alpha)\ =\sum_{k=1}^{K}\frac{1}{|\mathcal{E}_k|^{\alpha}}\sum_{i,j\ \in\mathcal{E}_k}A_{ij} \qquad \textbf{(2.1)}$$

- What does the equation mean?
- Why maximization?

$$R(e;\ \alpha) = \sum_{k=1}^{K} \frac{1}{|\mathcal{E}_k|^{\alpha}} \sum_{i,j\, \in \mathcal{E}_k} A_{ij} \qquad \textbf{(2.1)}$$

Node label vector

Community k

Adj. Matrix

$|\mathcal{E}_k|$: The # of nodes in community k

α : nodes hyperparameter

$$R(e;\ \alpha) = \sum_{k=1}^{K} \frac{1}{|\mathcal{E}_k|^{\alpha}} \sum_{i,j\ \in \mathcal{E}_k} A_{ij} \qquad \textbf{(2.1)}$$

What label would node “ i ” be? ⇔ What community does node “ i ” belong to?

Why maximizaiton? ⇔ within community edges connection ↑↑↑

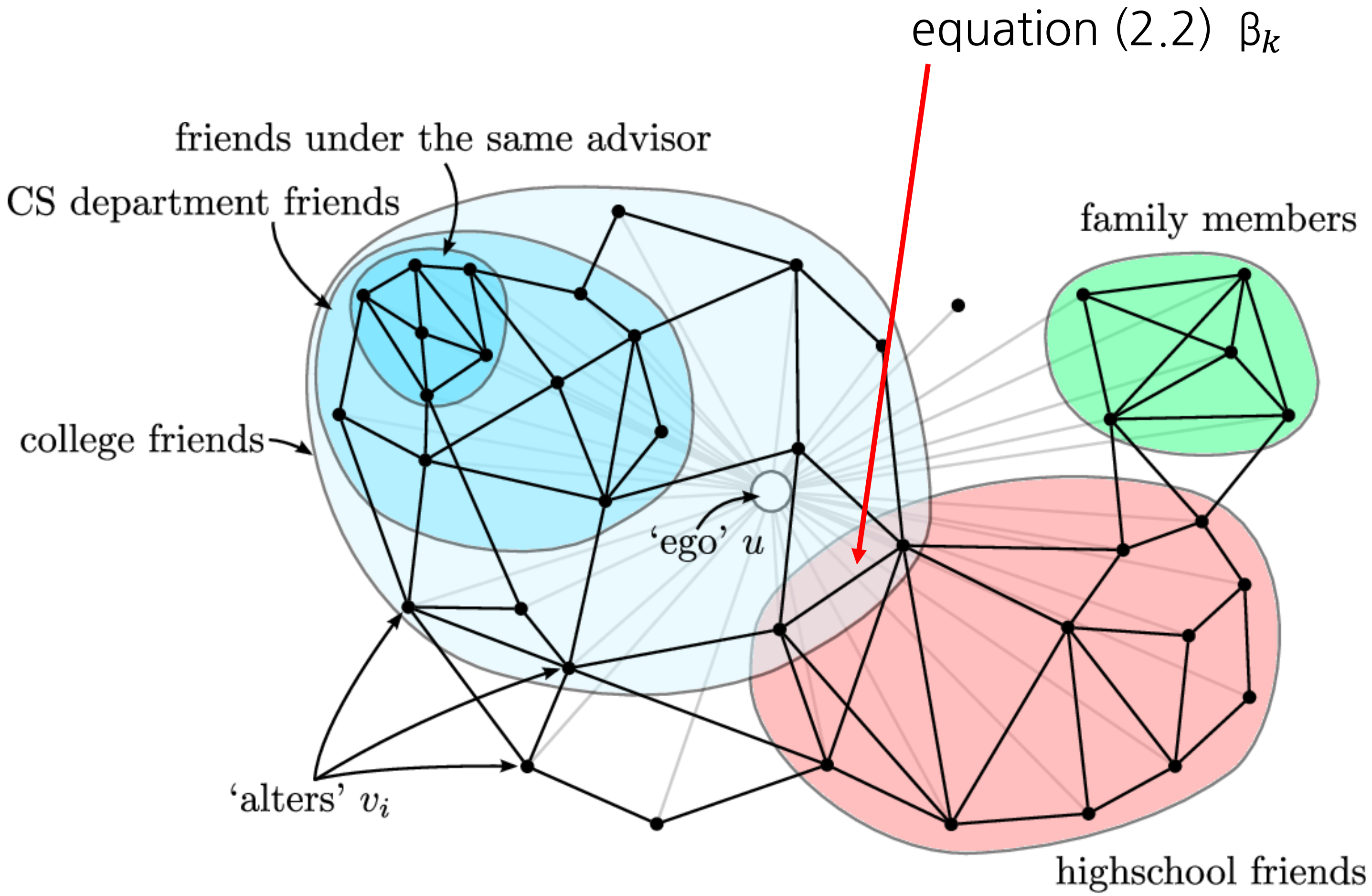
Joint community detection criterion

added in edge weight

$$R(e, \beta; w_n) = \sum_{k=1}^{K} \frac{1}{|\varepsilon_k|^{\alpha}} \sum_{i,j \in \varepsilon_k} A_{ij} W(\underline{f_i, f_j}, \beta_k; \underline{w_n}) \qquad (2.2)$$

balance of adj matrix and features set information

features vector of node i, j


equation (2.2) β_k
friends under the same advisor
CS department friends
family members
college friends
'ego' u
'alters' v_i
highschool friends

3. Estimation

Fixed weights, label assignments

$$\frac{S_{kk} + 2\overbrace{S_{i \leftrightarrow k}}^{\text{The sum of edges weights between node i and community k.}}}{(|\varepsilon_k| + 1)^{\alpha}} + \frac{S_{ll}}{|\varepsilon_l|^{\alpha}} > \frac{S_{kk}}{|\varepsilon_k|^{\alpha}} + \frac{S_{ll} + 2S_{i \leftrightarrow l}}{(|\varepsilon_l| + 1)^{\alpha}} \qquad \textbf{(3.1)}$$

i $\in$ community k

switching label

i $\notin$ community l

$$\frac{S_{i\leftrightarrow k}}{|\varepsilon_k|} * \left(\frac{|\varepsilon_k|}{|\varepsilon_l|}\right)^{1-\alpha} > \frac{S_{i\leftrightarrow l}}{|\varepsilon_l|} \qquad (3.1)$$

local update! computing is simple.

$\alpha = 1$,

$$\frac{S_{i\leftrightarrow k}}{|\varepsilon_k|} > \frac{S_{i\leftrightarrow l}}{|\varepsilon_l|}$$

Avg(weights) of all edges connecting node i to the community k, l.

Fixed label, optimize weights

$$R(e, \beta; w_n) = \sum_{k=1}^{K} \frac{1}{|\varepsilon_k|^{\alpha}} \sum_{i,j \in \varepsilon_k} A_{ij} W(f_i, f_j, \beta_k; w_n)$$

Why?

$$\Rightarrow \quad R(e, \beta; w_n) = \sum_{k=1}^{K} \frac{1}{|\varepsilon_k|^{\alpha}} \sum_{i,j \in \varepsilon_k} A_{ij} W\left(f_i, f_j, \beta_k; w_n\right) - \lambda \left|\left|\beta_k\right|\right|_1$$

$\because$ Corr(β_k, feature similarity) → (O)

$\therefore$ Tend to feature similarity ↑↑ → β_k ↑↑

Algorithm A.2

Algorithm 1 JCDC algorithm

label e

Input: $A \in \mathbb{R}^{n \times n}$, $\phi \in \mathbb{R}^{n \times n \times p}$, α, w_n, λ, m, m_u, m_v

for $t = 1$ to m **do**

  for $u = 1$ to m_u **do**

    for $i = 1$ to n **do** Update:

      $i \leftarrow \arg\max_k \frac{S_{i \leftrightarrow k}}{|\mathcal{E}_k|^\alpha}$

O(mm$_u$n), 약 O(n^3)

β_k

  for $v = 1$ to m_v **do**

    for $k = 1$ to K **do** Update:

      $\beta_k \leftarrow \arg\max_{\beta_k} \frac{1}{|\mathcal{E}_k|^\alpha} \sum_{i,j \in \mathcal{E}_k} A_{ij} \left(w_n - e^{-\langle \phi_{ij}, \beta_k \rangle} \right) - \lambda \|\beta_k\|_1$

Why Consistency?

- Condition 1 - guarantees proportons of nodes do not vanish
- Condition 2 - enforces assortativity

Theorem 1. *Under conditions 1 and 2, if $n\rho_n \to \infty$, and the parameter α satisfies*

$$\frac{\max_{k,l} 2(K-1)P_{kl}}{\min_{k,l}(P_{kk}, P_{ll})} \leq \alpha \leq 1 \tag{4.1}$$

then we have, for any fixed $\delta > 0$,

$$\mathbb{P}\left(|\arg\max_{e \in \mathcal{E}^{\pi_0}} R(e;\alpha) - c| > \delta \right) \to 0 \tag{4.2}$$

5. Simulation studies

r, μ and NMI

- r ↑↑↑ = out-in probability ratio

= “Between” > “within” = community detection ↓

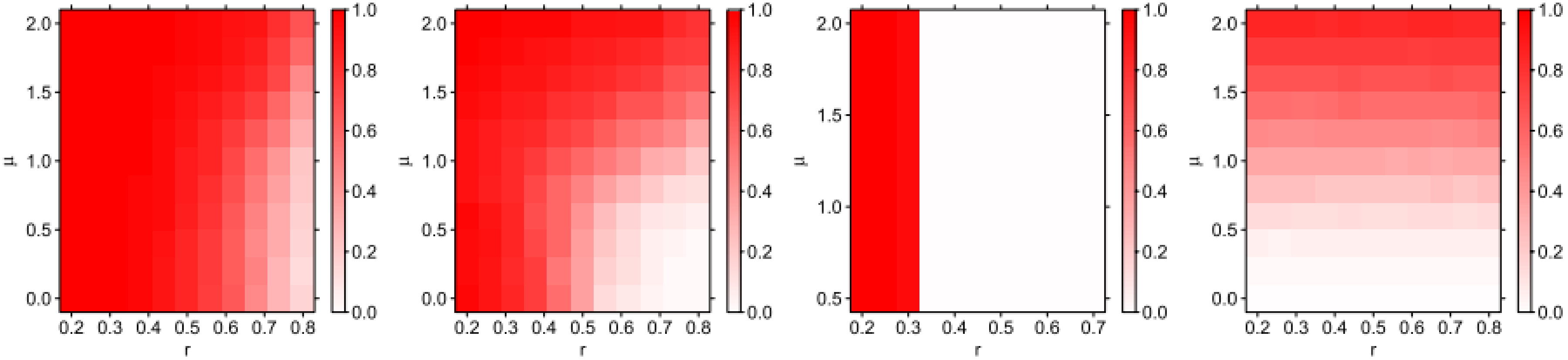
= harder problem

- μ ↑↑↑ = feature signal strength ↑ = easier problem

- NMI → 1 = Predict community structure ↛ True structure
- NMI → 0 = Predict community structure → True structure

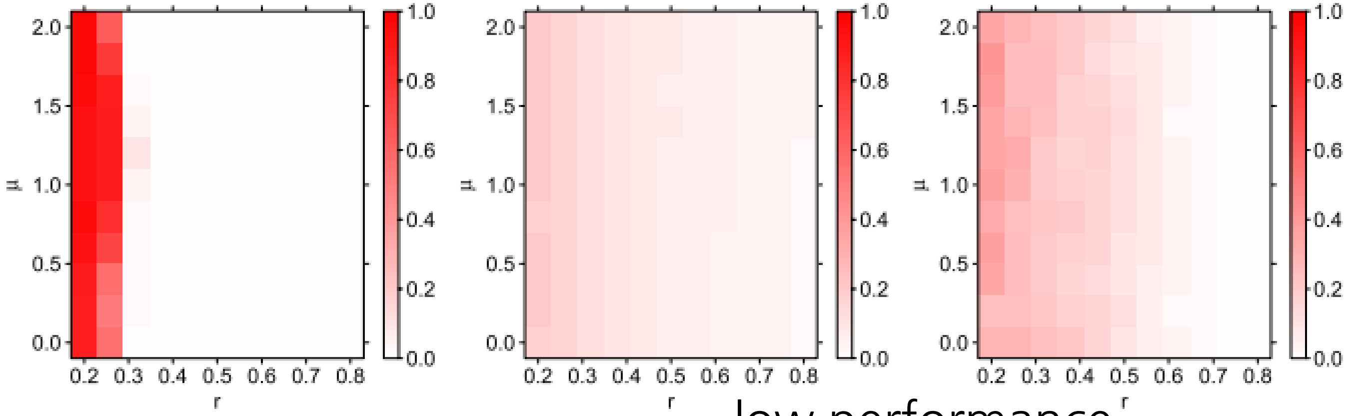
Fig.1, Heat map

only network information, ∴ only r



(a) JCDC, $w = 5$ (b) JCDC, $w = 1.5$ (c) SC (d) KM

only features information
∴ only μ

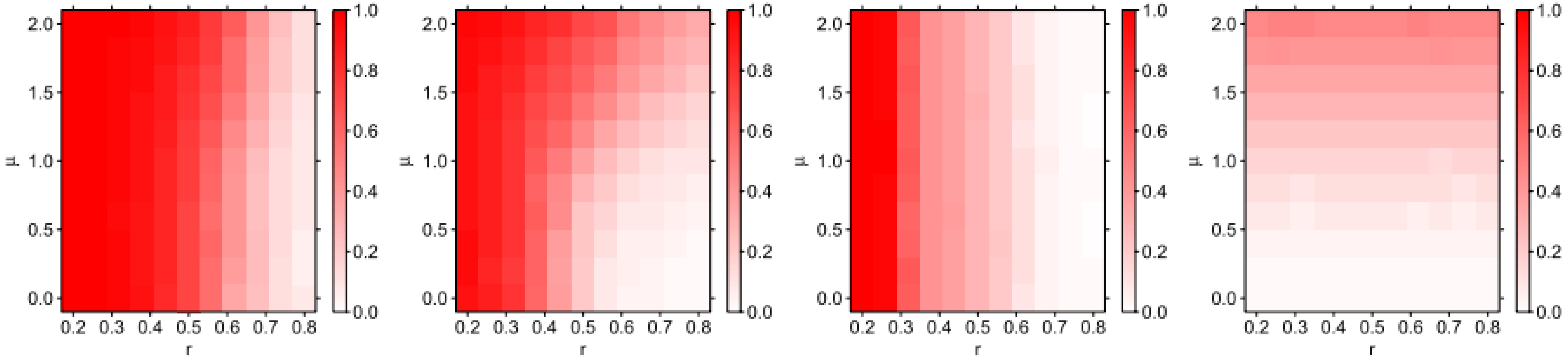


low performance

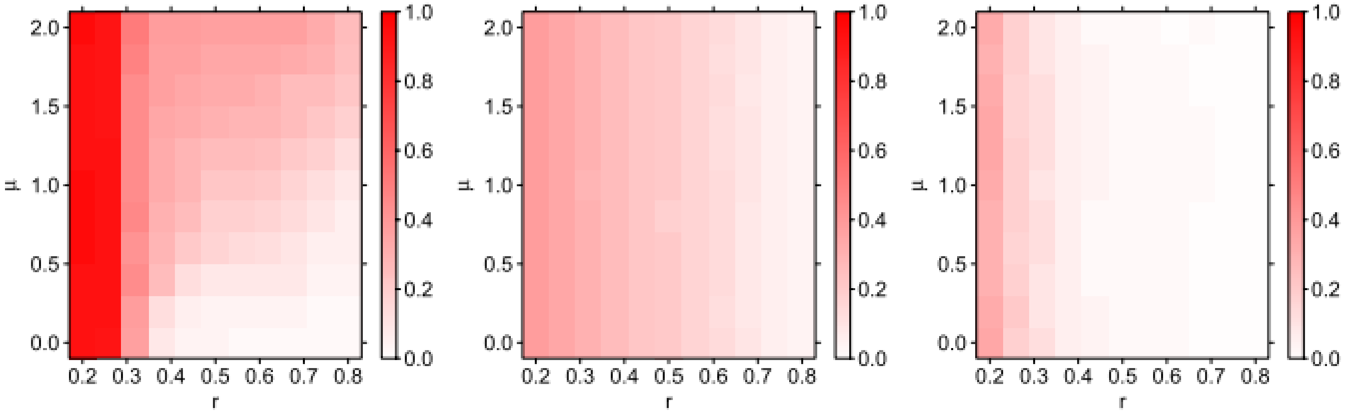
(e) CASC (f) CESNA (g) BAGC

K = 2 , n1 = 100, n2= 50

Fig.2



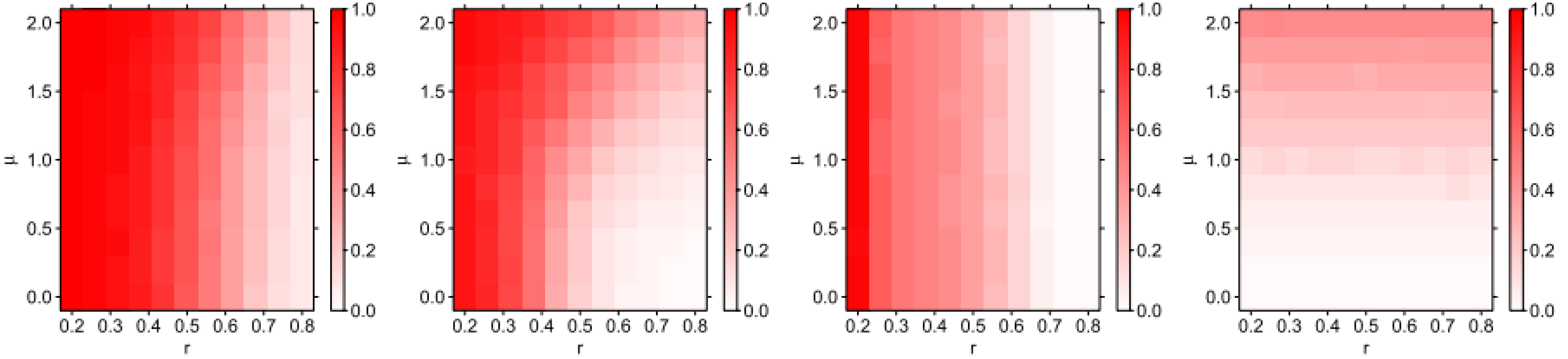
(a) JCDC, $w = 5$ (b) JCDC, $w = 1.5$ (c) SC (d) KM



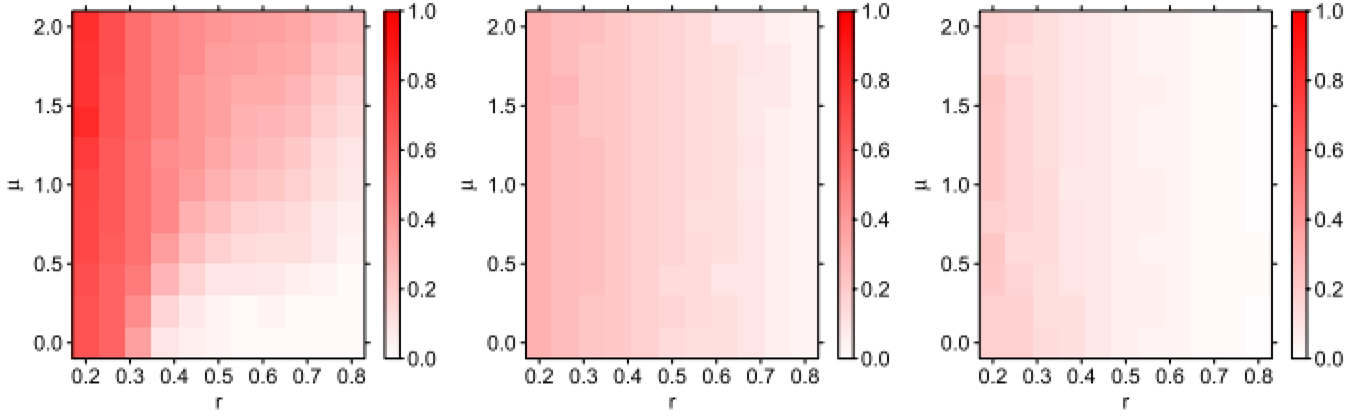
(e) CASC (f) CESNA (g) BAGC

K = 3 , n1 = n2 = n3 = 50

Fig.3

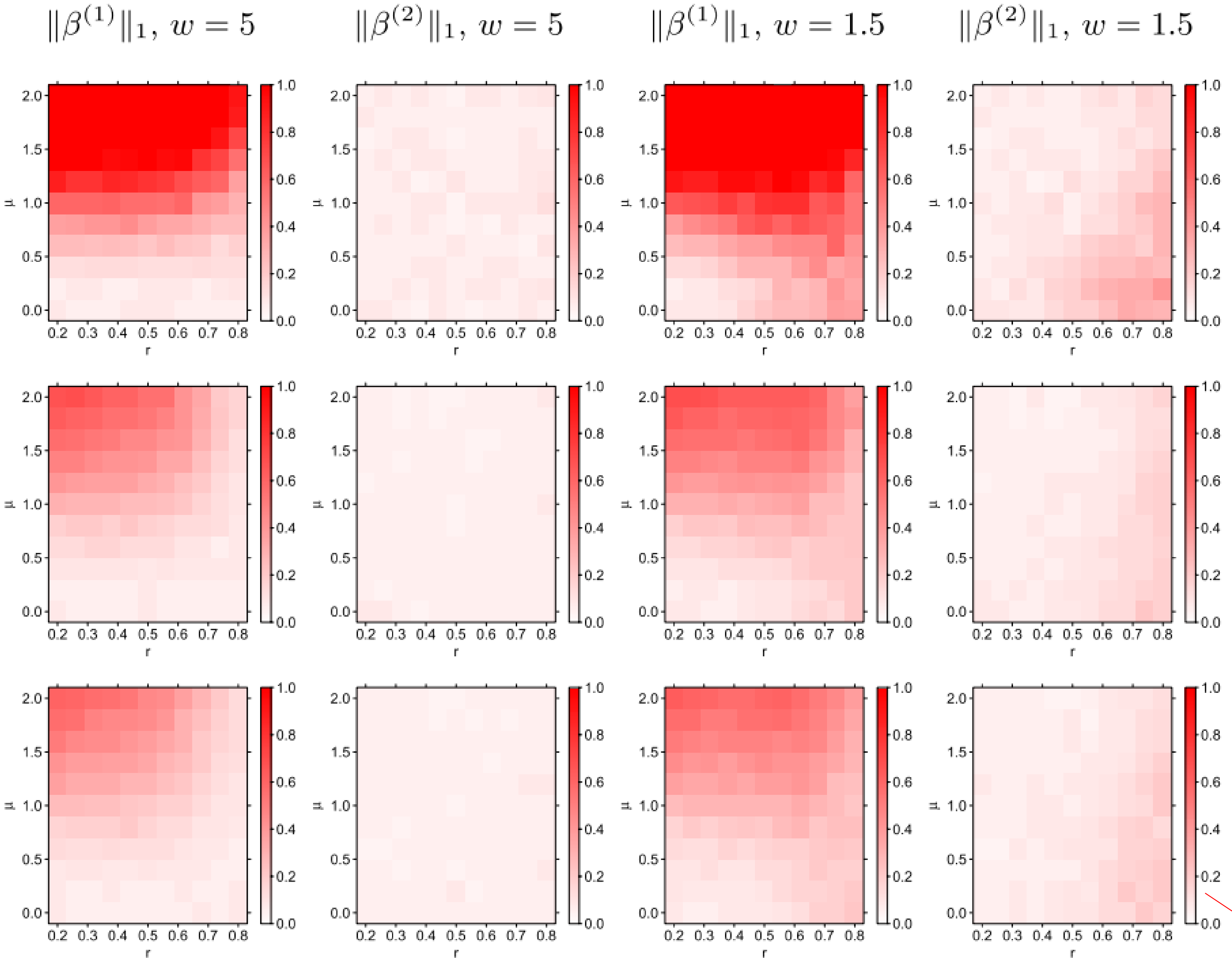


(a) JCDC, $w = 5$ (b) JCDC, $w = 1.5$ (c) SC (d) KM



(e) CASC (f) CESNA (g) BAGC

K = 3 , n1: 30, n2: 50, n3: 70



μ, r ↓ => beta ↓
But, not large impact r.

K = 2 , n1 = 100, n2= 50

K = 3 , n1 = n2 = n3 = 50

K = 3 , n1: 30, n2: 50, n3: 70

Estimated $\frac{\|\hat{\beta}^{(l)}\|_1}{k}$

6. Data applications

Data : World Trade data

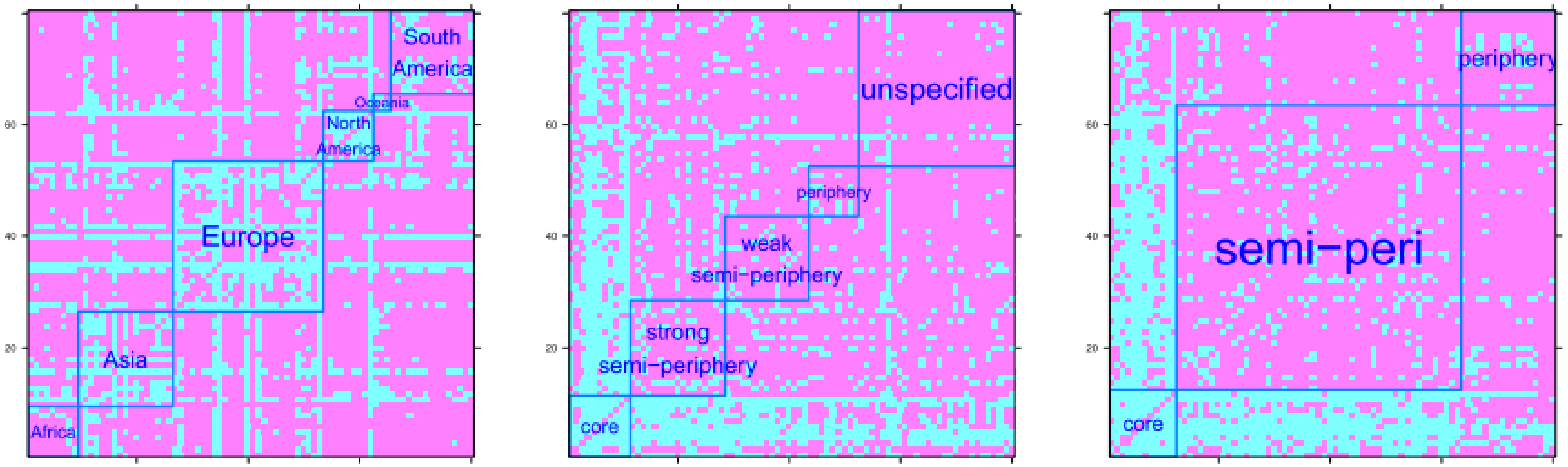
```
Number of nodes: 89
Number of edges: 1012
```

```
[12] G.nodes

NodeView(('Algeria', 'Argentina', 'Australia', 'Austria', 'Barbados', 'Bangladesh', 'Belgium /Lux.', 'Belize', 'Bolivia', 'Brazil', 'Car
'Fiji', 'Finland', 'France Mon.', 'French Guiana', 'Germany', 'Greece', 'Guadeloupe', 'Guatemala', 'Honduras', 'Hong Kong', 'Hungary', '
'Madagascar', 'Malaysia', 'Martinique', 'Mauritius', 'Mexico', 'Morocco', 'Netherlands', 'New Zealand', 'Nicaragua', 'Norway', 'Oman', '
'Seychelles', 'Singapore', 'Slovenia', 'Southern Africa', 'Spain', 'Sri Lanka', 'Sweden', 'Switzerland', 'Thailand', 'Trinidad Tobago',
'*Vertices', 'Continent', 'World_system', '*Vector', 'x_coordinates', 'y_coordinates.vec', 'GDP_1995.vec'))
```

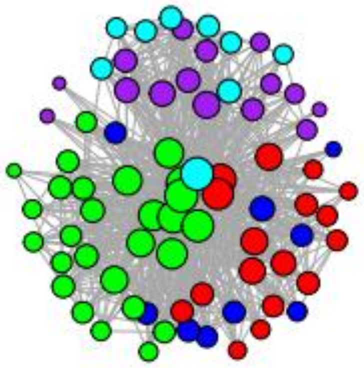
```
G.edges

OutMultiEdgeView([('Argentina', 'Chile', 0), ('Argentina', 'Uruguay', 0), ('Argentina', 'Brazil', 0), ('Argentina', 'Bolivia', 0), ('Argentina', 'Paraguay', 0), ('A
('Australia', 'Fiji', 0), ('Australia', 'New Zealand', 0), ('Australia', 'Philippines', 0), ('Australia', 'Singapore', 0), ('Australia', 'Malaysia', 0), ('Australia
'France Mon.', 0), ('Austria', 'Finland', 0), ('Austria', 'United Kingdom', 0), ('Austria', 'Spain', 0), ('Austria', 'Switzerland', 0), ('Austria', 'Hungary', 0), (
('Austria', 'Denmark', 0), ('Austria', 'Germany', 0), ('Austria', 'Netherlands', 0), ('Austria', 'New Zealand', 0), ('Austria', 'Belgium /Lux.', 0), ('Austria', 'It
'Latvia', 0), ('Austria', 'Tunisia', 0), ('Austria', 'Turkey', 0), ('Austria', 'Ireland', 0), ('Austria', 'Egypt', 0), ('Austria', 'Israel', 0), ('Austria', 'Portug
0), ('Austria', 'Greece', 0), ('Austria', 'Poland', 0), ('Austria', 'Romania', 0), ('Austria', 'Croatia', 0), ('Austria', 'Southern Africa', 0), ('Barbados', 'Beliz
/Lux.', 'Norway', 0), ('Belgium /Lux.', 'France Mon.', 0), ('Belgium /Lux.', 'Finland', 0), ('Belgium /Lux.', 'Martinique', 0), ('Belgium /Lux.', 'Reunion', 0), ('B
Lanka', 0), ('Belgium /Lux.', 'Switzerland', 0), ('Belgium /Lux.', 'Hungary', 0), ('Belgium /Lux.', 'Sweden', 0), ('Belgium /Lux.', 'Denmark', 0), ('Belgium /Lux.',
('Belgium /Lux.', 'New Zealand', 0), ('Belgium /Lux.', 'Italy', 0), ('Belgium /Lux.', 'Tunisia', 0), ('Belgium /Lux.', 'Turkey', 0), ('Belgium /Lux.', 'Singapore',
/Lux.', 'Ireland', 0), ('Belgium /Lux.', 'Mauritius', 0), ('Belgium /Lux.', 'Argentina', 0), ('Belgium /Lux.', 'Egypt', 0), ('Belgium /Lux.', 'Honduras', 0), ('Belg
('Belgium /Lux.', 'Portugal', 0), ('Belgium /Lux.', 'Algeria', 0), ('Belgium /Lux.', 'Slovenia', 0), ('Belgium /Lux.', 'Venezuela', 0), ('Belgium /Lux.', 'Czech Rep
'Poland', 0), ('Belgium /Lux.', 'Romania', 0), ('Belgium /Lux.', 'Jordan', 0), ('Belgium /Lux.', 'Austria', 0), ('Belgium /Lux.', 'Croatia', 0), ('Belgium /Lux.', '
('Brazil', 'Chile', 0), ('Brazil', 'Fiji', 0), ('Brazil', 'United States', 0), ('Brazil', 'Mexico', 0), ('Brazil', 'Belize', 0), ('Brazil', 'Argentina', 0), ('Brazi
'Honduras', 0), ('Brazil', 'El Salvador', 0), ('Brazil', 'Bolivia', 0), ('Brazil', 'Trinidad Tobago', 0), ('Brazil', 'Panama', 0), ('Brazil', 'Paraguay', 0), ('Braz
'Southern Africa', 0), ('Brazil', 'Ecuador', 0), ('Canada', 'Chile', 0), ('Canada', 'United States', 0), ('Canada', 'New Zealand', 0), ('Canada', 'Australia', 0), (
('Canada', 'Nicaragua', 0), ('Canada', 'Colombia', 0), ('Canada', 'Trinidad Tobago', 0), ('Canada', 'Panama', 0), ('Canada', 'Venezuela', 0), ('Canada', 'Peru', 0),
'Argentina', 0), ('Chile', 'Uruguay', 0), ('Chile', 'Bolivia', 0), ('Chile', 'Paraguay', 0), ('Chile', 'Peru', 0), ('China', 'Norway', 0), ('China', 'Chile', 0), ('
```

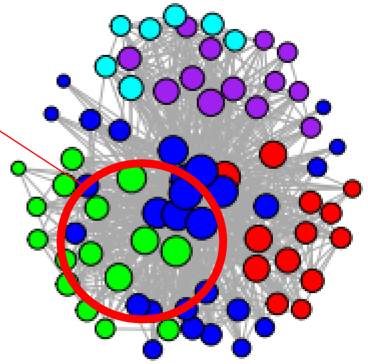


(a) A by continent (b) A by position '80 (c) A by position '94

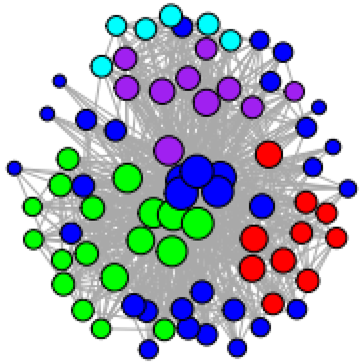
Europe two separate.
Don't capture Africa.



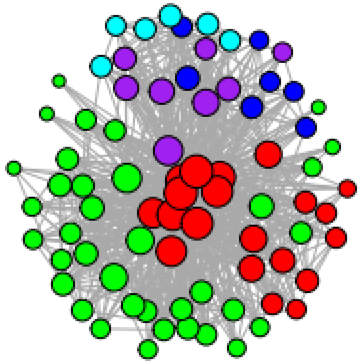
(d) Continent



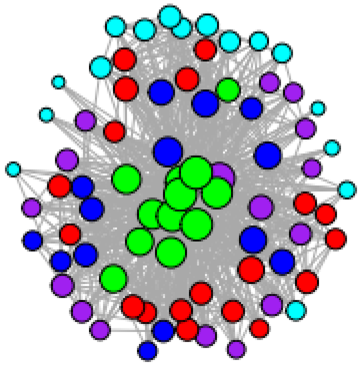
(e) JCDC, $w_n = 5$
NMI=0.54



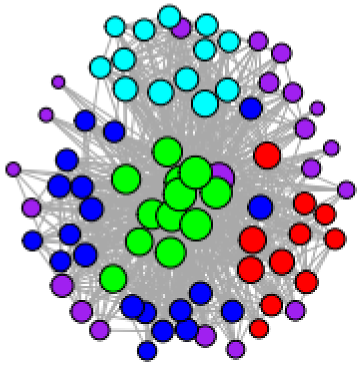
(f) JCDC, $w_n = 1.5$
NMI=0.50



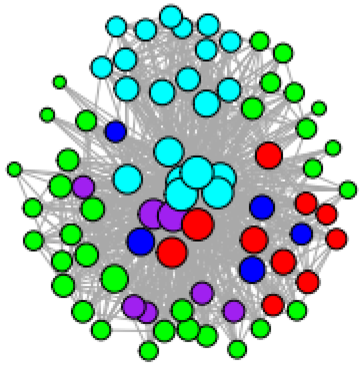
(g) SC
NMI=0.47



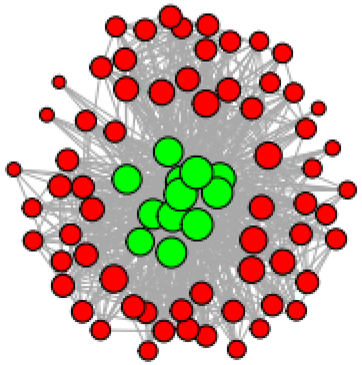
(h) KM
NMI=0.25



(i) CASC
NMI=0.39



(j) CESNA
NMI=0.26



(k) BAGC
NMI=0.11

Reference

- https://timbr.ai/community-detection-algorithm/
- https://mathworld.wolfram.com/AdjacencyMatrix.html
- https://convex-optimization-for-all.github.io/contents/chapter23/2021/03/28/23_01_Coordinate_descent/
- https://onlinelibrary.wiley.com/doi/epdf/10.1002/cpe.4040
- http://vlado.fmf.uni-lj.si/pub/networks/data/esna/metalWT.htm

- F. Glover. Future paths for integer programming and links to artificial intelligence. Comput. Oper. Res., 13(5):533–549, May 1986.
- J. McAuley and J. Leskovec. Learning to discover social circles in ego networks. In Advances in Neural Information Processing Systems 25, pages 548–556, 2012.
- Wouter De Nooy, Andrej Mrvar, and Vladimir Batagelj. Exploratory social network analysis with Pajek, volume 27. Cambridge University Press, 2011.