Community detection on networks

Caterina De Bacco

CSSS, Santa Fe, 2018

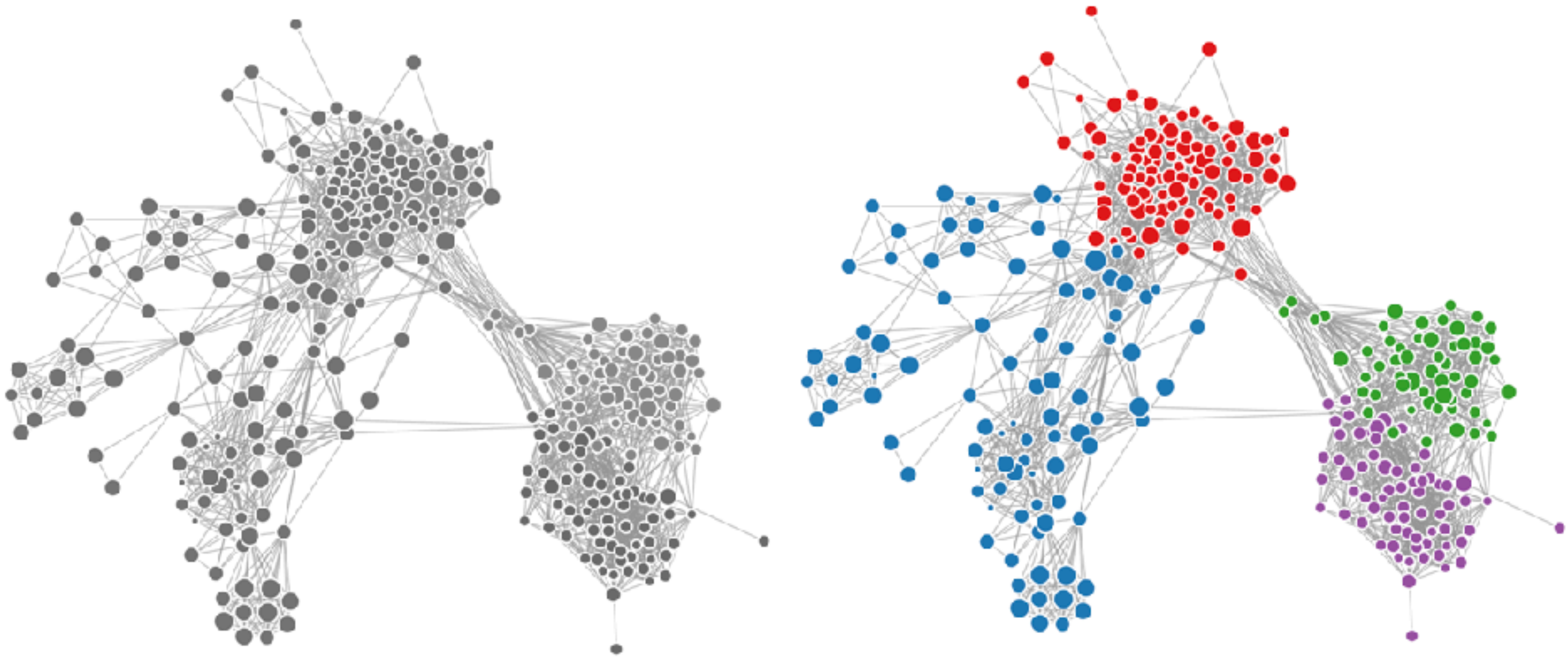
caterina.debacco@gmail.com https://github.com/cdebacco

Lecture 2: *implementing the model*

1. **Testing the model (20mins)**
   - Measuring correlation with 'ground truth': F1score, Mutual Information
   - No ground truth: AUC and MDL
2. **Analysing the results (20 mins)**
   - Normalize and extract max group
   - Visualize results
   - Statistical significance: many local optima
3. *Advanced topics (10mins): Metadata*

Community detection: *testing the model.*

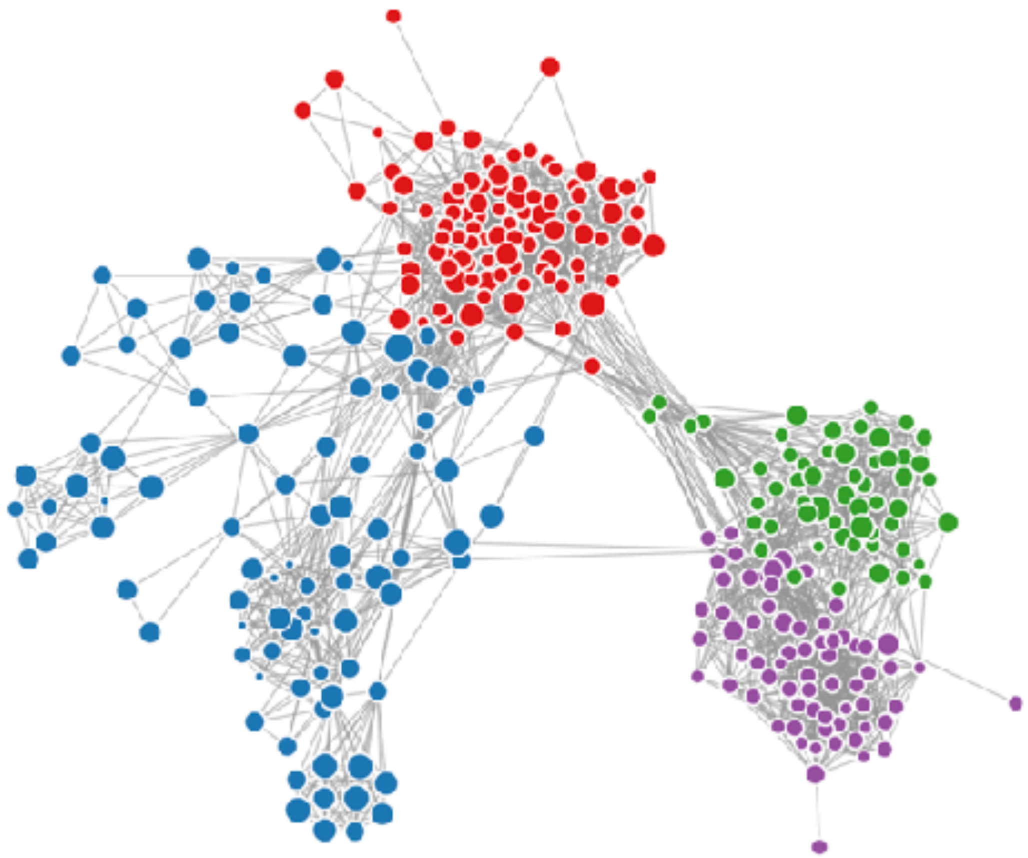
How do you evaluate if the model is 'good'?



Community detection: *testing the model.*

How do you evaluate if the model is 'good'?

'*Easy', if we have ground truth communities*



Cluster matching

4 5 11 24 28 50 69 90 95 113
17 20 27 56 58 59 62 63 65 70 76 87 95 96 113
0 4 9 16 23 41 90 93 104
3 5 10 11 52 58 74 81 82 84 97 98 107
2 3 10 40 52 72 74 81 98 102 107
19 29 30 35 44 55 79 80 82 93 94 101 109
1 25 33 37 45 62 89 103 105 109
12 14 18 26 31 34 36 38 39 42 43 54 55 61 71 79 80 85 99
35 42 44 48 52 56 57 59 63 66 75 80 86 87 91 92 96 97 98 112
7 8 9 21 22 23 40 47 50 51 64 68 77 78 82 108 111
2 6 13 15 32 39 47 60 64 82 100 106
13 24 25 32 46 48 49 52 53 58 67 69 73 80 83 84 88 89 107 110 114

Inferred: C*

19 29 30 35 55 79 94 101
2 6 13 15 32 39 47 60 64 100 106
3 5 10 40 52 72 74 81 84 98 102 107
44 48 57 66 75 86 91 92 110 112
36 42 80 82 90
12 14 18 26 31 34 38 43 54 61 71 85 99
0 4 9 16 23 41 93 104
7 8 21 22 51 68 77 78 108 111
17 20 27 56 62 65 70 76 87 95 96 113
11 24 50 59 63 69 97
28 46 49 53 58 67 73 83 88 114
1 25 33 37 45 89 103 105 109

Ground truth: C

Cluster matching

```
4 5 11 24 28 50 69 90 95 113
17 20 27 56 58 59 62 63 65 70 76 87 95 96 113
0 4 9 16 23 41 90 93 104
3 5 10 11 52 58 74 81 82 84 97 98 107
2 3 10 40 52 72 74 81 98 102 107
19 29 30 35 44 55 79 80 82 93 94 101 109
1 25 33 37 45 62 89 103 105 109
12 14 18 26 31 34 36 38 39 42 43 54 55 61 71 79 80 85 99
35 42 44 48 52 56 57 59 63 66 75 80 86 87 91 92 96 97 98 112
7 8 9 21 22 23 40 47 50 51 64 68 77 78 82 108 111
2 6 13 15 32 39 47 60 64 82 100 106
13 24 25 32 46 48 49 52 53 58 67 69 73 80 83 84 88 89 107 110 114
```

Inferred: C*

```
19 29 30 35 55 79 94 101
2 6 13 15 32 39 47 60 64 100 106
3 5 10 40 52 72 74 81 84 98 102 107
44 48 57 66 75 86 91 92 110 112
36 42 80 82 90
12 14 18 26 31 34 38 43 54 61 71 85 99
0 4 9 16 23 41 93 104
7 8 21 22 51 68 77 78 108 111
17 20 27 56 62 65 70 76 87 95 96 113
11 24 50 59 63 69 97
28 46 49 53 58 67 73 83 88 114
1 25 33 37 45 89 103 105 109
```

Ground truth: C

Cluster matching

4 5 11 24 28 50 69 90 95 113
17 20 27 56 58 59 62 63 65 70 76 87 95 96 113
0 4 9 16 23 41 90 93 104
3 5 10 11 52 58 74 81 82 84 97 98 107
2 3 10 40 52 72 74 81 98 102 107
19 29 30 35 44 55 79 80 82 93 94 101 109
1 25 33 37 45 62 89 103 105 109
12 14 18 26 31 34 36 38 39 42 43 54 55 61 71 79 80 85 99
35 42 44 48 52 56 57 59 63 66 75 80 86 87 91 92 96 97 98 112
7 8 9 21 22 23 40 47 50 51 64 68 77 78 82 108 111
2 6 13 15 32 39 47 60 64 82 100 106
13 24 25 32 46 48 49 52 53 58 67 69 73 80 83 84 88 89 107 110 114

Inferred: C*

19 29 30 35 55 79 94 101
2 6 13 15 32 39 47 60 64 100 106
3 5 10 40 52 72 74 81 84 98 102 107
44 48 57 66 75 86 91 92 110 112
36 42 80 82 90
12 14 18 26 31 34 38 43 54 61 71 85 99
0 4 9 16 23 41 93 104
7 8 21 22 51 68 77 78 108 111
17 20 27 56 62 65 70 76 87 95 96 113
11 24 50 59 63 69 97
28 46 49 53 58 67 73 83 88 114
1 25 33 37 45 89 103 105 109

Ground truth: C

Cluster matching

```
C*:2 6 13 15 32 39 47 60 64 82 100 106
C: 2 6 13 15 32 39 47 60 64 100 106 121 132

82: False Positive
121,132: False Negatives
2,6,13,etc…: True Positives
```

Cluster matching

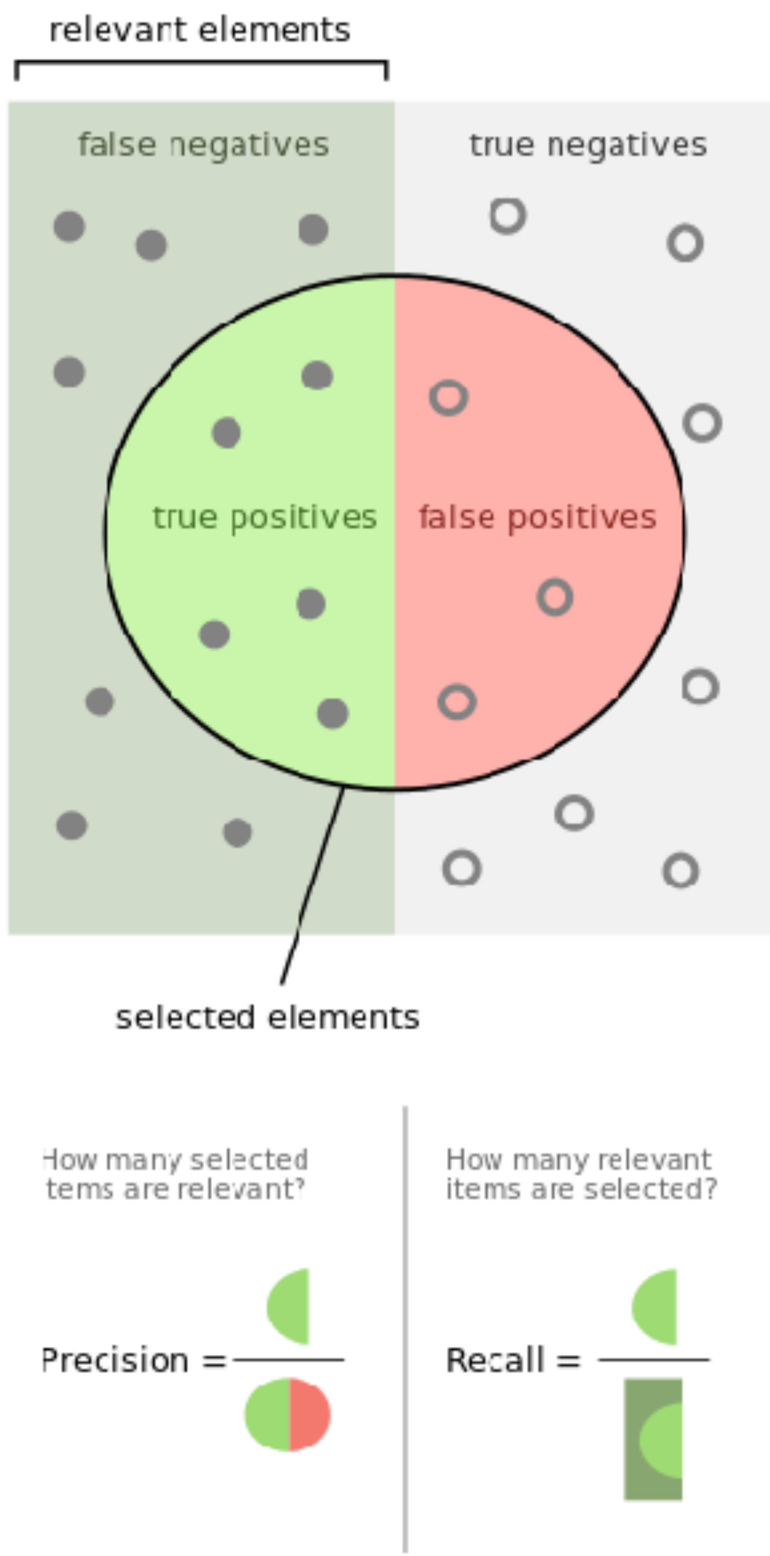
```
C*:2 6 13 15 32 39 47 60 64 82 100 106
C: 2 6 13 15 32 39 47 60 64 100 106 121 132

82: False Positive
121,132: False Negatives
2,6,13,etc…: True Positives
```

$$\delta(C_i^*, C_j)$$

Distance between ground truth and inferred

Cluster matching: F1-score




```
C*:2 6 13 15 32 39 47 60 64 82 100 106
C: 2 6 13 15 32 39 47 60 64 100 106 121 132
82: False Positive
121,132: False Negatives
2,6,13,etc…: True Positives
```

Precision= 11/12=0.92
Recall= 11/13=0.85
F1-score=2 x 0.92*0.85/1.77=**0.88**

$$\text{F1-score} = 2\,\frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$$

https://en.wikipedia.org/wiki/F1_score

Cluster matching: F1-score

```
C: 2 6 13 15 32 39 47 60 64 82 100 106
C*:2 6 13 15 32 39 47 60 64 82 100 106
none: False Positives
none: False Negative
```

Take as C* all possible subsets of nodes

Precision= 12/12=1
Recall= 12/12=1.
F1-score=2 x 1/2.=**1.0!!**

Cluster matching: F1-score

```
C: 2 6 13 15 32 39 47 60 64 100 106 121 132
C*:2 6 13 15 32 39 47 60 64 82 100 106
121,132: False Positives
82: False Negative
```

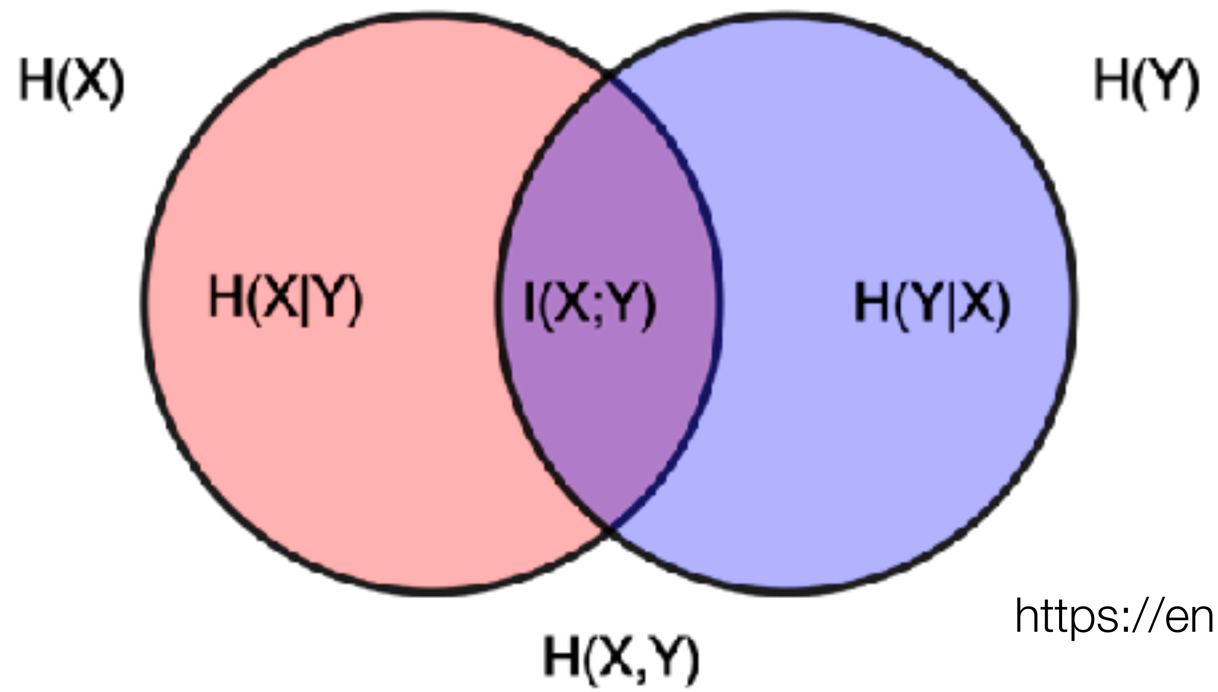
Precision= 11/13=0.85
Recall= 11/12=0.92
F1-score=2 x 0.92*0.85/1.77=**0.88**

Consider both C* and C (in turn) as reference, and calculate F1 on both sides

$$\frac{1}{2|C^*|}\sum_{C_i^* \in C^*} \max_{C_j \in C} \delta(C_i^*, C_j) + \frac{1}{2|C|}\sum_{C_j \in C} \max_{C_i^* \in C^*} \delta(C_i^*, C_j)$$

$\delta(C_i^*, C_j)$ **F1-score**

Information theory approach: NMI



https://en.wikipedia.org/wiki/Mutual_information

$$NMI(C, C^*) = \frac{I(C:C^*)}{Z(H(C), H(C^*))}$$

Mutual information

$$I(C:C^*) = \frac{1}{2}\left[H(C) - H(C|C^*) + H(C^*) - H(C^*|C)\right]$$

Normalization

$$Z(H(C), H(C^*)) = \max(H(C), H(C^*))$$

Danon et al. 2005

$$Z(H(C), H(C^*)) = \frac{1}{2}(H(C) + H(C^*))$$

McDaid et al. 2011

Cluster similarity measures

- Cluster matching (F1 score, etc…)
- Information theory (NMI, VI)
- Pair counting (Rand index, Jaccard, etc…)

See Fortunato et Hric 2016 for an overview

Overlapping communities: distance between vectors

U_i

```
0 0 0 0.0790622 0
3 0.0533999 0 0.139296 0
76 0 0 0.0805629 0.10241
127 0 0 0 0.207446
177 0 0.0738358 0.158358 0.024311
1 0 0 0.185704 0.0796013
6 0.0270201 0 0 0.0864358
17 0 0 0.192816 0.0215193
34 0 0 0.211593 0.0223843
```

U^*_i

```
0 0 0.00483982 0.00224297 0
3 0 0.01271 0.00688526 0
76 0 0.00115951 0.0137771 0.00347393
127 0 0.00605217 0.0168567 0
177 0.00267315 0 0.00549152 0.0149187
1 0 0.017335 0.00262135 0.0033609
6 0.00186055 0 0 0.00820299
17 0 0.0184664 0 0
34 0 0.0183341 0 0.00153079
```

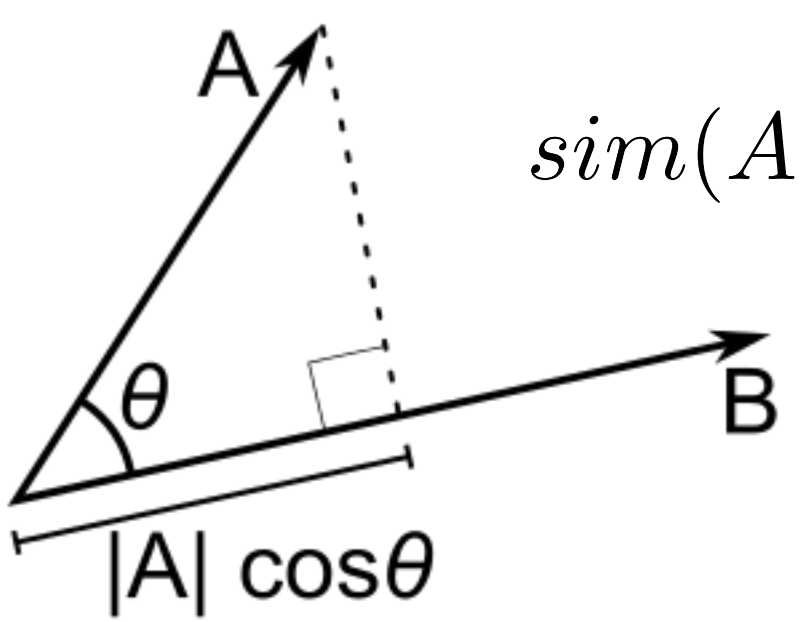
Overlapping communities: cosine-similarity, L1, etc…

U_i

```
0 0 0 0.0790622 0
3 0.0533999 0 0.139296 0
76 0 0 0.0805629 0.10241
127 0 0 0 0.207446
177 0 0.0738358 0.158358 0.024311
1 0 0 0.185704 0.0796013
6 0.0270201 0 0 0.0864358
17 0 0 0.192816 0.0215193
34 0 0 0.211593 0.0223843
```

U^*_i

```
0 0 0.00483982 0.00224297 0
3 0 0.01271 0.00688526 0
76 0 0.00115951 0.0137771 0.00347393
127 0 0.00605217 0.0168567 0
177 0.00267315 0 0.00549152 0.0149187
1 0 0.017335 0.00262135 0.0033609
6 0.00186055 0 0 0.00820299
17 0 0.0184664 0 0
34 0 0.0183341 0 0.00153079
```



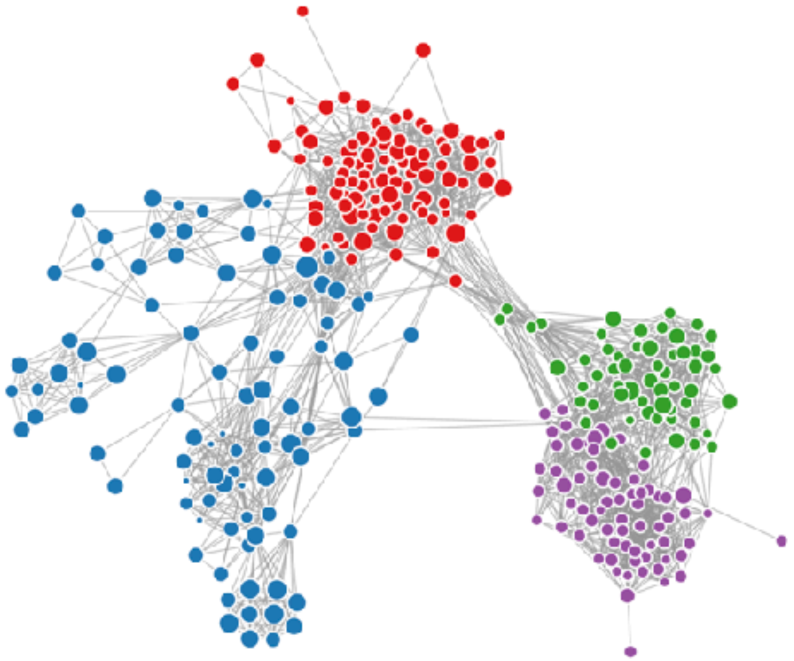
$$sim(A,B) = \cos(\Theta) = \frac{A \cdot B}{||A||\,||B||}$$

Testing the model: *compare structural and functional communities.*

Structural: set of nodes with a particular connectivity structure (edge density, average degree, etc..)
Functional: set of nodes with common function (e.g. common role, attribute, etc..).

Idea: different functional communities have different structures

Yang and Leskovec, Defining and evaluating network communities based on ground-truth ,2015



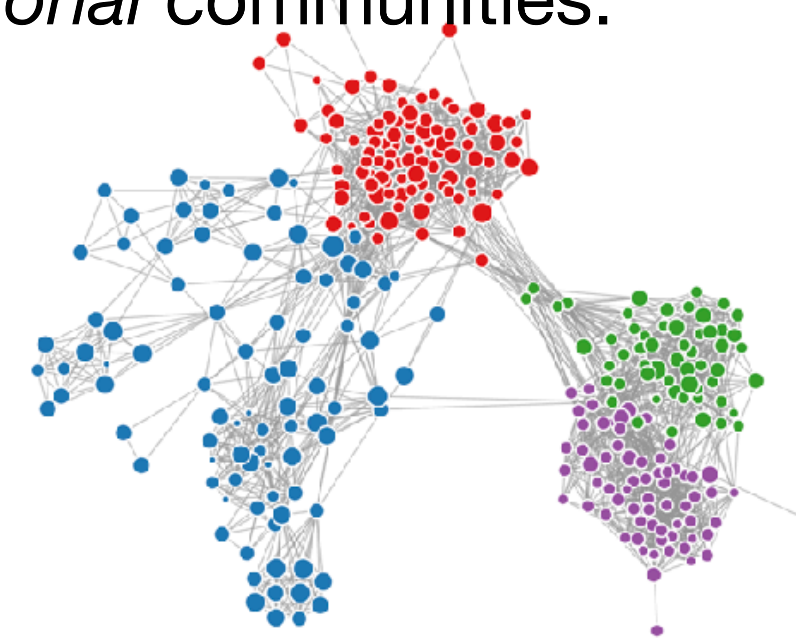
Testing the model: *compare structural and functional communities.*

Structural: set of nodes with a particular connectivity structure (edge density, average degree, etc..)
Functional: set of nodes with common function (e.g. common role, attribute, etc..).

Idea: different functional communities have different structures

1. Detect communities based on *structure.*
2. See if these correspond to ground truth *functional* communities.

   Need to define a 'proper' scoring function.

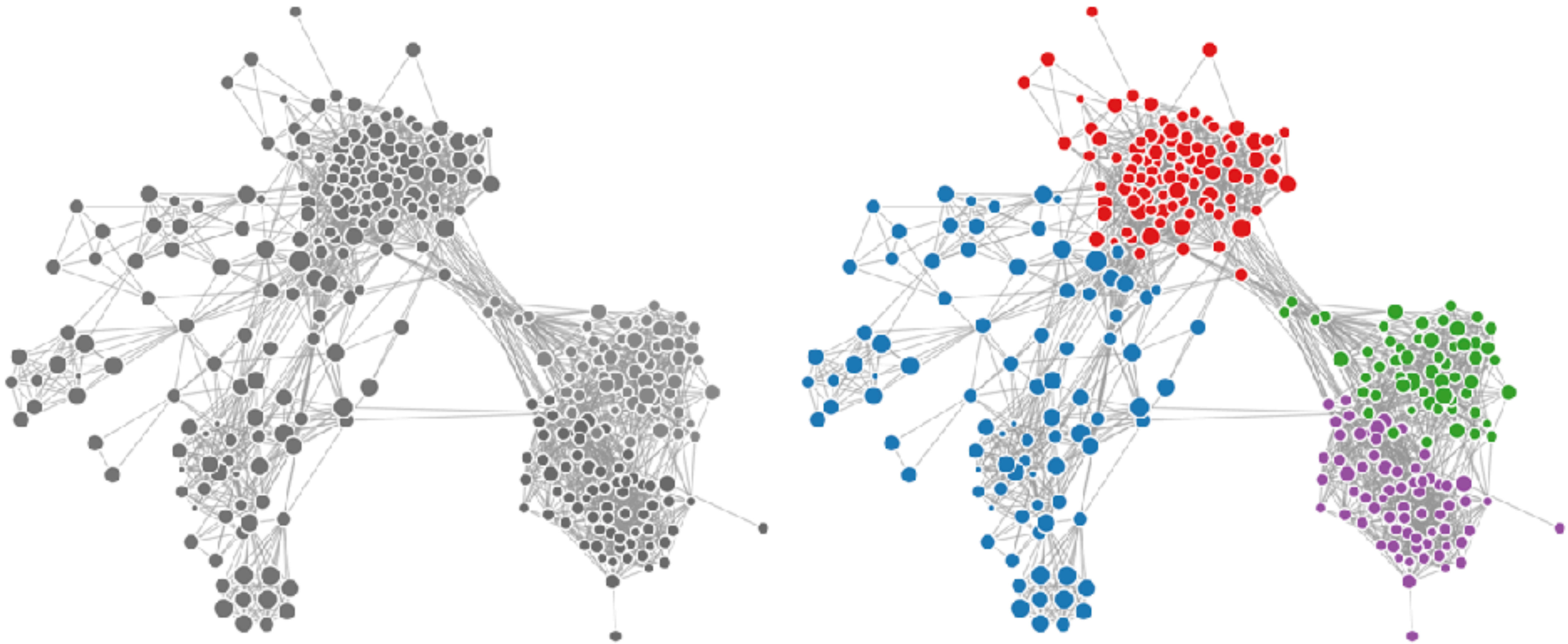


Yang and Leskovec, Defining and evaluating network communities based on ground-truth ,2015

Community detection: *testing the model.*

How do you evaluate if the model is 'good'?

Unsupervised: no ground truth to compare with ….

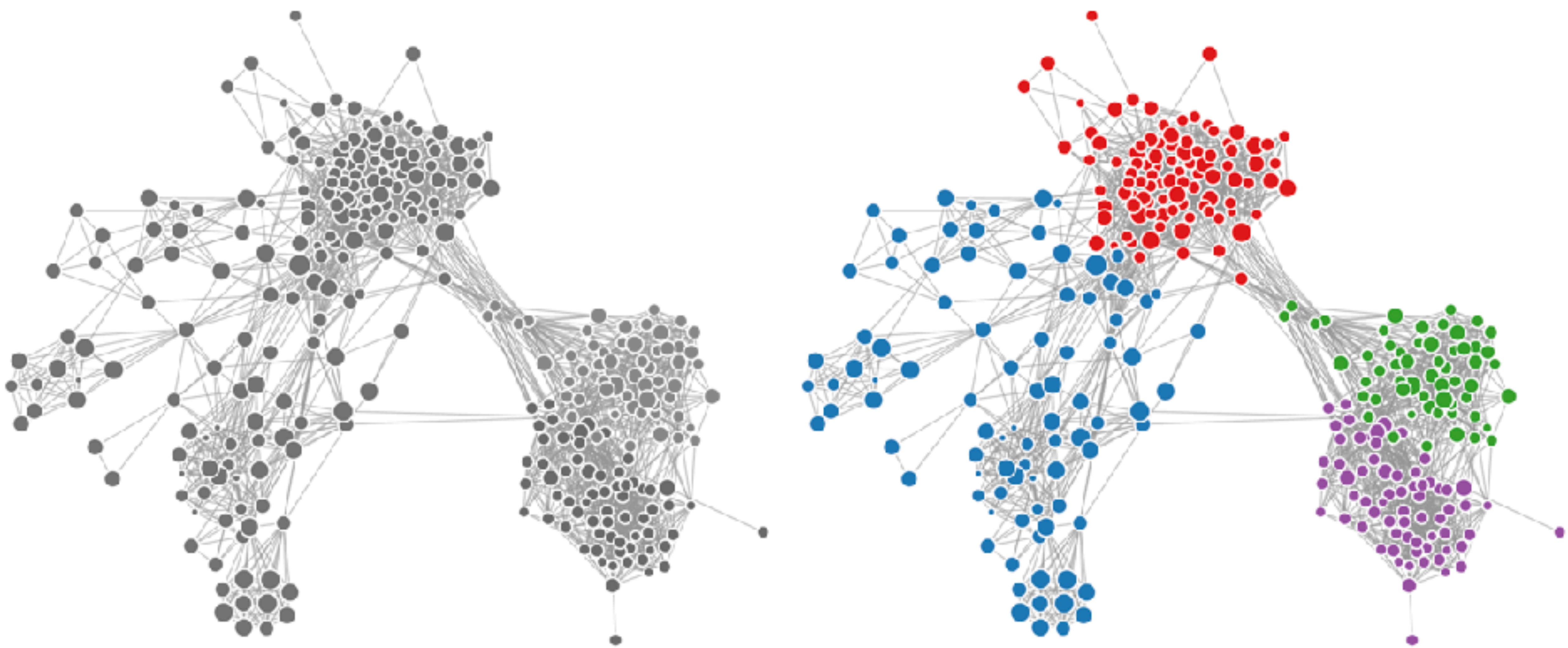


Community detection: *testing the model.*

How do you evaluate if the model is 'good'?

Unsupervised: no ground truth to compare with ….

Don't be fooled by the visualization!



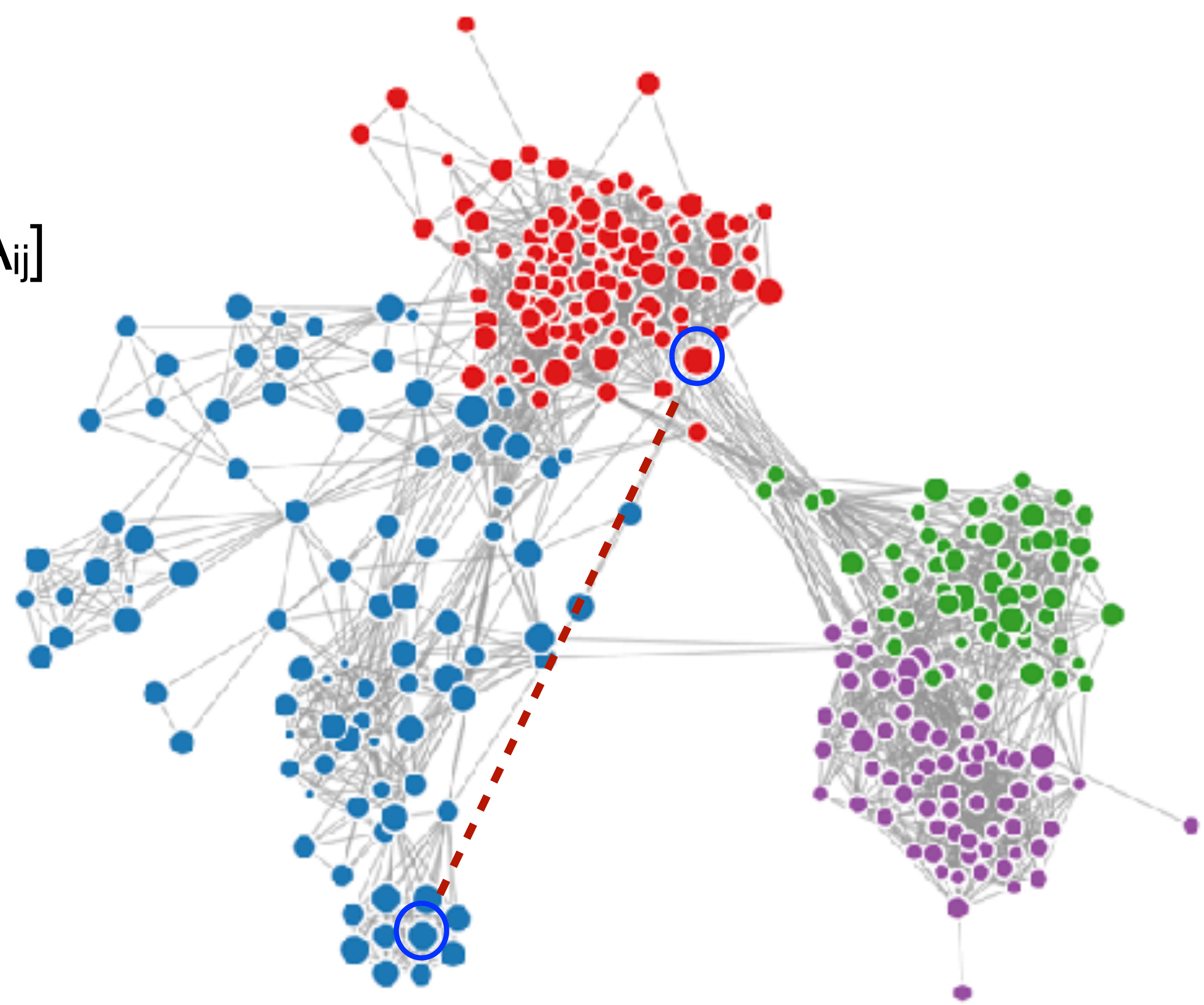
Testing the model: *predictive performance*

Idea: a model is 'good' if it's able to predict the existence of links

Compare the observed A_{ij} with the inferred $E[A_{ij}]$

$$A_{ij}^{\alpha} \sim Poi(M_{ij}^{\alpha})$$

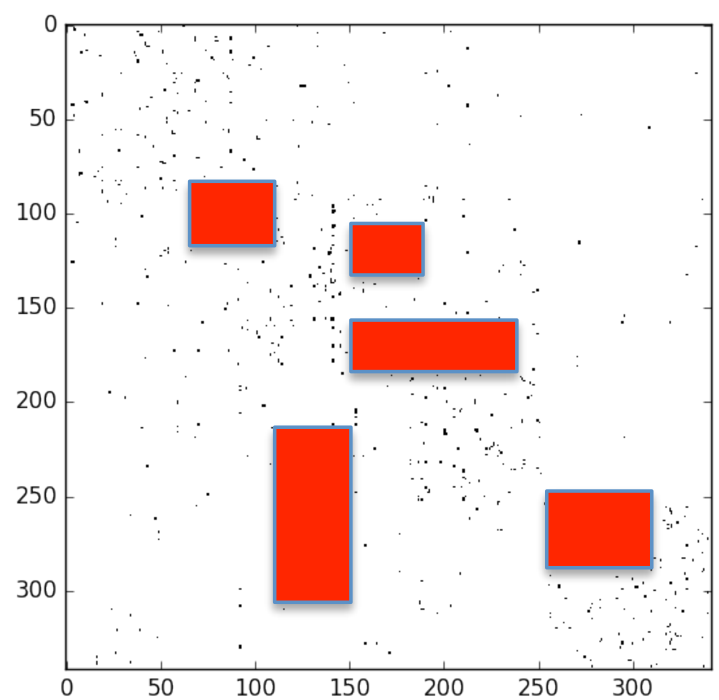
$$M_{ij}^{\alpha} = \sum_{k,q} u_{ik} v_{jk} w_{kq}^{\alpha}$$



Testing the model: *predictive performance*

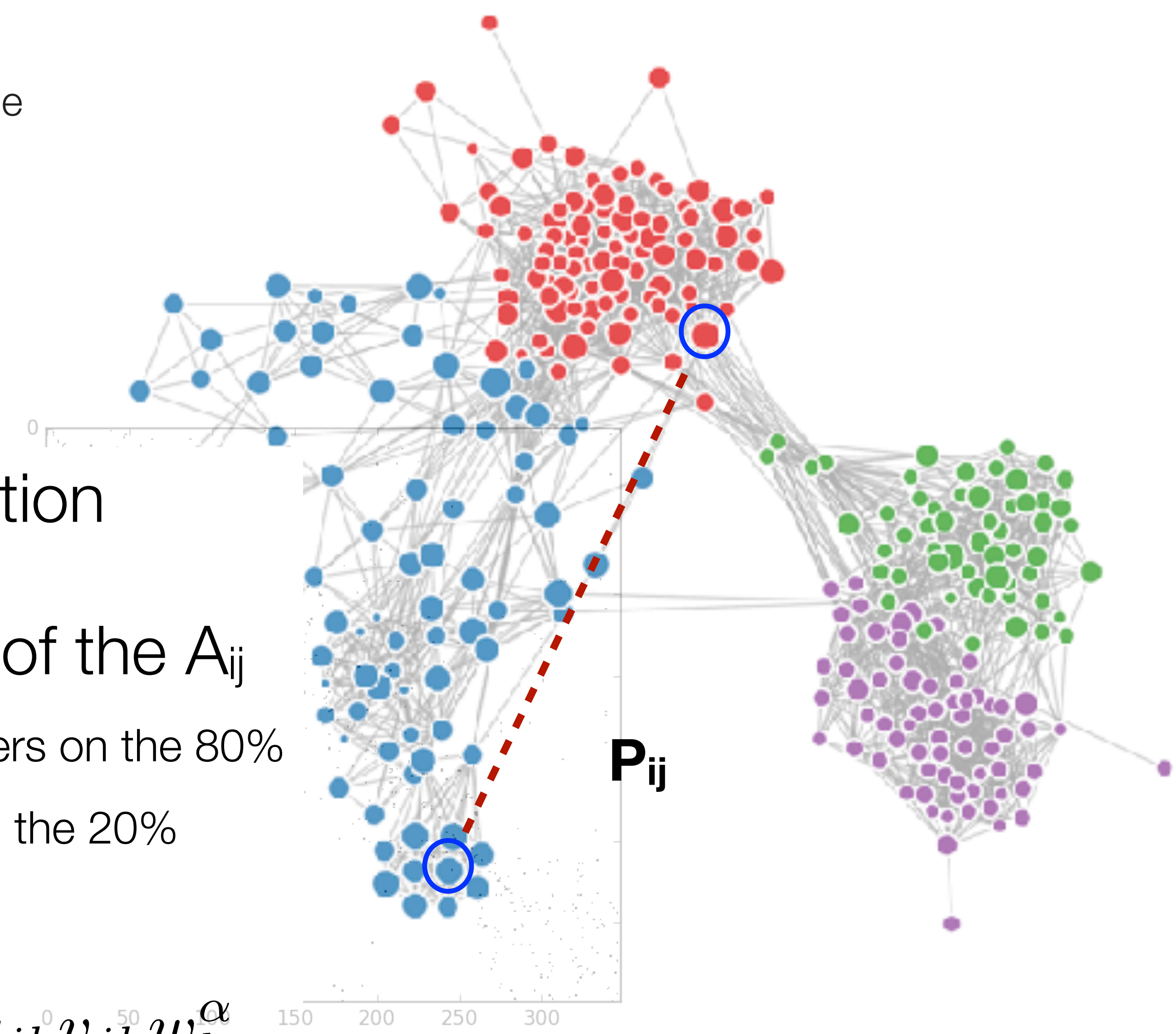
Need a metric for comparing the observed A_{ij} with the inferred $E[A_{ij}]$

AUC (Area under the curve):
the probability that a randomly chosen missing connection (a true positive) is given a higher score than a randomly chosen pair of unconnected vertices (a true negative)



Cross-validation

- Hide 20% of the A_{ij}
- Fit the parameters on the 80%
- Calculate Mij on the 20%

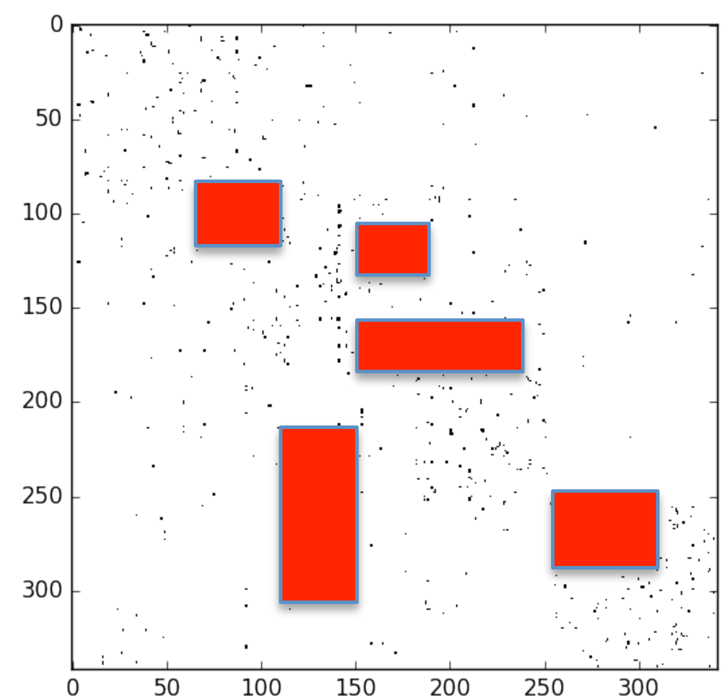


$$M_{ij}^{\alpha} = \sum_{k,q} u_{ik} v_{jk} w_{kq}^{\alpha}$$

Testing the model: *predictive performance*

Need a metric for comparing the observed A_{ij} with the inferred $E[A_{ij}]$

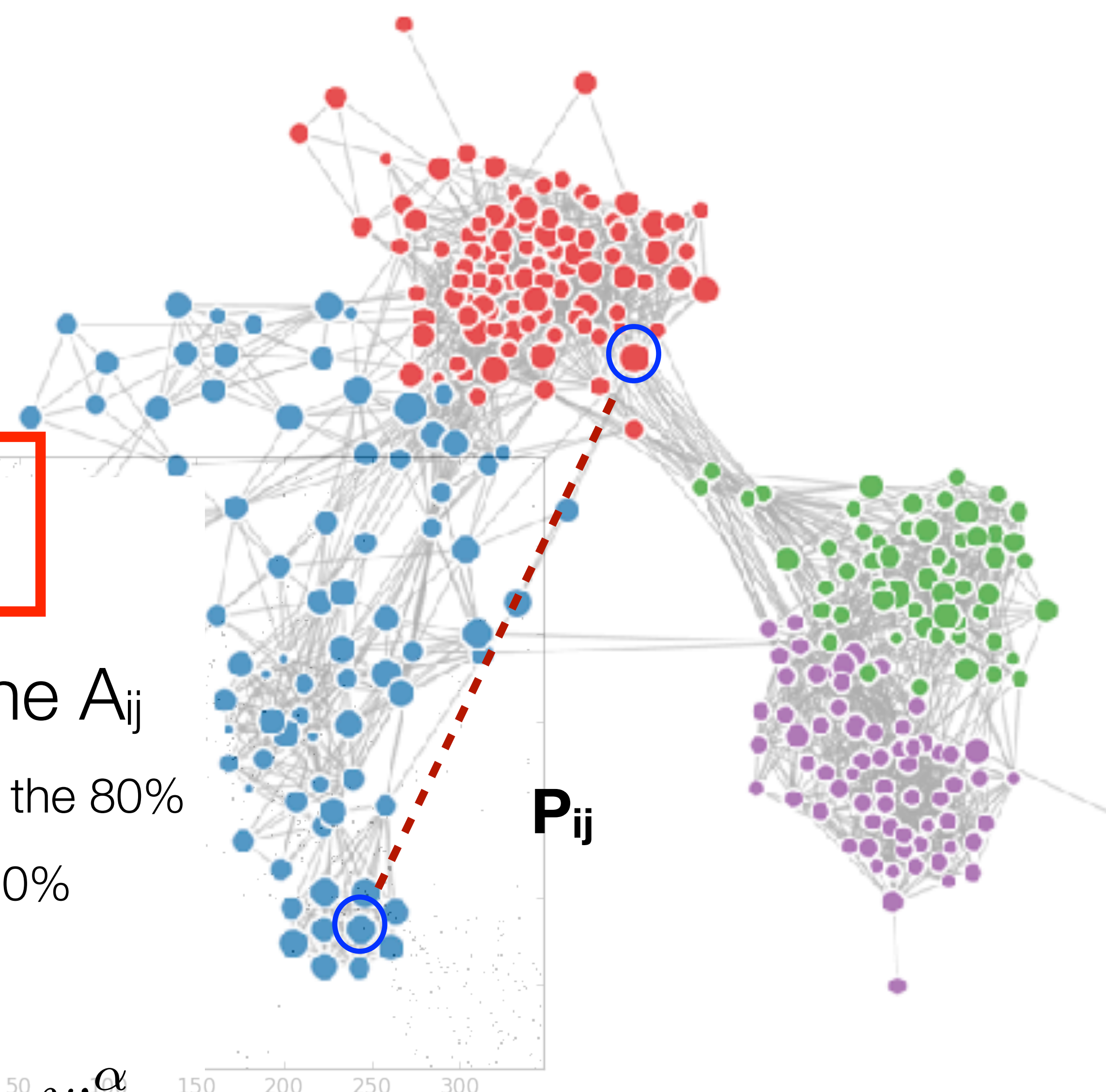
AUC (Area under the curve):
the probability that a randomly chosen missing connection (a true positive) is given a higher score than a randomly chosen pair of unconnected vertices (a true negative)



Cross-validation

- Hide 20% of the A_{ij}
- Fit the parameters on the 80%
- Calculate Mij on the 20%

$$M_{ij}^{\alpha} = \sum_{k,q} u_{ik} v_{jk} w_{kq}^{\alpha}$$

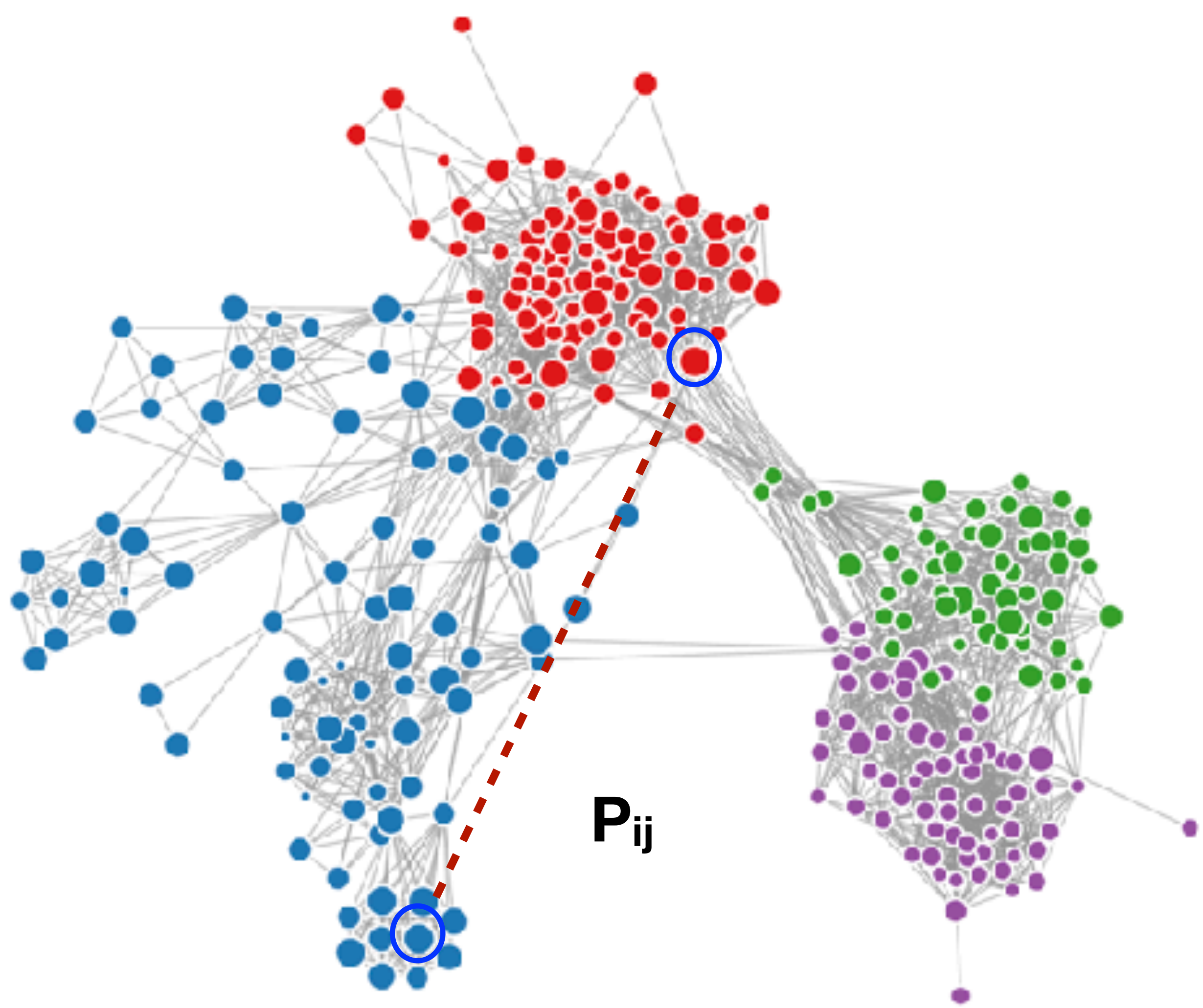


Testing the model: *predictive performance*

Idea: a model is ‘good’ if it’s able to predict the existence of links

AUC (Area under the curve):
the probability that a randomly chosen missing connection (a true positive) is given a higher score than a randomly chosen pair of unconnected vertices (a true negative)

A_{ij}	M_{ij}
1	1.98
1	1.03
1	0.75
0	0.67
0	0.42
1	0.21
0	0.12
0	0.01
0	0.00



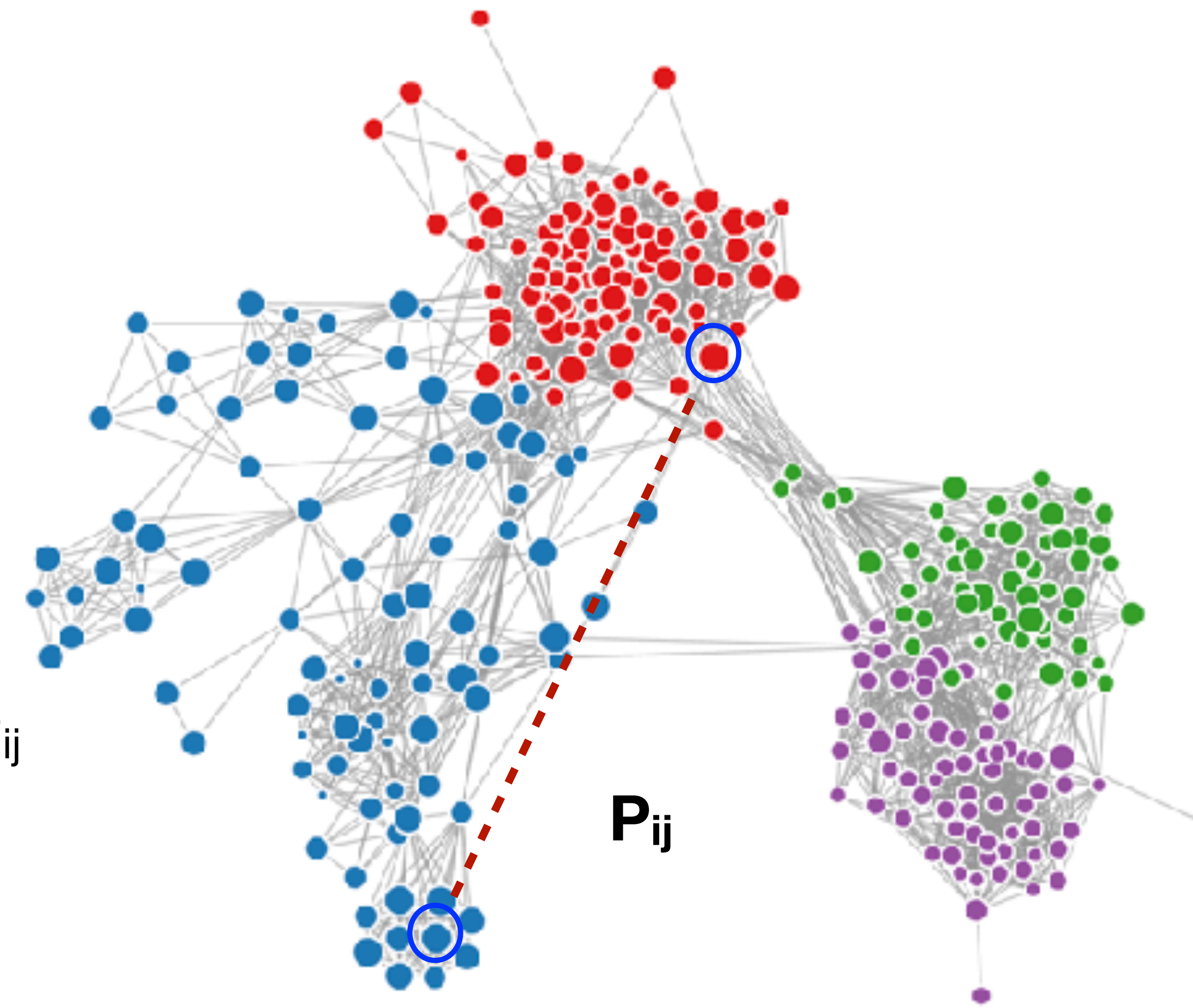
Testing the model: *predictive performance*

Idea: a model is ‘good’ if it’s able to predict the existence of links

AUC (Area under the curve):
the probability that a randomly chosen missing connection (a true positive) is given a higher score than a randomly chosen pair of unconnected vertices (a true negative)

A_{ij}	M_{ij}
1	1.98
1	1.03
1	0.75
0	0.67
0	0.42
1	0.21
0	0.12
0	0.01
0	0.00

2 non-edge have higher P_{ij} than a true edge: bad!

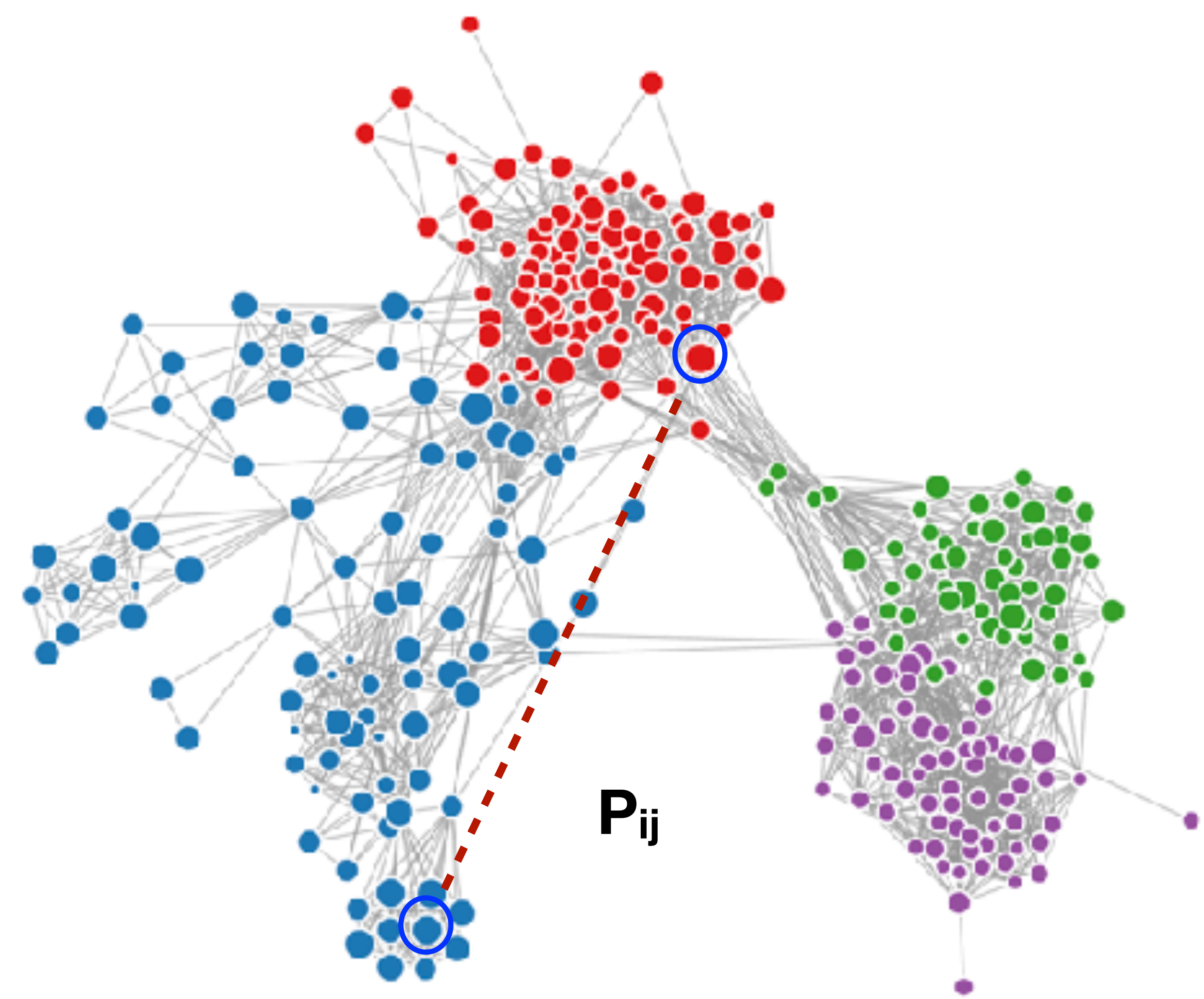


Testing the model: *predictive performance*

Idea: a model is ‘good’ if it’s able to predict the existence of links

AUC (Area under the curve):
the probability that a randomly chosen missing connection (a true positive) is given a higher score than a randomly chosen pair of unconnected vertices (a true negative)

A_{ij}	M_{ij}
1	1.98
1	1.03
1	0.75
0	0.67
0	0.42
1	0.21
0	0.12
0	0.01
0	0.00



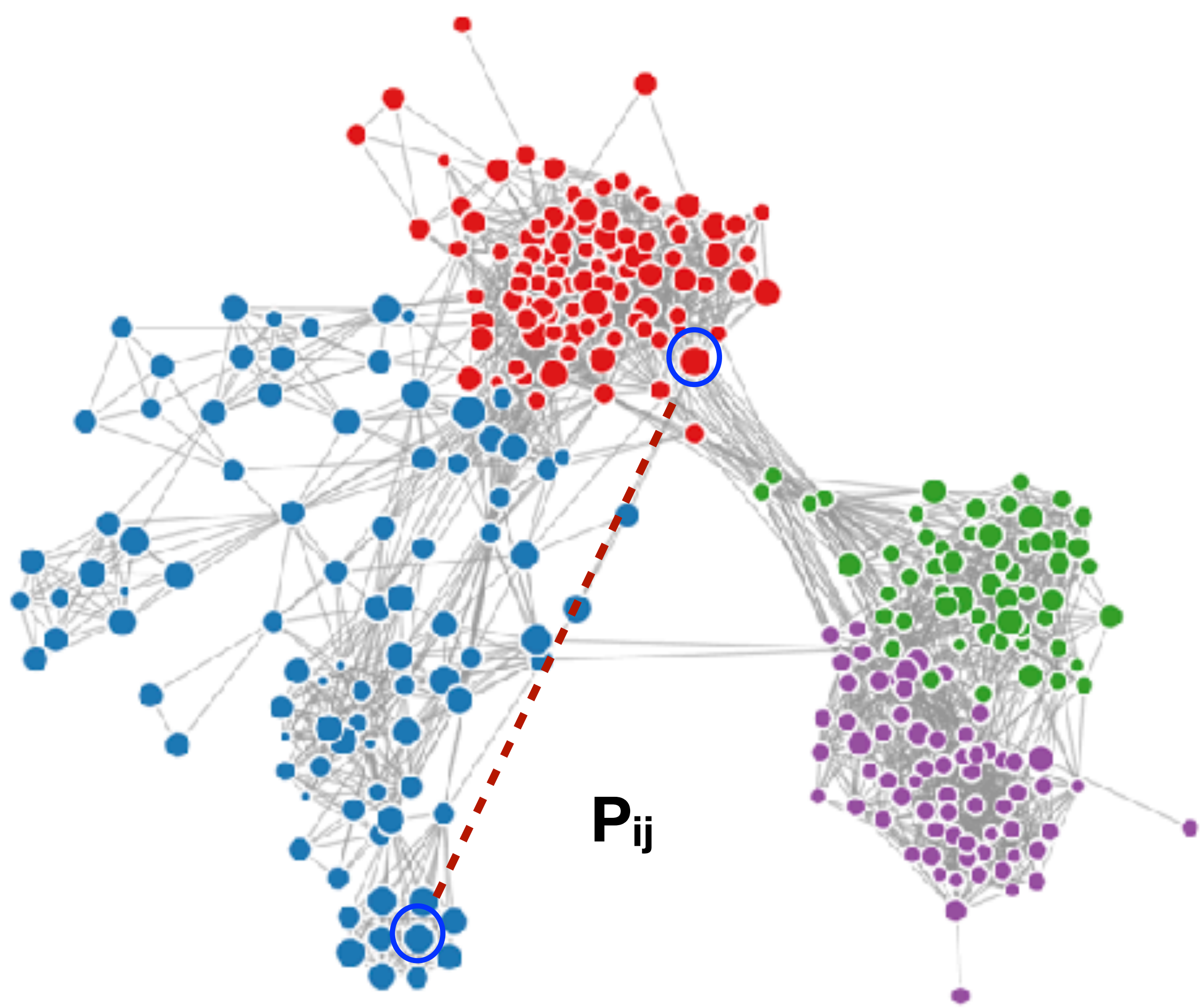
AUC= 1- (#bad)/(#edges x # non-edges)
=1-(1+1)/(4 x 5) = **0.9**

Testing the model: *predictive performance*

Idea: a model is ‘good’ if it’s able to predict the existence of links

AUC (Area under the curve):
the probability that a randomly chosen missing connection (a true positive) is given a higher score than a randomly chosen pair of unconnected vertices (a true negative)

- Does not depend on the details of the model (Poisson, Priors, etc..)
- Compares with random strategy 0.5
- Does not need to tune parameters
- Deals with overfitting

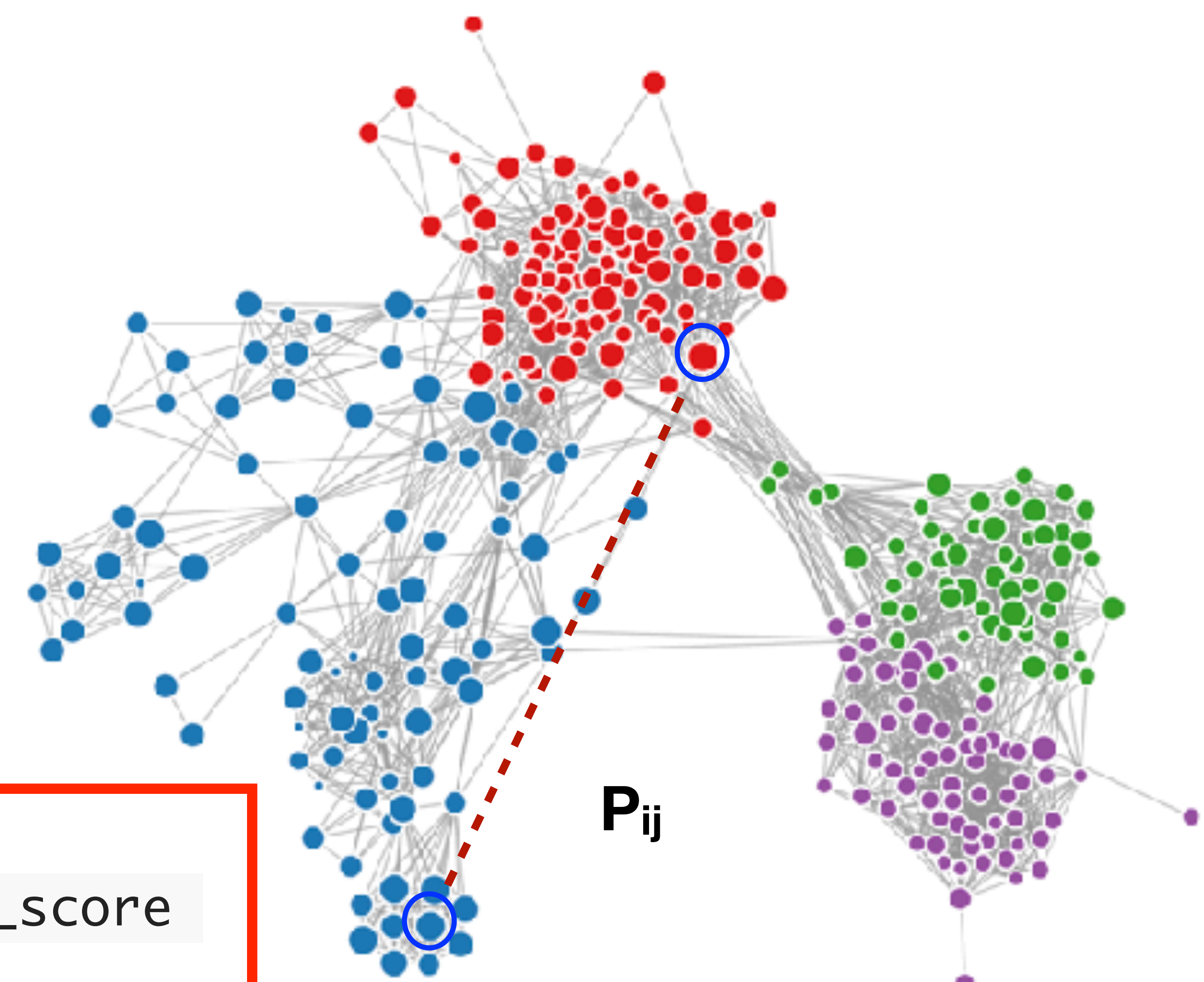


Testing the model: *predictive performance*

Idea: a model is 'good' if it's able to predict the existence of links

AUC (Area under the curve):
the probability that a randomly chosen missing connection (a true positive) is given a higher score than a randomly chosen pair of unconnected vertices (a true negative)

- Does not depend on the details of the model (Poisson, Priors, etc..)
- Compares with random strategy 0.5
- Does not need to tune parameters
- Deals with overfitting

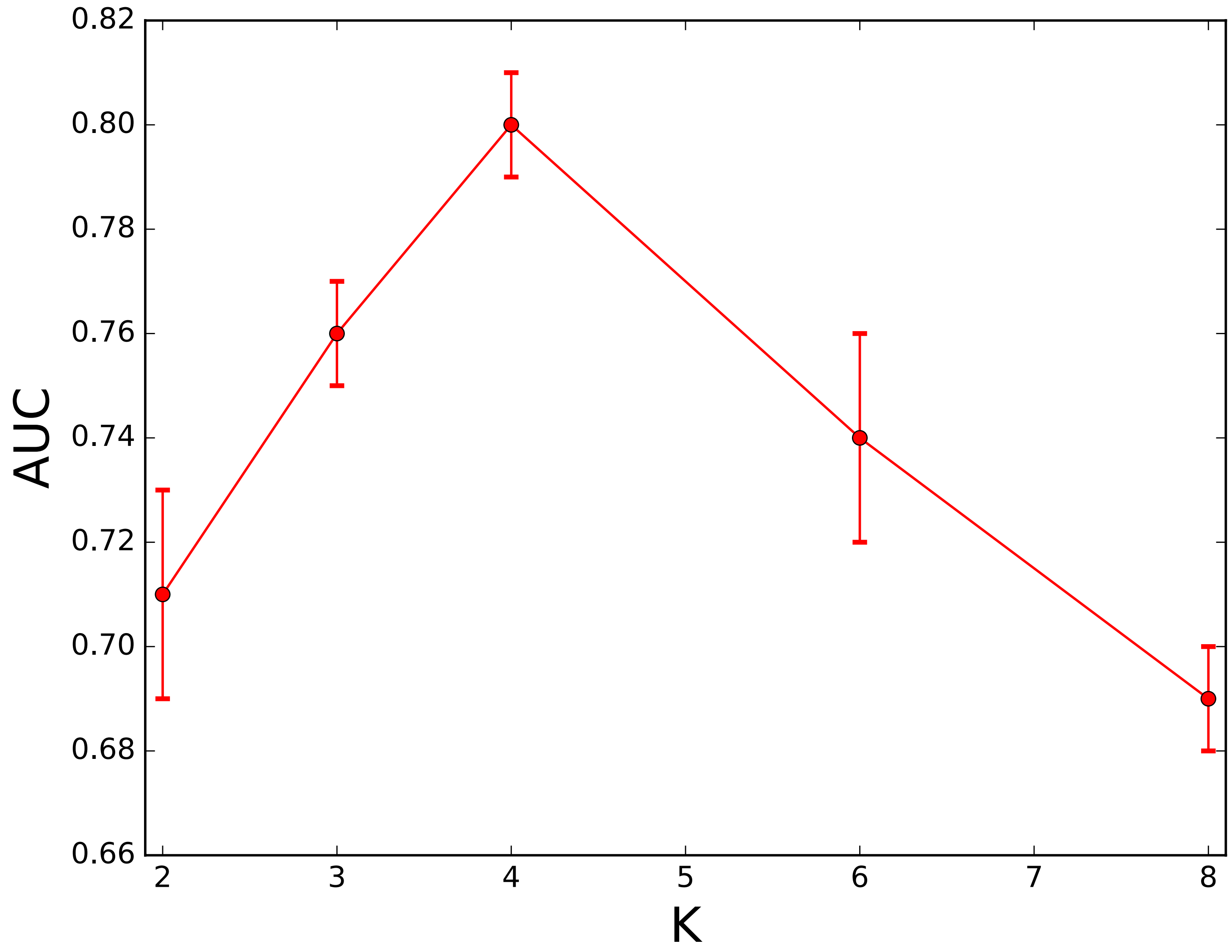



```
>>> import numpy as np
>>> from sklearn.metrics import roc_auc_score
>>> y_true = np.array([0, 0, 1, 1])
>>> y_scores = np.array([0.1, 0.4, 0.35, 0.8])
>>> roc_auc_score(y_true, y_scores)
0.75
```

http://scikit-learn.org/stable/modules/generated/sklearn.metrics.roc_auc_score.html

Testing the model: *predictive performance*

Can also be used as model selection criteria



Other metrics can be used, as held-out log-likelihood, etc…
see Kawamoto et Kabashima 2017

Testing the model: *most plausible*, *highest probability given the data*

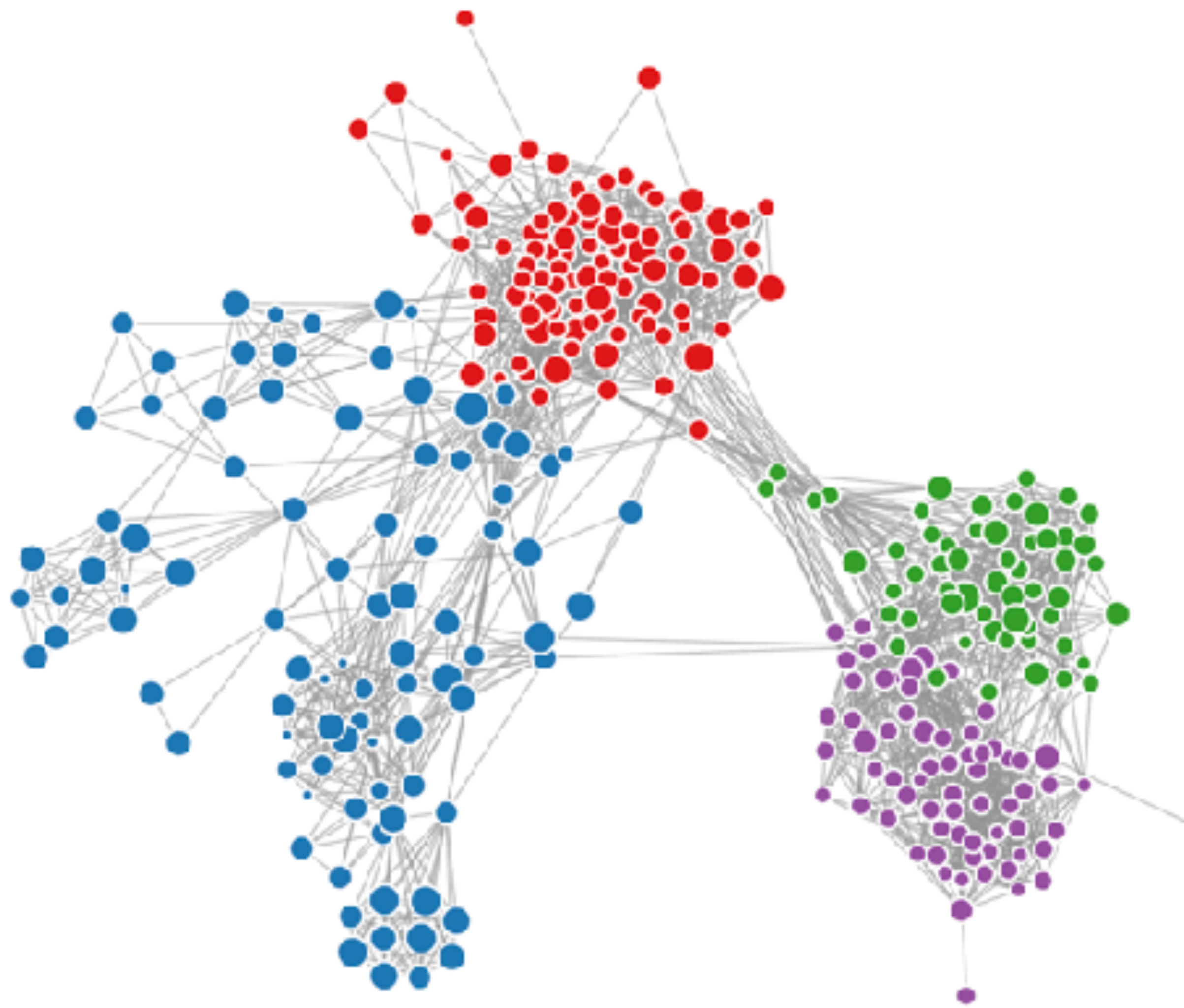
Idea: a model is ‘good’ if it compresses the data the most

MDL (Minimum description length):
Peter D. Grünwald 2007,Rosvall and Bergstrom 2007

$$P(A|\Theta)P(\Theta) = 2^{-\Sigma(A;\Theta)}$$

Posterior: Likelihood x Prior

The description length of a message is:
bits required to send the compressed message + # bits in encoding scheme.



See Vallès-Català et al. 2017 for comparison AUC vs MDL

Testing the model: *most plausible*, *highest probability given the data*

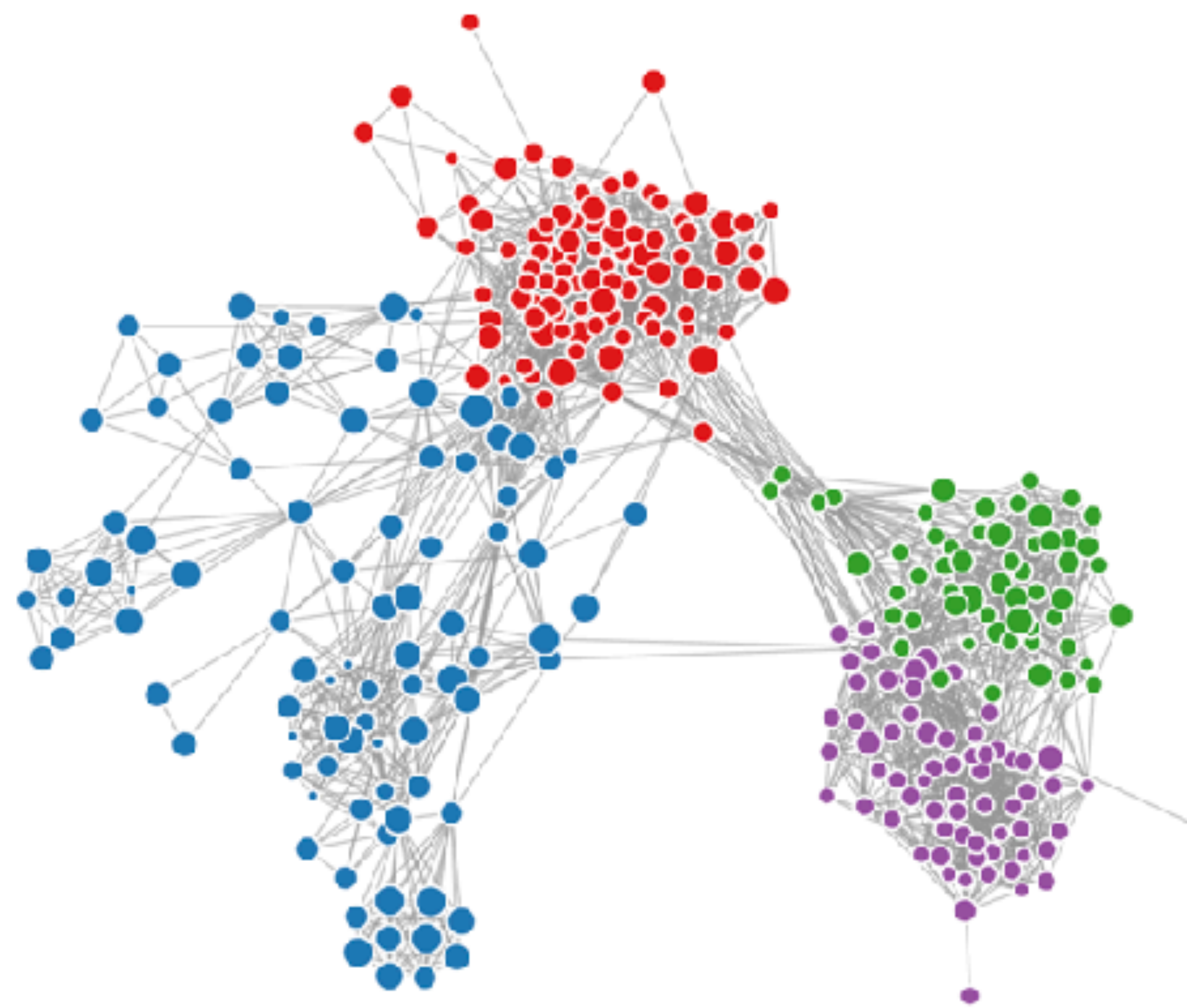
Idea: a model is ‘good’ if it compresses the data the most

MDL (Minimum description length):

Peter D. Grünwald 2007,Rosvall and Bergstrom 2007

$$\boxed{P(A|\Theta)}P(\Theta) = 2^{-\Sigma(A;\Theta)}$$

Posterior: Likelihood x Prior



See Vallès-Català et al. 2017 for comparison AUC vs MDL

Testing the model: *most plausible*, *highest probability given the data*

Idea: a model is 'good' if it compresses the data the most

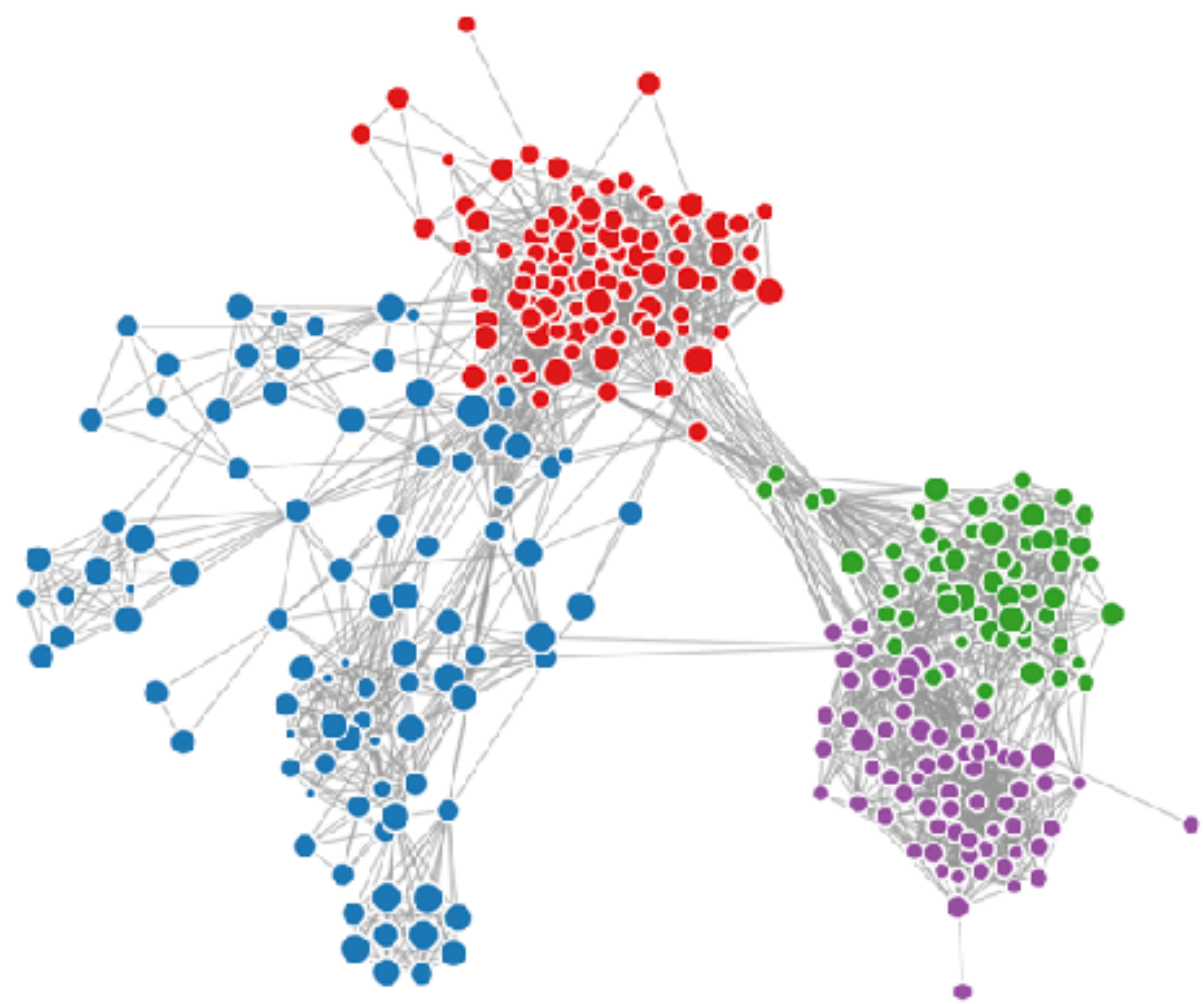
MDL (Minimum description length):

Peter D. Grünwald 2007,Rosvall and Bergstrom 2007

$$\boxed{P(A|\Theta)}P(\Theta) = 2^{-\Sigma(A;\Theta)}$$

Posterior: Likelihood x Prior

- Higher likelihood —> lower MDL



See Vallès-Català et al. 2017 for comparison AUC vs MDL

Testing the model: *most plausible*, *highest probability given the data*

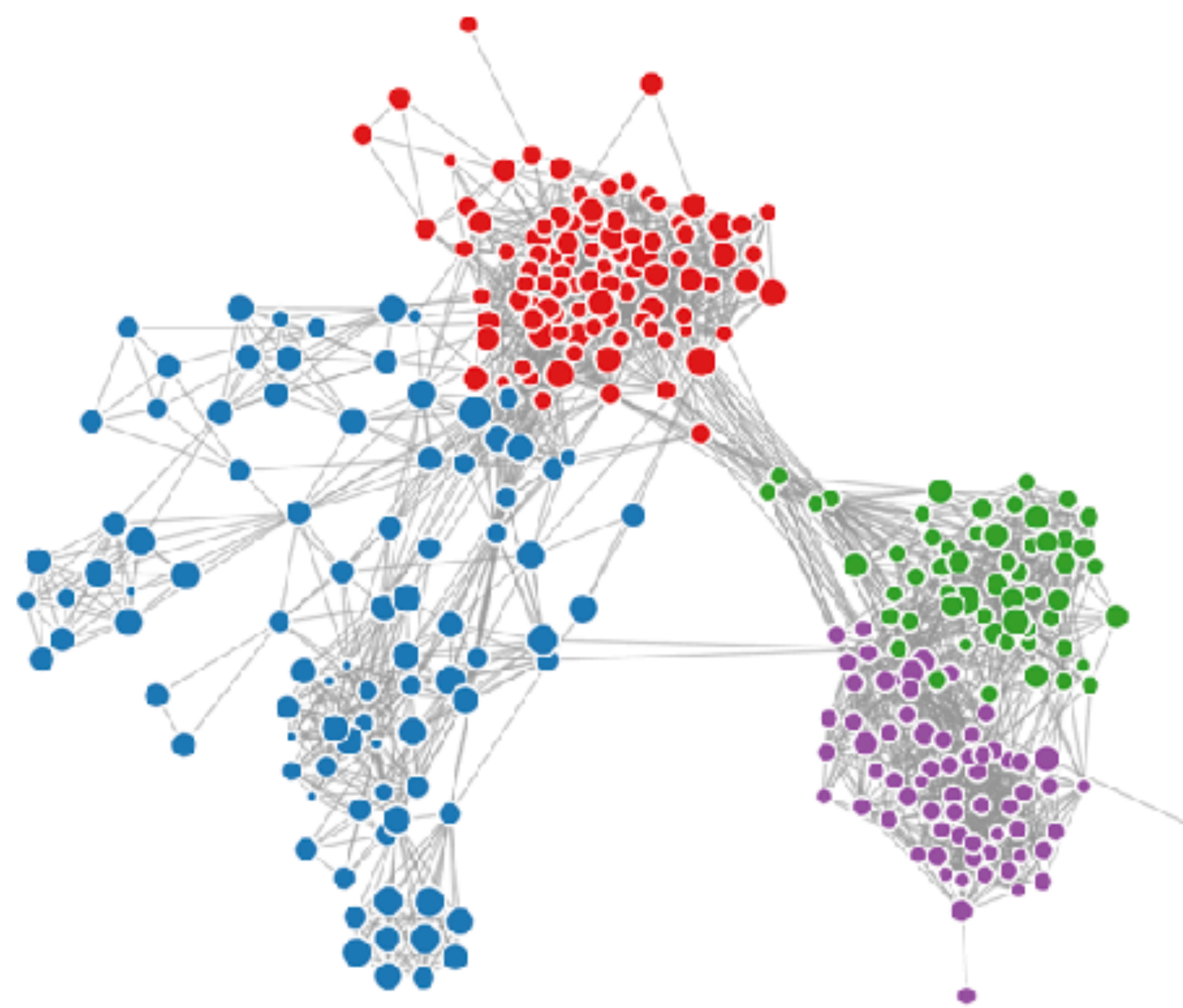
Idea: a model is ‘good’ if it compresses the data the most

MDL (Minimum description length):

Peter D. Grünwald 2007,Rosvall and Bergstrom 2007

$$P(A|\Theta)\boxed{P(\Theta)} = 2^{-\Sigma(A;\Theta)}$$

Posterior: Likelihood x Prior



See Vallès-Català et al. 2017 for comparison AUC vs MDL

Testing the model: *most plausible*, *highest probability given the data*

Idea: a model is 'good' if it compresses the data the most

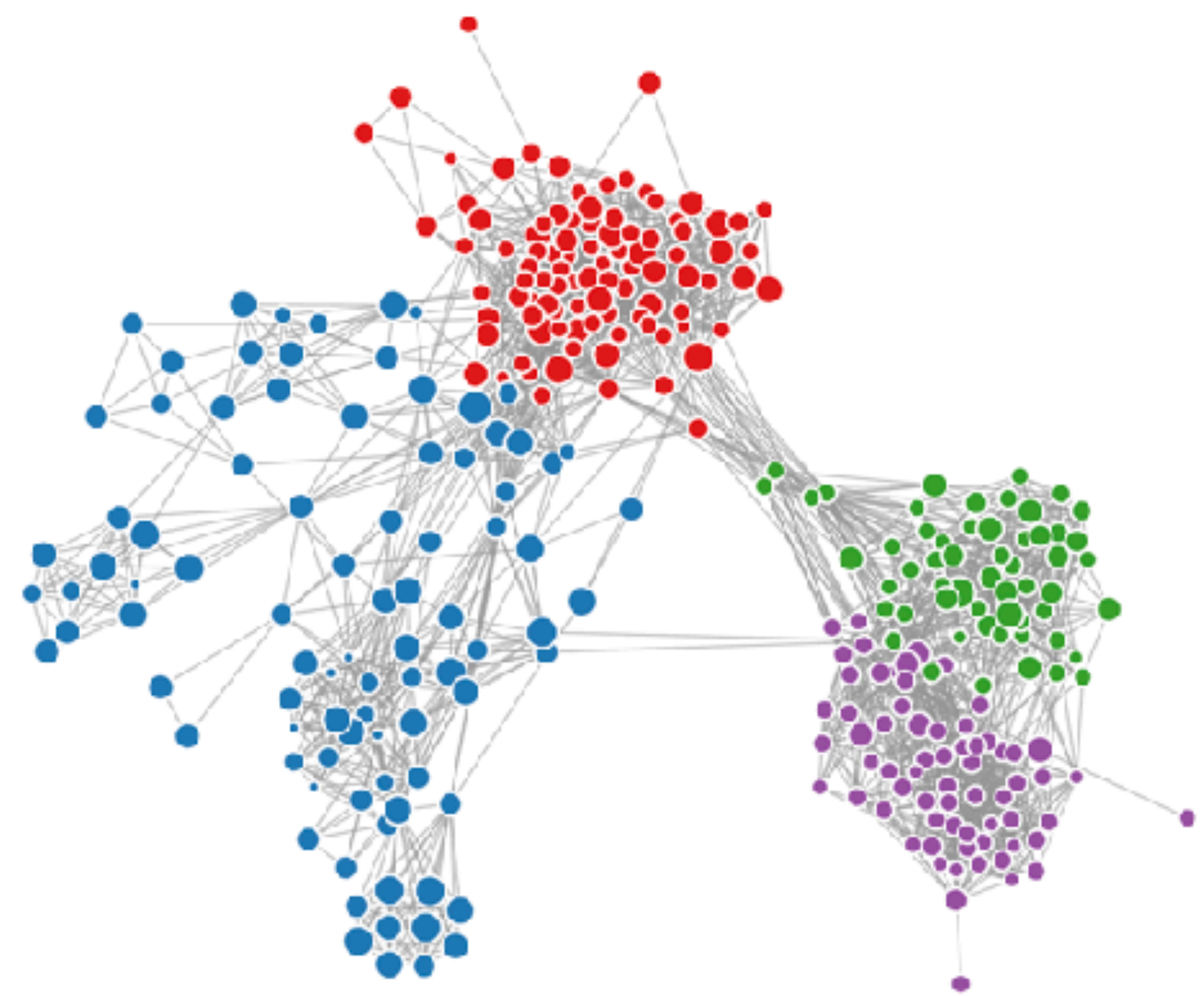
MDL (Minimum description length):

Peter D. Grünwald 2007,Rosvall and Bergstrom 2007

$$P(A|\Theta)\boxed{P(\Theta)} = 2^{-\Sigma(A;\Theta)}$$

Posterior: Likelihood x Prior

- Higher likelihood —> lower MDL
- More parameters —> smaller prior —> bigger MDL



See Vallès-Català et al. 2017 for comparison AUC vs MDL

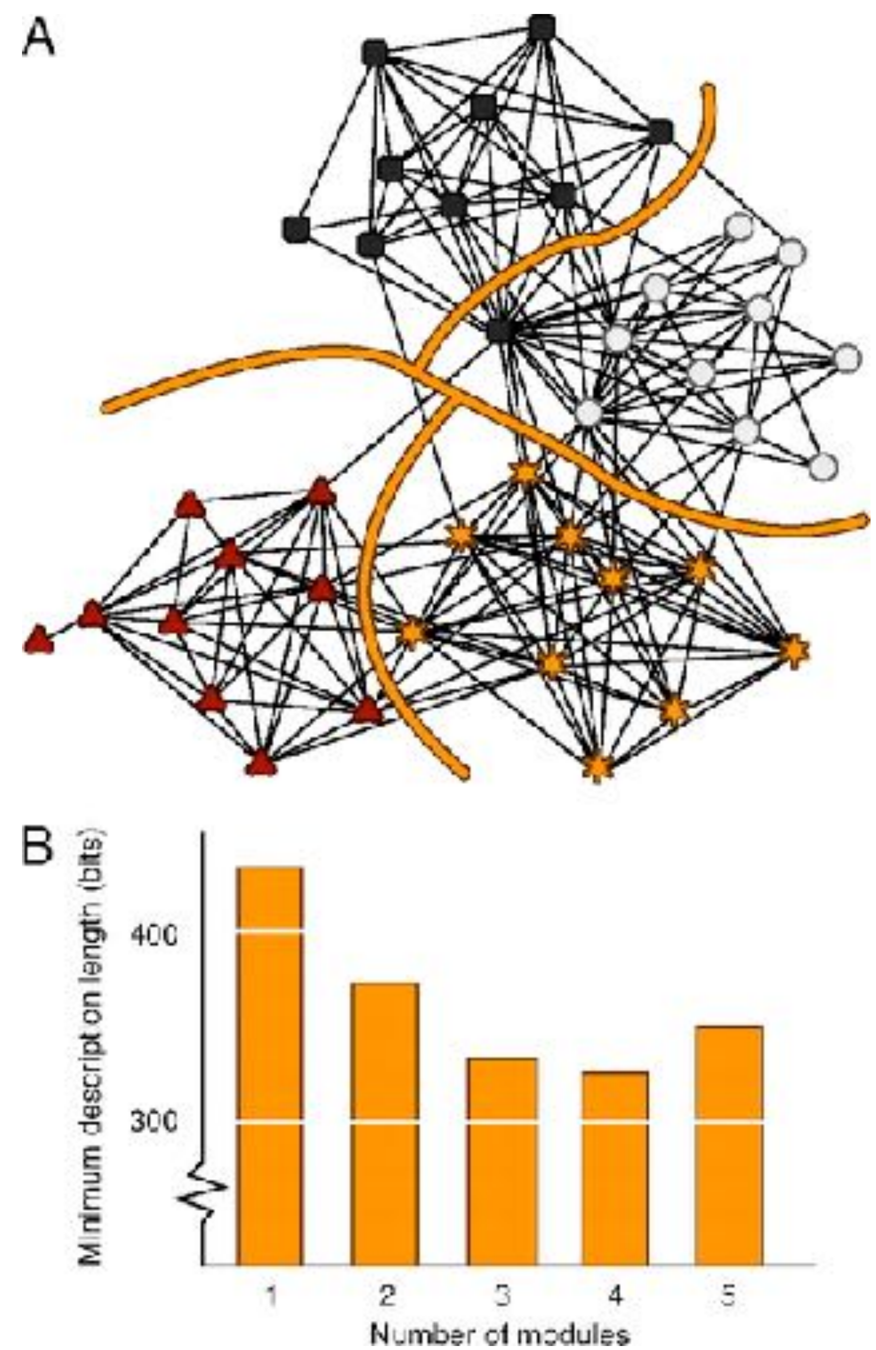
Testing the model: *most plausible*, *highest probability given the data*

Often used as model selection criteria

$B = 5$,
non-degree-corrected,
overlapping, $\Lambda = 1$

$B = 4$,
non-degree-corrected,
overlapping, $\Lambda \simeq 2 \times 10^{-4}$,

TP Peixoto PRX 2015, PRX 2014



Rosvall and Bergstrom 2007

- **If you have ground truth:** several methods are available, metrics used in statistical learning theory (F1, NMI, etc…)
    - Always good to generate synthetic data and ‘plant’ ground truth
    - Metadata should used carefully
- **Otherwise**: predictive (AUC) vs MDL approaches

1. **Testing the model (20mins)**
    - Measuring correlation with ‘ground truth’: F1score, Mutual Information
    - No ground truth: AUC and MDL
2. Analysing the results (20 mins)
    - Normalize and extract max group
    - Visualize results
    - Statistical significance: many local optima
3. Advanced topics (10mins): Metadata

Analysing the results: *process parameters*

Analysing the results: *process parameters*

U_i

```
0 0 0 0.0790622 0
3 0.0533999 0 0.139296 0
76 0 0 0.0805629 0.10241
127 0 0 0 0.207446
177 0 0.0738358 0.158358 0.024311
1 0 0 0.185704 0.0796013
6 0.0270201 0 0 0.0864358
17 0 0 0.192816 0.0215193
34 0 0 0.211593 0.0223843
```

V_i

```
0 0 0.00483982 0.00224297 0
3 0 0.01271 0.00688526 0
76 0 0.00115951 0.0137771 0.00347393
127 0 0.00605217 0.0168567 0
177 0.00267315 0 0.00549152 0.0149187
1 0 0.017335 0.00262135 0.0033609
6 0.00186055 0 0 0.00820299
17 0 0.0184664 0 0
34 0 0.0183341 0 0.00153079
```

W^a

```
a= 0
0 0 0 2.66716
0 3.79189 0 0
0 0 2.75449 0
2.0515 0 0 0

a= 1
0 0 0 1.30754
0.031726 1.07808 0.0602251 0
0.271431 0.256695 1.07306 0
0.750644 0 0 0

a= 2
0 0 0 1.98783
0 2.65475 0 0
0 0 1.82214 0
1.27939 0 0 0

a= 3
0 0 0 2.74153
0 1.96755 0.0878885 0
0 0 3.99113 0
1.93712 0 0 0
```

Analysing the results: *process parameters*

U_i

```
0 0 0 0.0790622 0
3 0.0533999 0 0.139296 0
76 0 0 0.0805629 0.10241
127 0 0 0 0.207446
177 0 0.0738358 0.158358 0.024311
1 0 0 0.185704 0.0796013
6 0.0270201 0 0 0.0864358
17 0 0 0.192816 0.0215193
34 0 0 0.211593 0.0223843
```

V_i

```
0 0 0.00483982 0.00224297 0
3 0 0.01271 0.00688526 0
76 0 0.00115951 0.0137771 0.00347393
127 0 0.00605217 0.0168567 0
177 0.00267315 0 0.00549152 0.0149187
1 0 0.017335 0.00262135 0.0033609
6 0.00186055 0 0 0.00820299
17 0 0.0184664 0 0
34 0 0.0183341 0 0.00153079
```

W^a

```
a= 0
0 0 0 2.66716
0 3.79189 0 0
0 0 2.75449 0
2.0515 0 0 0

a= 1
0 0 0 1.30754
0.031726 1.07808 0.0602251 0
0.271431 0.256695 1.07306 0
0.750644 0 0 0

a= 2
0 0 0 1.98783
0 2.65475 0 0
0 0 1.82214 0
1.27939 0 0 0

a= 3
0 0 0 2.74153
0 1.96755 0.0878885 0
0 0 3.99113 0
1.93712 0 0 0
```

Analysing the results: *process parameters*

U_i

```
0 0 0 1.0 0
3 0.25 0 0.75 0
76 0 0 0.4 0.6
127 0 0 0 1.0
177 0 0.4 0.55 0.05
1 0 0 0.7 0.3
6 0.2 0 0 0.8
17 0 0 0.9 0.1
34 0 0 0.95 0.05
```

```
0 0 0 0.0790622 0
3 0.0533999 0 0.139296 0
76 0 0 0.0805629 0.10241
127 0 0 0 0.207446
177 0 0.0738358 0.158358 0.024311
1 0 0 0.185704 0.0796013
6 0.0270201 0 0 0.0864358
17 0 0 0.192816 0.0215193
34 0 0 0.211593 0.0223843
```

- Normalize vectors

Analysing the results: *process parameters*

- Normalize vectors
- Form communities

U_i

0	0	0	1.0	0
3	0.25	0	0.75	0
76	0	0	0.4	0.6
127	0	0	0	1.0
177	0	0.4	0.55	0.05
1	0	0	0.7	0.3
6	0.2	0	0	0.8
17	0	0	0.9	0.1
34	0	0	0.95	0.05

3 6 …
177 …
3 177 1 17 34…
76 127 177 1 6 17 34…

Analysing the results: *process parameters*

- Normalize vectors
- Form communities
  - Take the max only: hard partition

U_i

0	0	0	1.0	0
3	0.25	0	0.75	0
76	0	0	0.4	0.6
127	0	0	0	1.0
177	0	0.4	0.55	0.05
1	0	0	0.7	0.3
6	0.2	0	0	0.8
17	0	0	0.9	0.1
34	0	0	0.95	0.05

```
0 3 6 177 1 17 34…
76 127 6 …
…
```

Analysing the results: *process parameters*

U_i

0	0	0	1.0	0
3	0.25	0	0.75	0
76	0	0	0.4	0.6
127	0	0	0	1.0
177	0	0.4	0.55	0.05
1	0	0	0.7	0.3
6	0.2	0	0	0.8
17	0	0	0.9	0.1
34	0	0	0.95	0.05

- Normalize vectors
- **Form communities**
  - Take the max only: hard partition
  - Threshold membership

```
0 3 76 177 1 17 34 …
177 …
76 127 1 6 …
…
```

Analysing the results: *process parameters*

- Normalize vectors
- **Form communities**
  - Take the max only: hard partition
  - Threshold membership

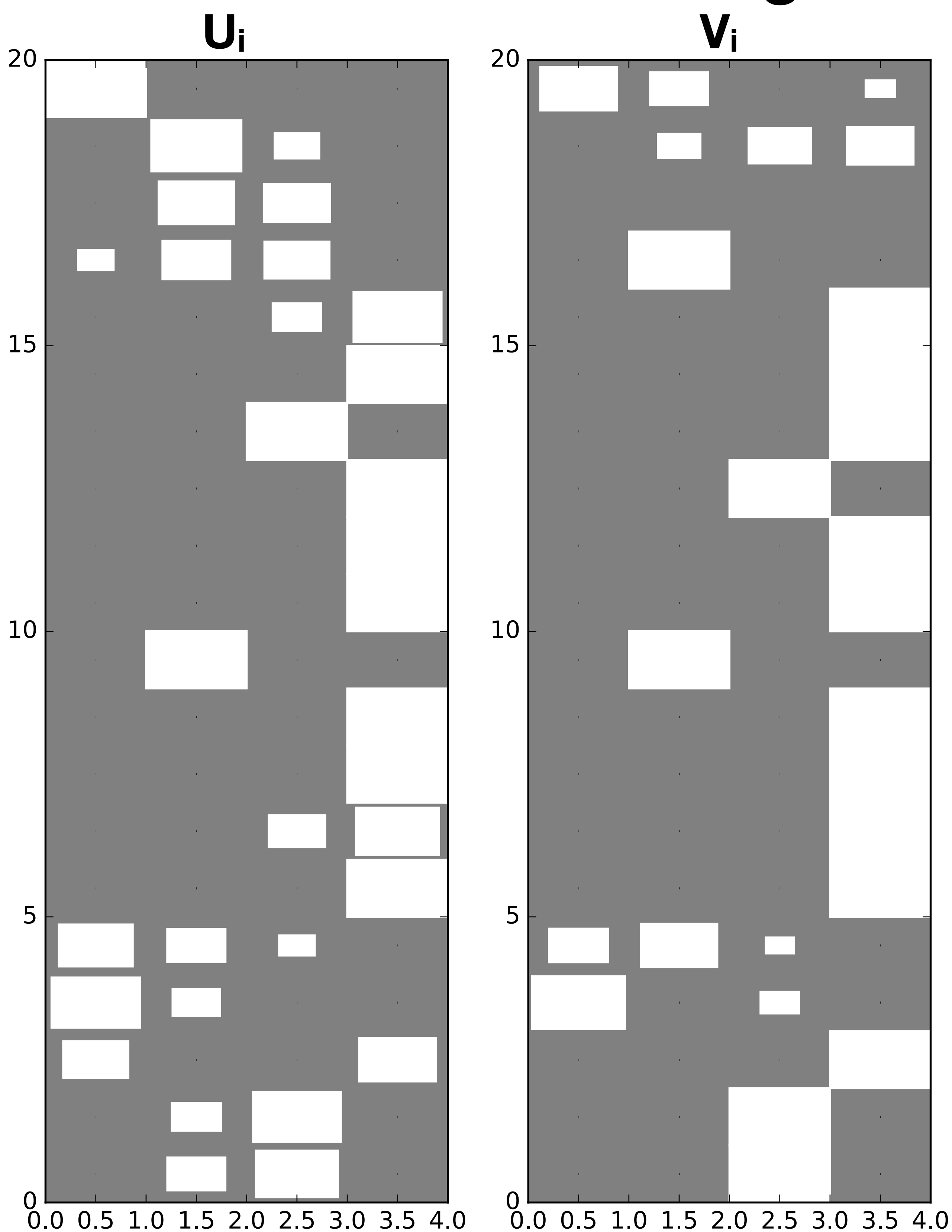
U_i

0	0	0	1.0	0
3	0.25	0	0.75	0
76	0	0	0.4	0.6
127	0	0	0	1.0
177	0	0.4	0.55	0.05
1	0	0	0.7	0.3
6	0.2	0	0	0.8
17	0	0	0.9	0.1
34	0	0	0.95	0.05

```
3 6 …
177 …
3 177 1 17 34…
76 127 177 1 6 17 34…
```

Lose the info on how much does a node belong to each community

Visualization: *Hinton diagrams*

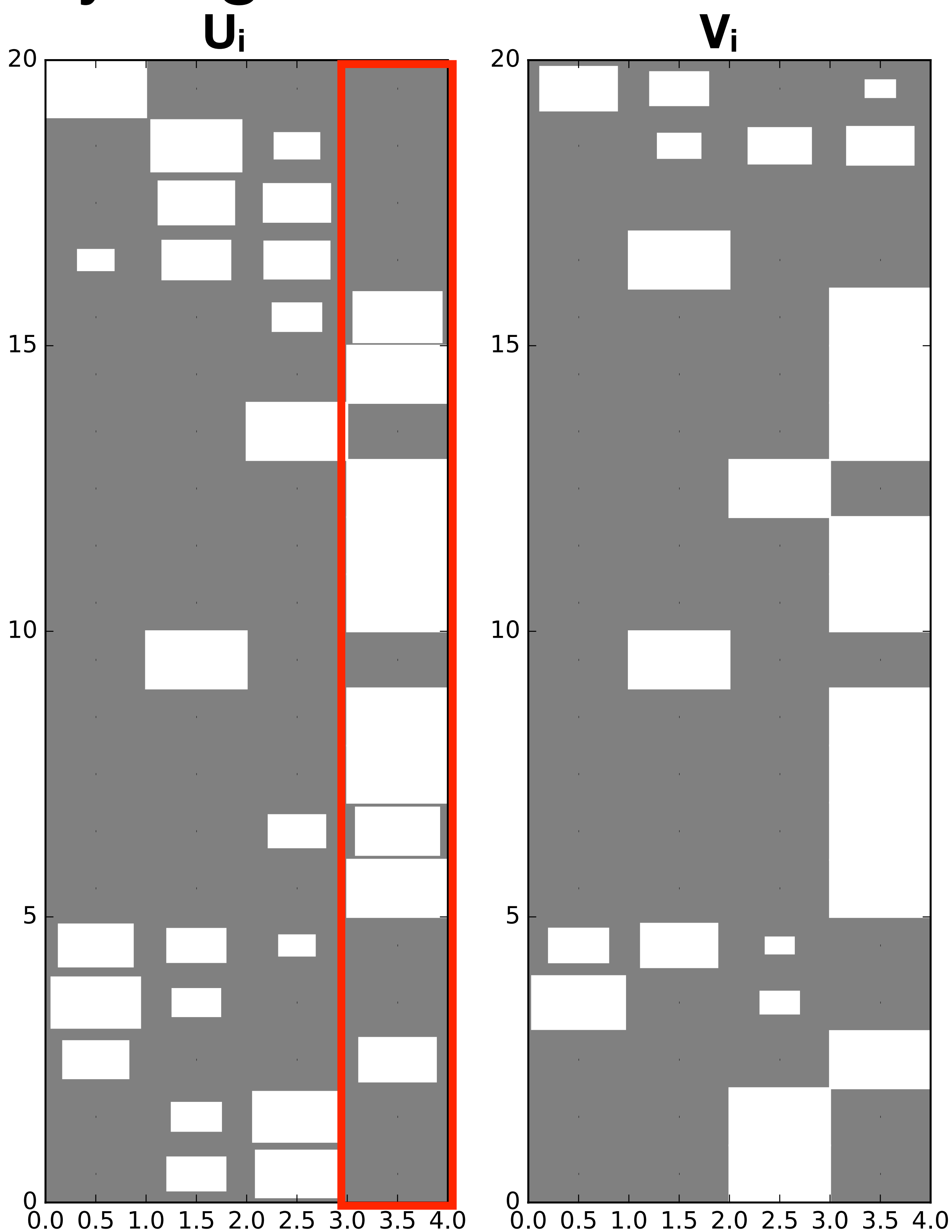


- Normalize vectors
- Form communities
  - Take the max only: hard partition
  - Threshold membership
- **Visualize**
  - Hinton diagram

U_i

0	0	0	1.0	0
3	0.25	0	0.75	0
76	0	0	0.4	0.6
127	0	0	0	1.0
177	0	0.4	0.55	0.05
1	0	0	0.7	0.3
6	0.2	0	0	0.8
17	0	0	0.9	0.1
34	0	0	0.95	0.05

Analysing the results: *visualization*

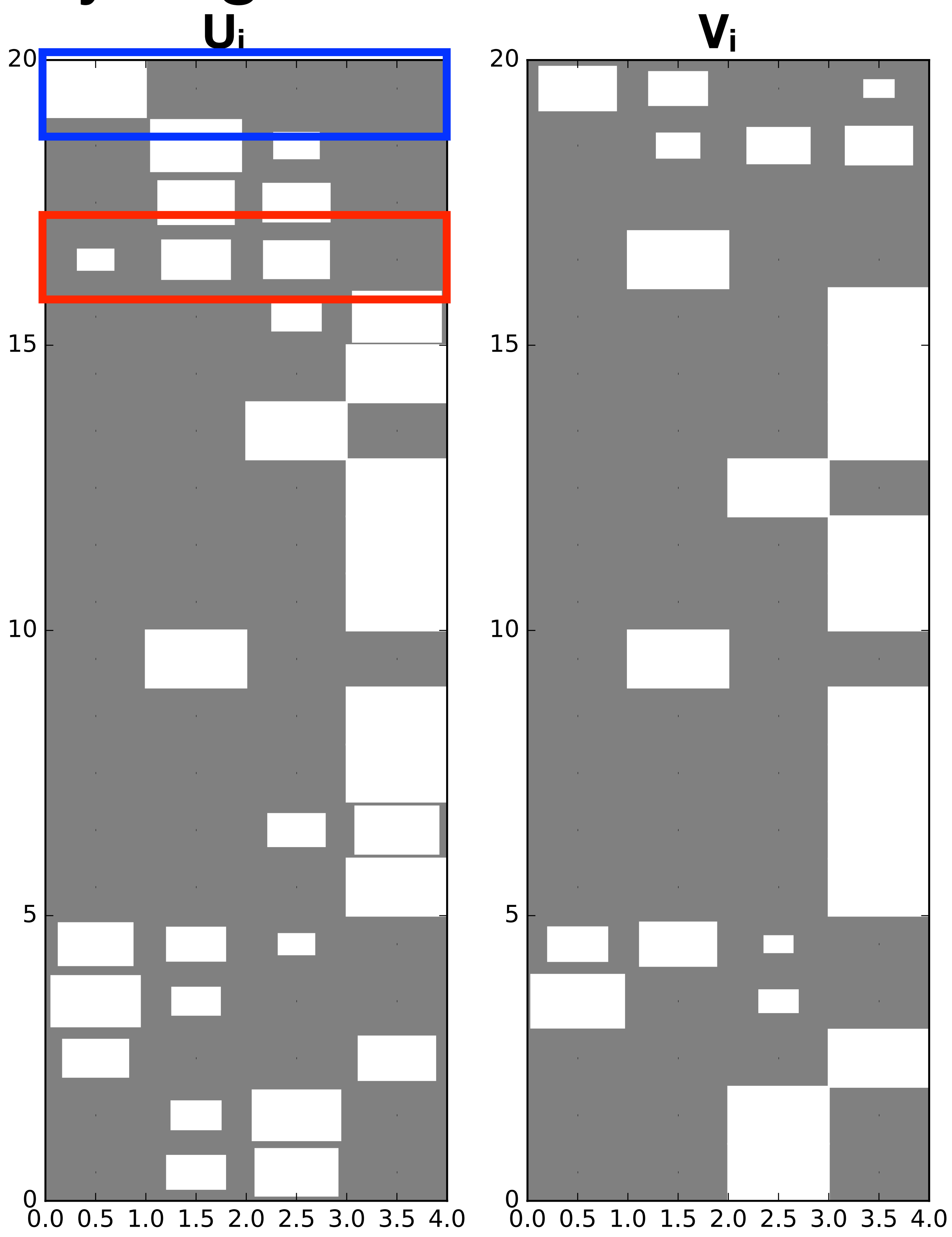


- Normalize vectors
- Form communities
  - Take the max only: hard partition
  - Threshold membership
- **Visualize**
  - **Hinton diagram**

U_i

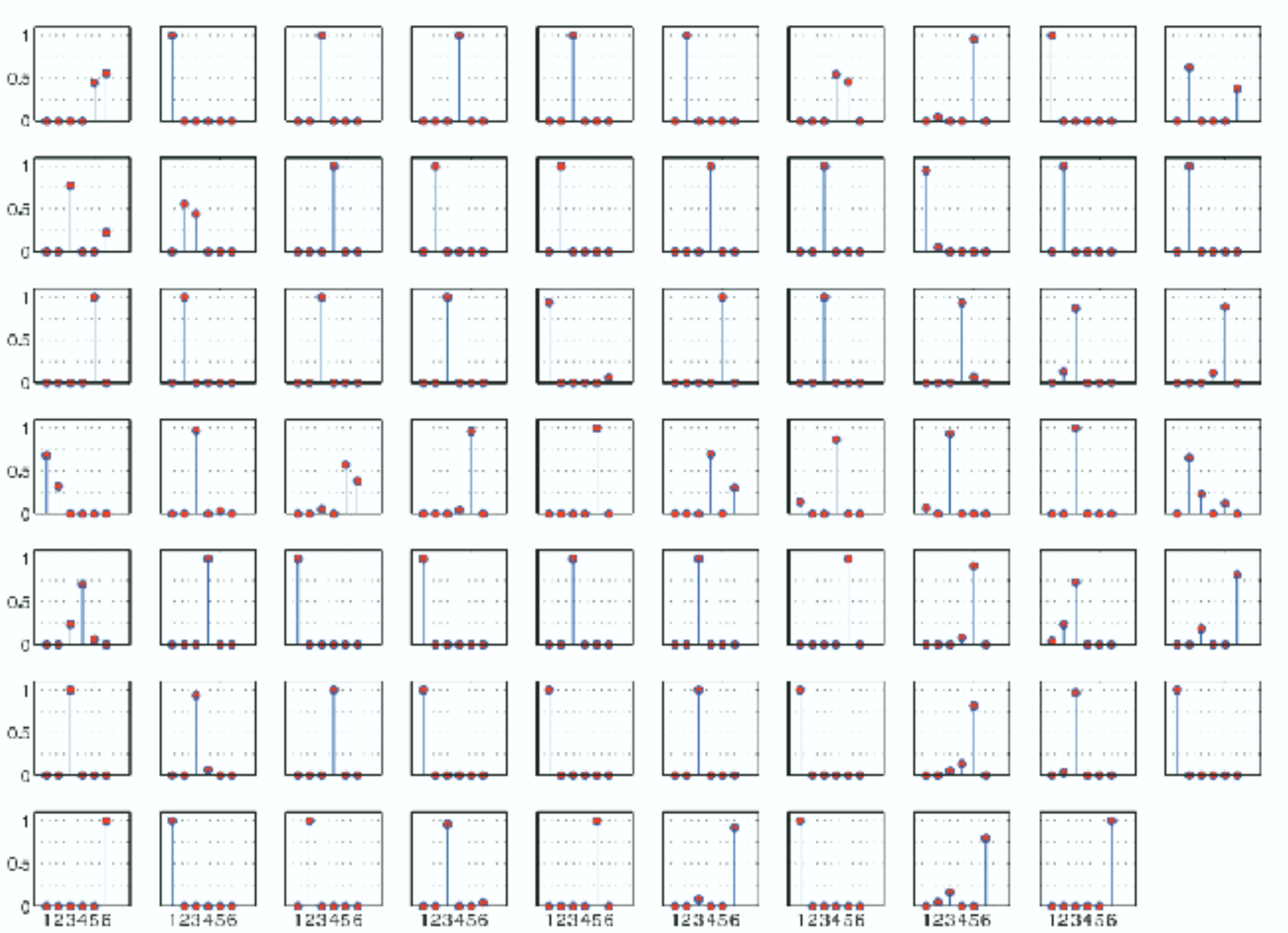
0	0	0	1.0	0
3	0.25	0	0.75	0
76	0	0	0.4	0.6
127	0	0	0	1.0
177	0	0.4	0.55	0.05
1	0	0	0.7	0.3
6	0.2	0	0	0.8
17	0	0	0.9	0.1
34	0	0	0.95	0.05

Analysing the results: *visualization*



- Normalize vectors
- Form communities
  - Take the max only: hard partition
  - Threshold membership
- **Visualize**
  - Hinton diagram

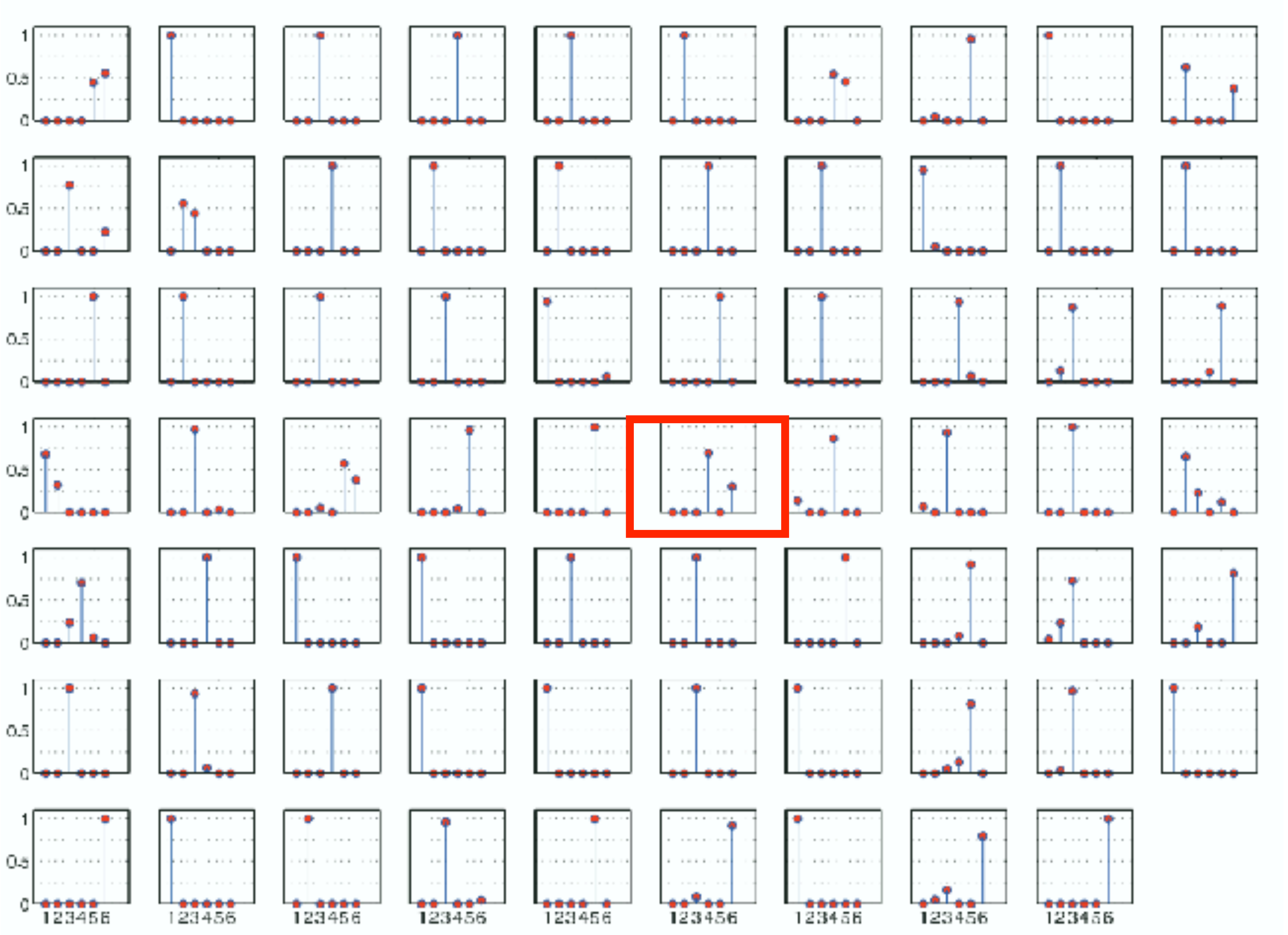
Analysing the results: *visualization*



Airoldi et al. 2008

- Normalize vectors
- Form communities
  - Take the max only: hard partition
  - Threshold membership
- Visualize
  - Hinton diagram
  - Per node

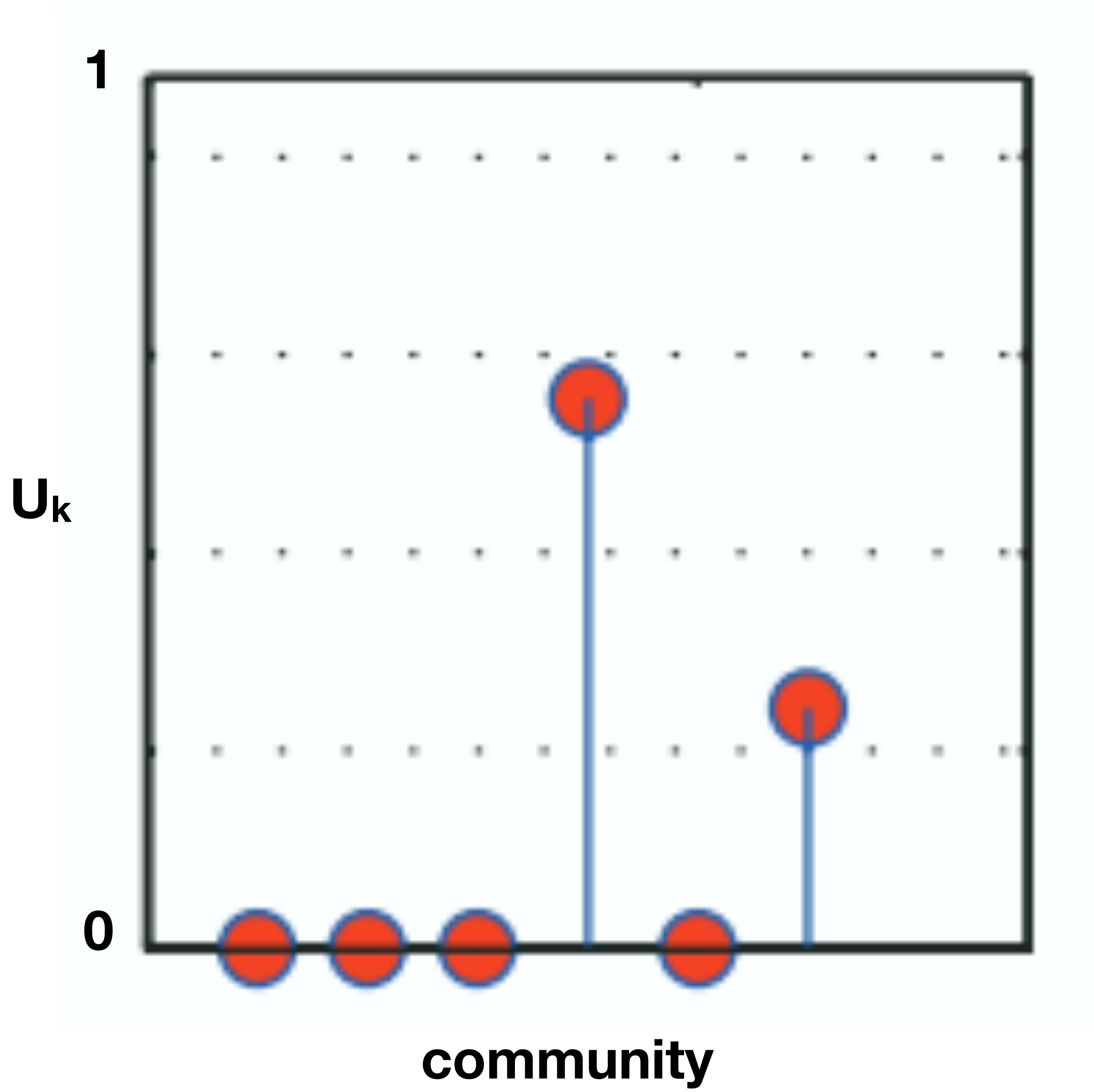
Analysing the results: *visualization*



Airoldi et al. 2008

- Normalize vectors
- Form communities
  - Take the max only: hard partition
  - Threshold membership
- **Visualize**
  - Hinton diagram
  - Per node

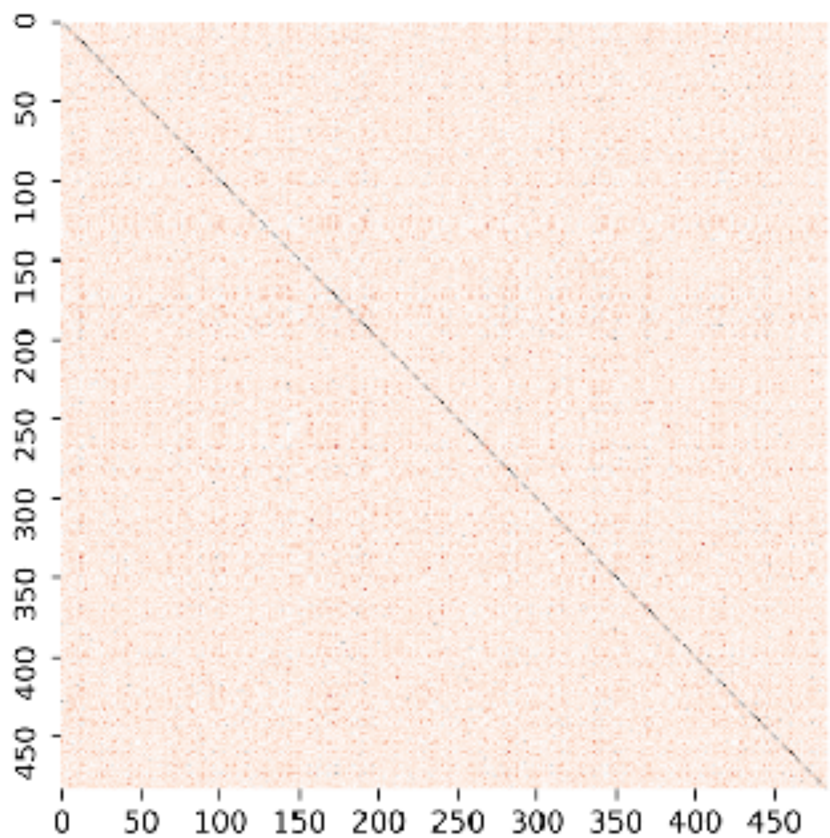
Analysing the results: *visualization*



- Normalize vectors
- Form communities
  - Take the max only: hard partition
  - Threshold membership
- Visualize
  - Hinton diagram
  - Per node

Airoldi et al. 2008

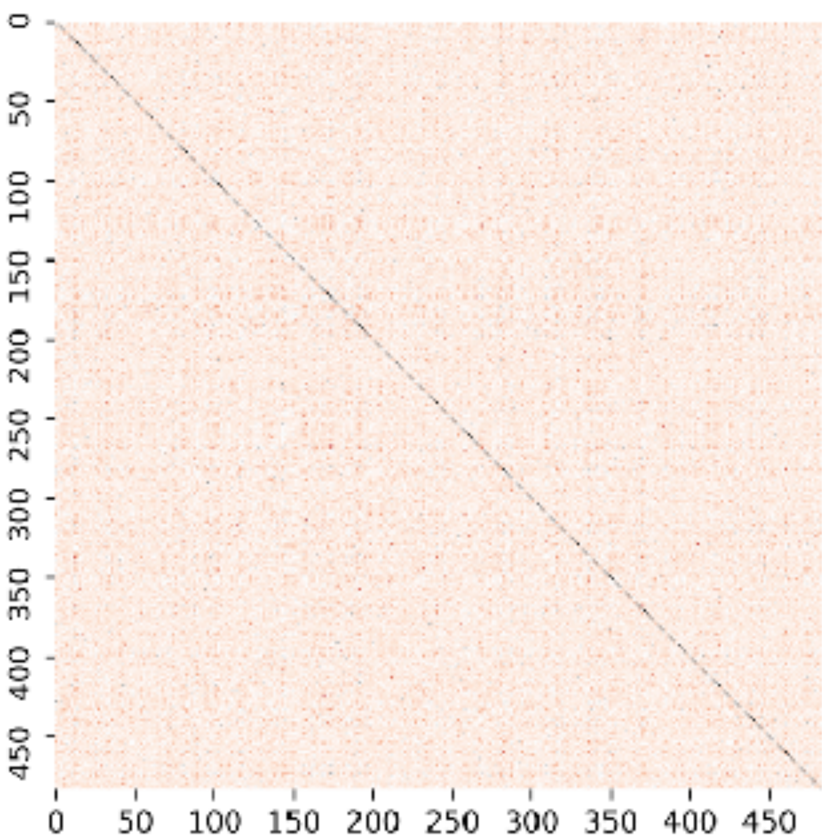
Analysing the results: *visualization*

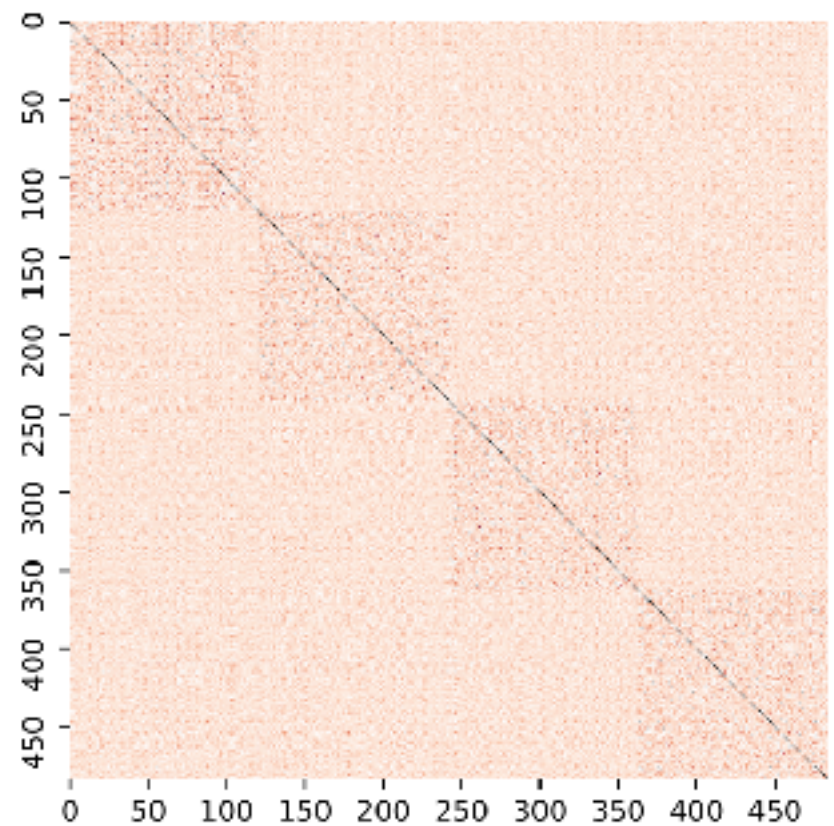


- Normalize vectors
- Form communities
  - Take the max only: hard partition
  - Threshold membership
- Visualize
  - Hinton diagram
  - Per node
  - Adjacency matrix

Analysing the results: *visualization*

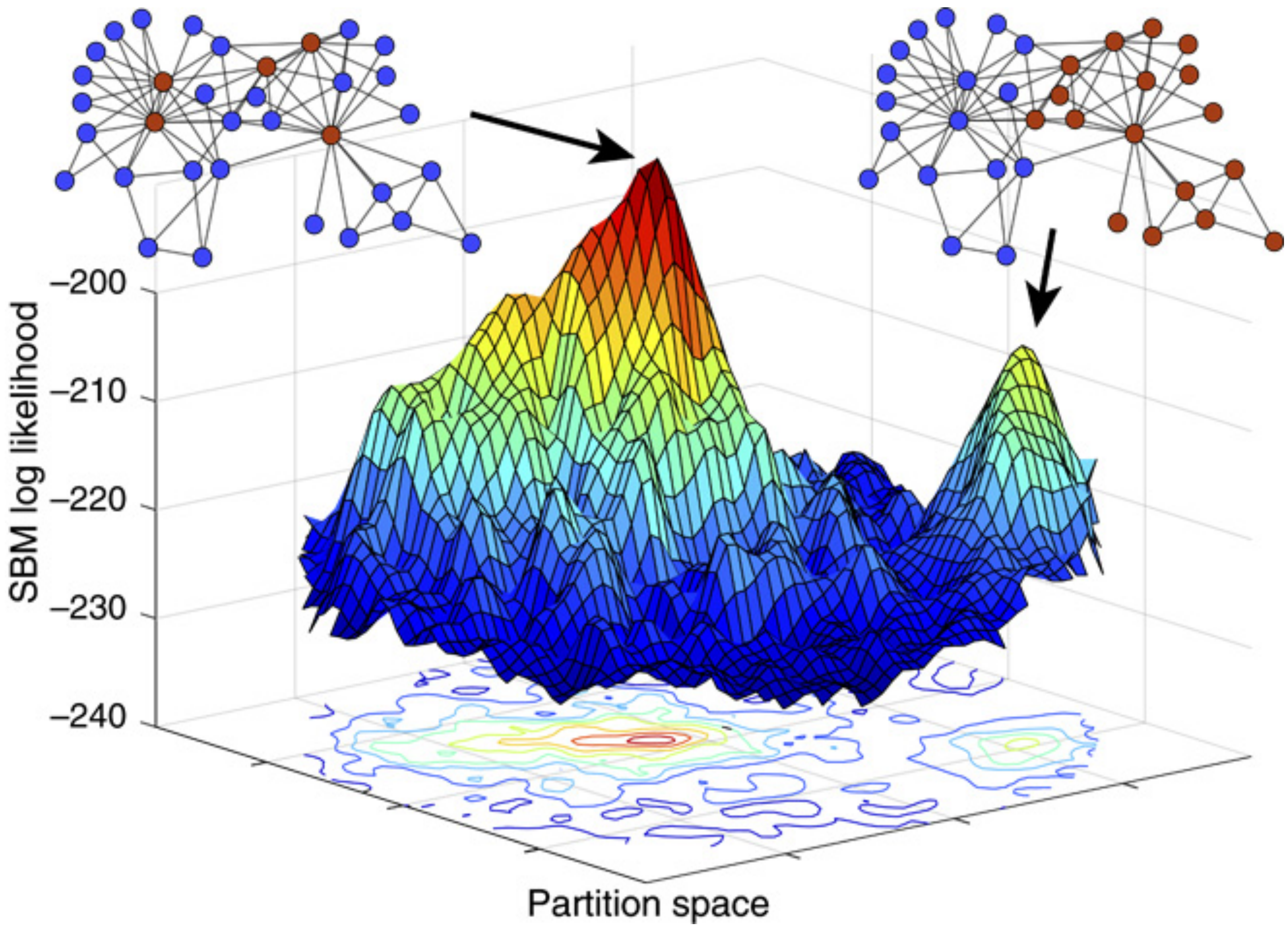
- Normalize vectors
- Form communities
  - Take the max only: hard partition
  - Threshold membership
- Visualize
  - Hinton diagram
  - Per node
  - Adjacency matrix





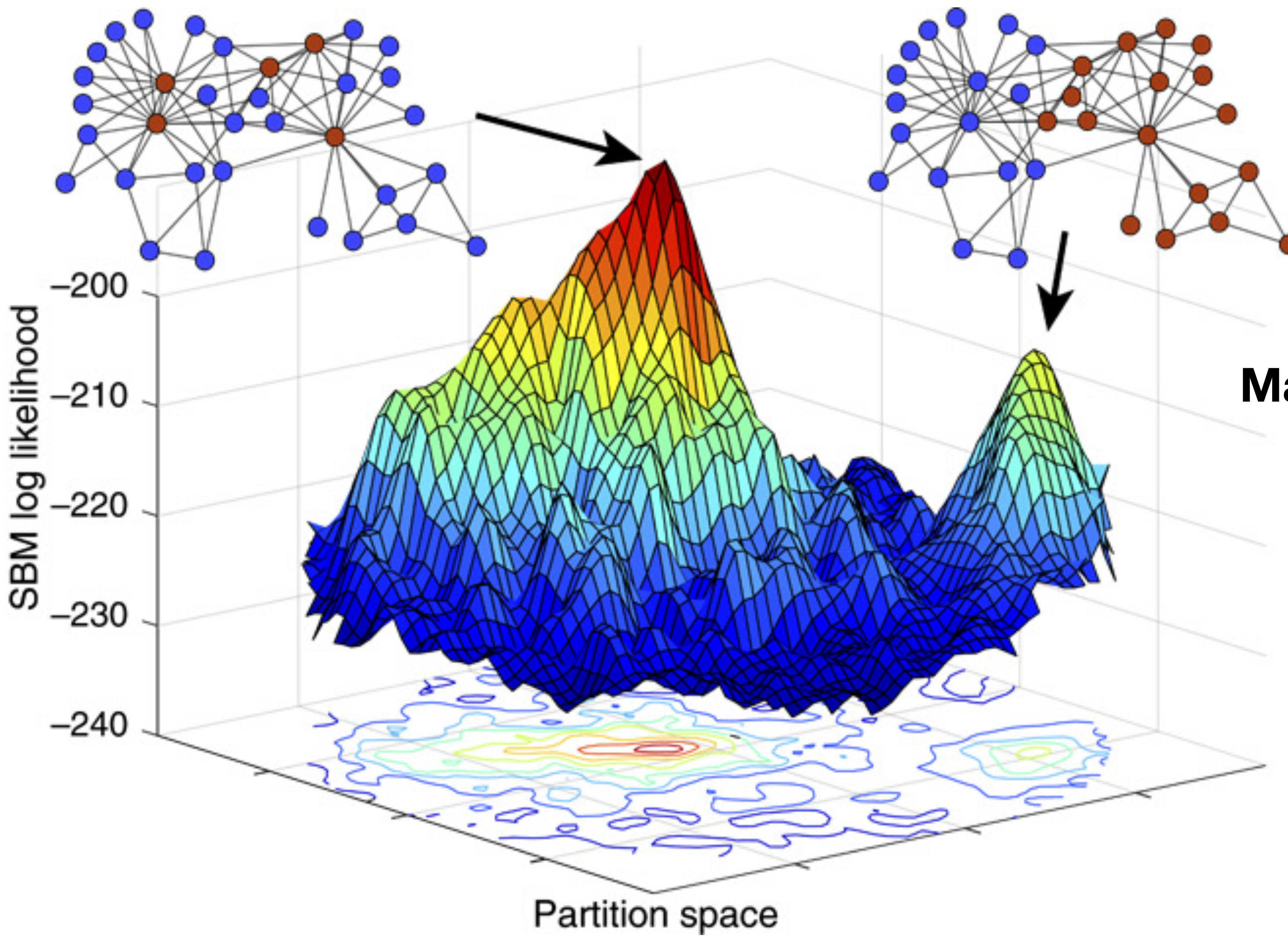
Rows have been reordered by the community membership: i.e. $node_1$-$node_{ck}$ are the nodes belonging to community 1 of dimension ck

Analysing the results: *statistical significance*



Peel et al. 2017

Analysing the results: *statistical significance*

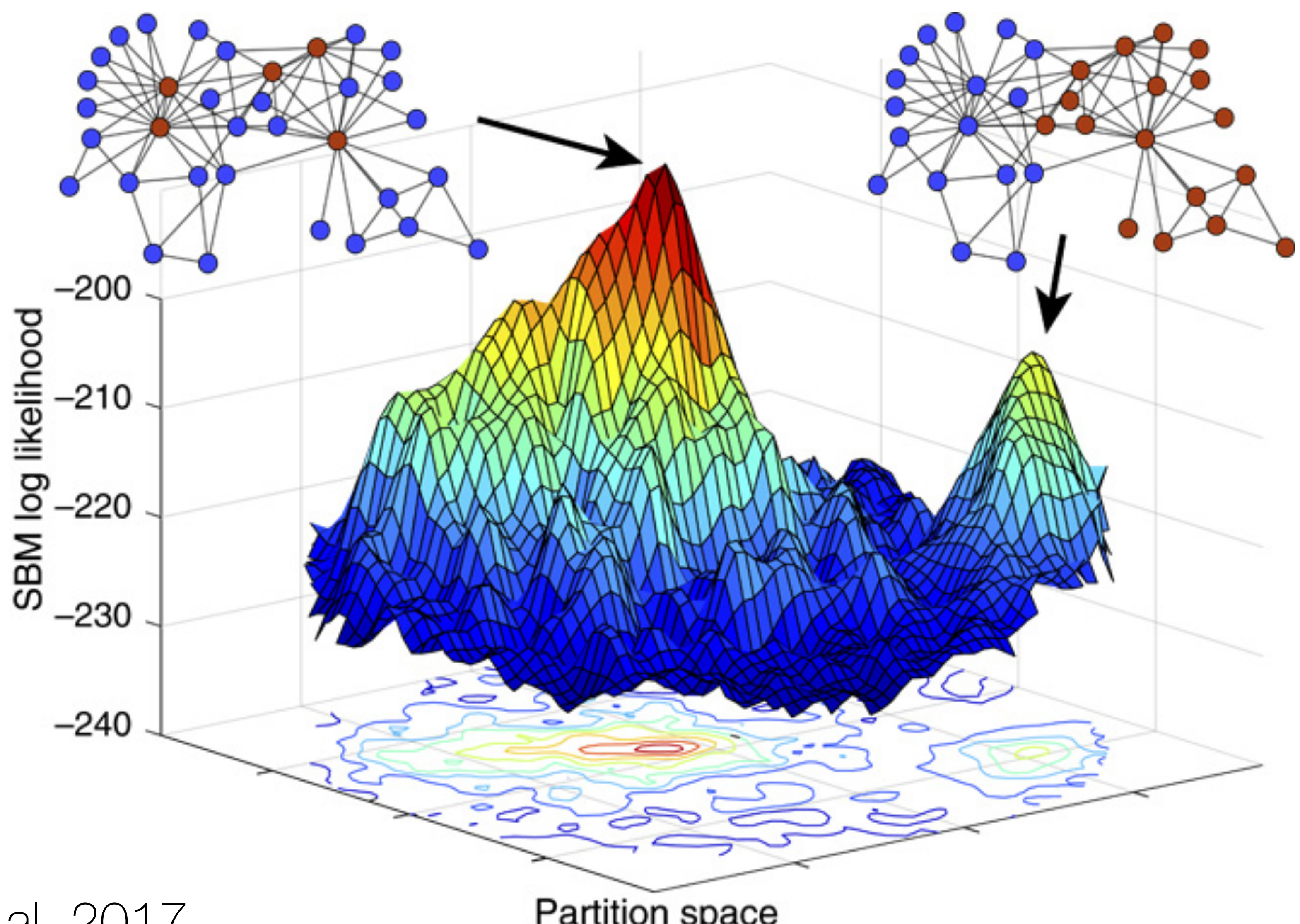


Many local optima!

Peel et al. 2017

Statistical significance: *Maximization vs Sampling*

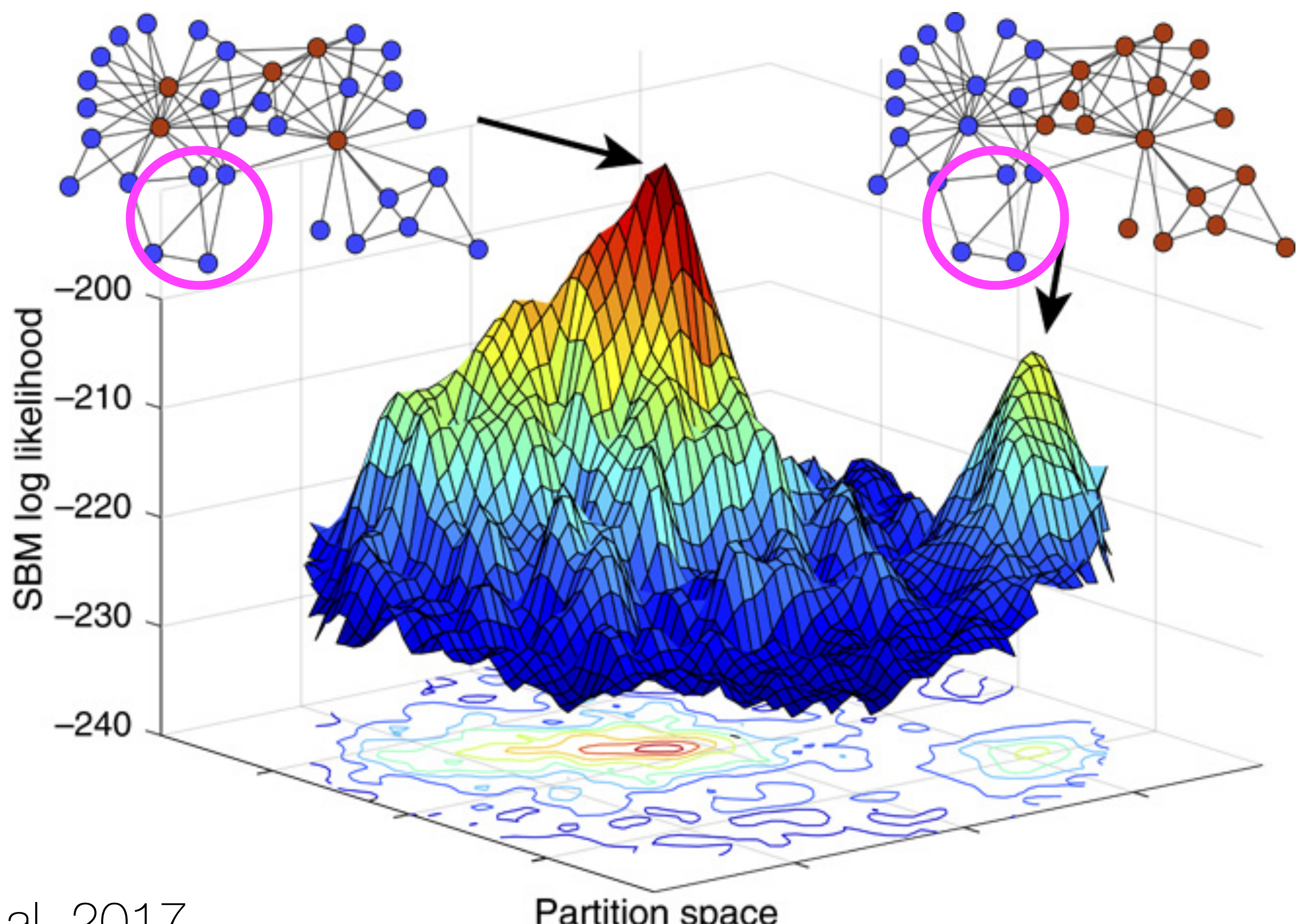
$$P(Observable|A) = \sum_{\Theta} P(Observable|A,\Theta)\boxed{P(\Theta|A)}$$



Peel et al. 2017

Statistical significance: *Maximization vs Sampling*

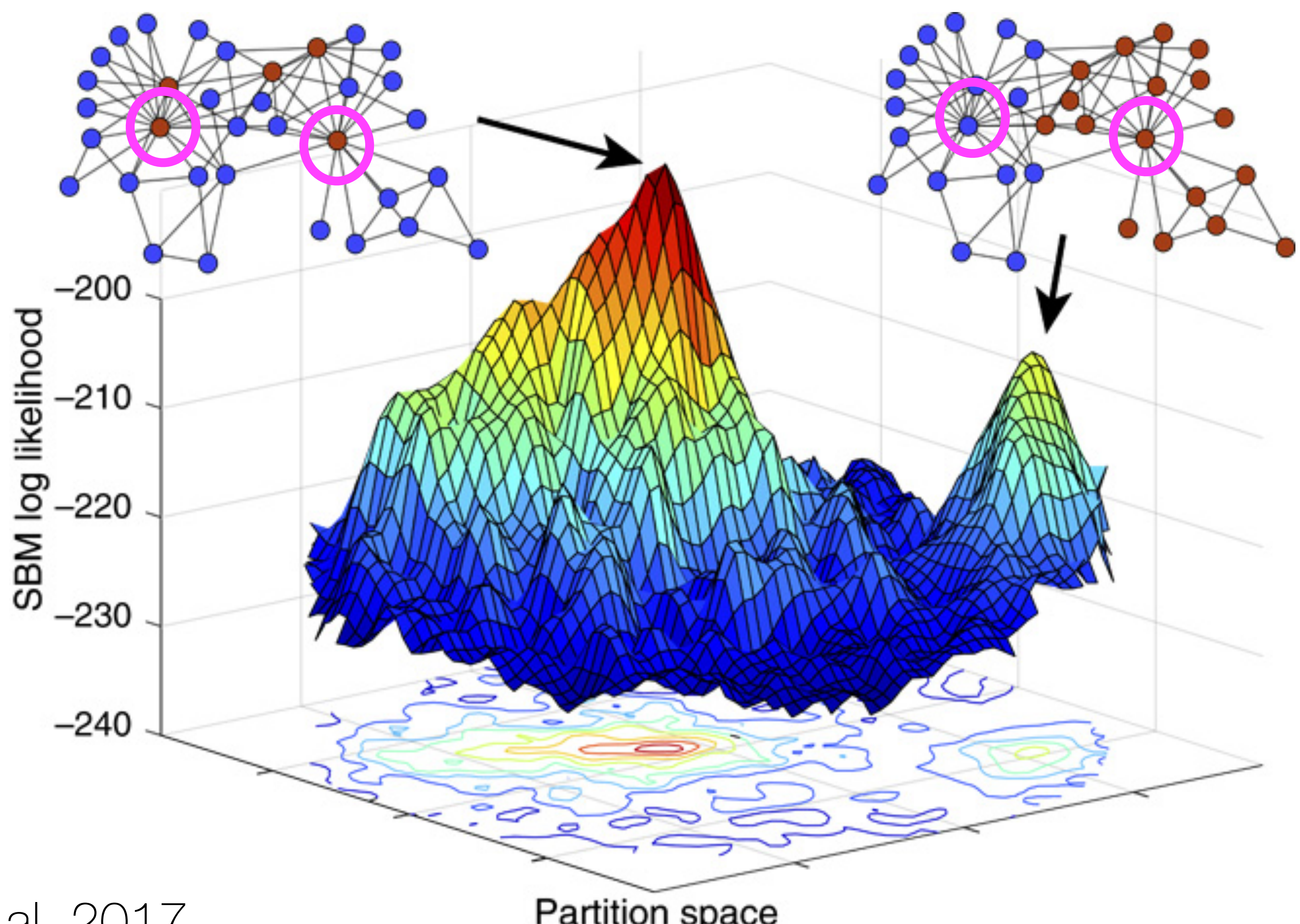
$$P(Observable|A) = \sum_{\Theta} P(Observable|A, \Theta)\boxed{P(\Theta|A)}$$



Peel et al. 2017

Statistical significance: *Maximization vs Sampling*

$$P(Observable|A) = \sum_{\Theta} P(Observable|A, \Theta)\boxed{P(\Theta|A)}$$



Peel et al. 2017

- Is a good practice to **question** the significance of the inferred community
- And we haven't even touched the 'detectability' issue …

Advanced topic: *metadata*

A lot of information that we haven't used!

Age, gender, education, religion, etc…

Advanced topic: *metadata*

A lot of information that we haven't used!

Age, gender, education, religion, etc…

Should we use it ?!??! How?

Advanced topic: *metadata*

A lot of information that we haven't used!

Age, gender, education, religion, etc…

Should we use it ?!??! How?

Not as ground truth!!!!!

Advanced topic: *metadata*

A possible answer: use them *a posteriori*

$$y_i = \sum_n \alpha_n \, x_i^n$$

Group membership of i

Coefficient to be estimated

Metadata of node i

Advanced topic: *metadata*

A possible answer: use them *a posteriori*

$$y_i = \sum_n \alpha_n \, x_i^n$$

Group membership of i

Coefficient to be estimated

Metadata of node i

What about incorporating it *a priori*?

Metadata: *questions* for an *a priori* incorporation

- Do metadata **explain** the network structure?

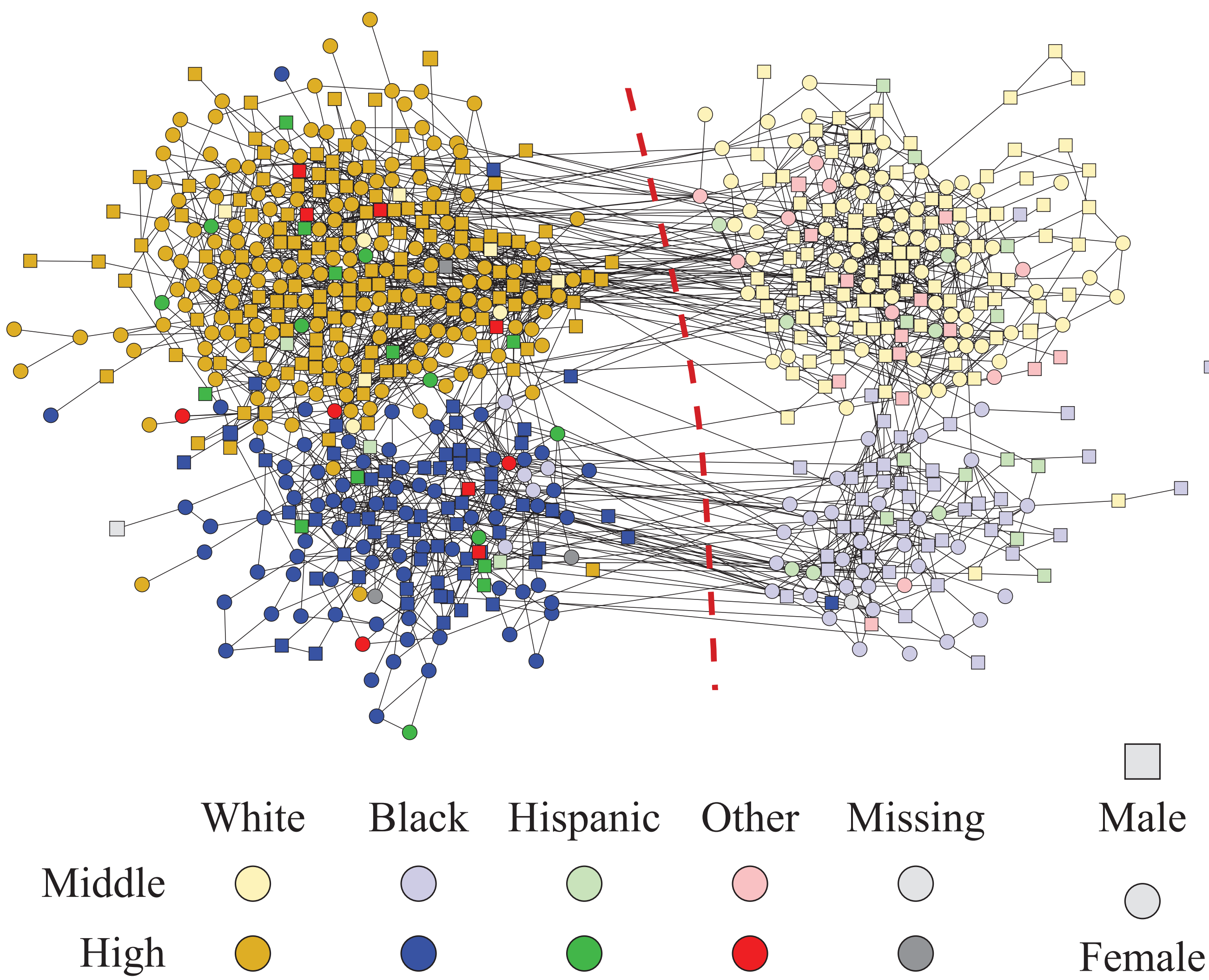
"BESTest,neoSBM", Peel et al. 2017

- Can we **use** them to guide community detection?

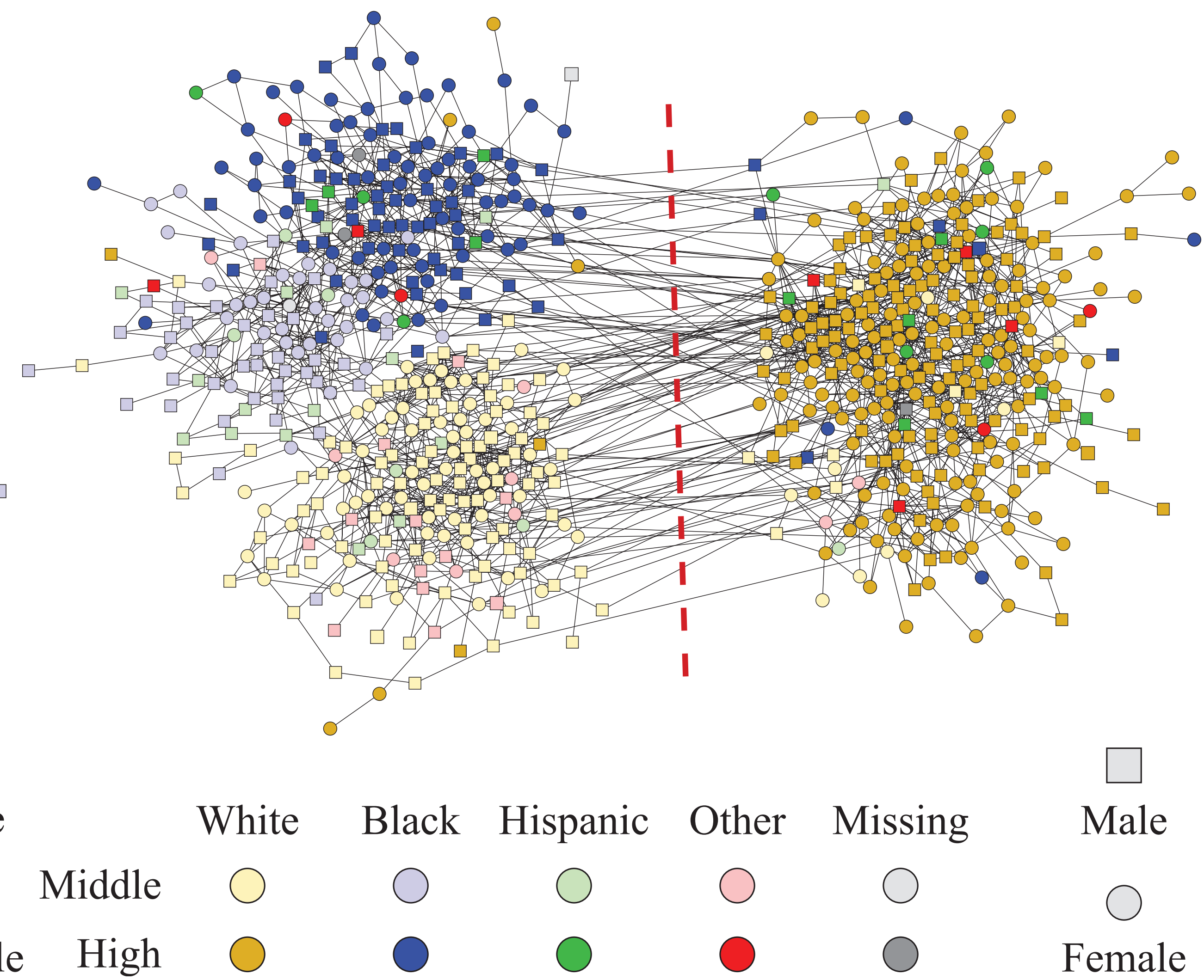
Newman and Clauset 2016
Yang et al. 2014
Hric et al 2016 …

Metadata: *questions* for an *a priori* incorporation

Use age: strong correlation with the partition



Use gender: no correlation with the partition



Newman & Clauset, *Nat. Comms.* 7, 11863 (2016)

Lecture 2: *implementing the model*

1. **Testing the model (20mins)**
   - Measuring correlation with ‘ground truth’: F1score, Mutual Information
   - No ground truth: AUC and MDL
2. **Analysing the results (20 mins)**
   - Normalize and extract max group
   - Visualize results
   - Statistical significance: many local optima
3. *Advanced topics (10mins): Metadata*

Positions opening coming soon

- Postdoc or PhD
- Max Planck Institute for Intelligent Systems, Tubingen, Germany
- Starting date flexible, from Jan 2019 onwards
- Contact me if you are interested in working on inference and optimization problems with statistical physics, probabilistic modeling and interdisciplinary applications
- Website with more details coming soon …

- Positions openings in other groups at https://cyber-valley.de/en

caterina.debacco@gmail.com https://github.com/cdebacco