RANDOM GRAPH MODELS

WHY USING RANDOM GRAPH MODELS

- Several good reasons:
  - Compare an observed network with a randomized version
    - Is observed property X "exceptional", or identical in any network with same property X?
  - Study some properties in a "controlled environment"
    - How does property X behaves when increasing property Y ?
  - Explain a given phenomenon
    - Such simple mechanism can reproduce property X and Y
  - Generate synthetic datasets
    - Testing an algorithm on 100 variations of the same network

NULL MODELS

- Using Random Graphs as Null models
  - Assume some properties (X1, X2, …) of your data are given
  - And that everything else is random
    - =>Is what you are observing for property Y unexpected/random/exceptional?
  - Principle of a reference point
    - You should always interpret a number in context, relate it to what you expect by chance

Fundamental network models

Central quantities in network analysis

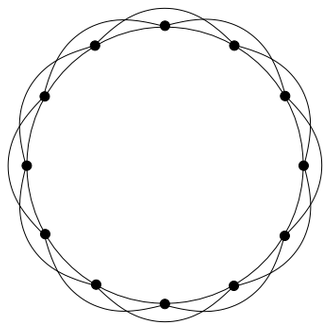
- Degree distribution: $P(k)$
- Clustering coefficient: C
- Average path length: $\langle d \rangle$

Network	Degree distribution	Path length	Clustering coefficient
Real world networks	broad	short	large

Regular lattices

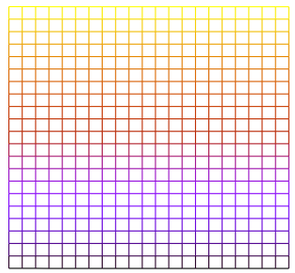
- Graphs where each node has the **same degree** k
- Translational symmetry in n directions

1D

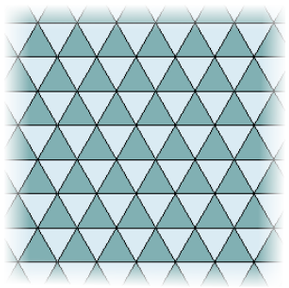


k=4

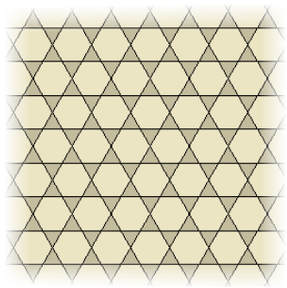
2D lattices



k=4

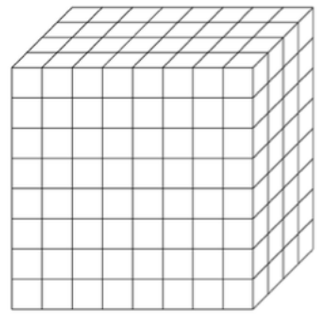


k=6

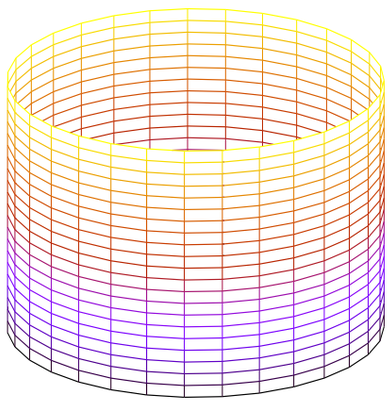


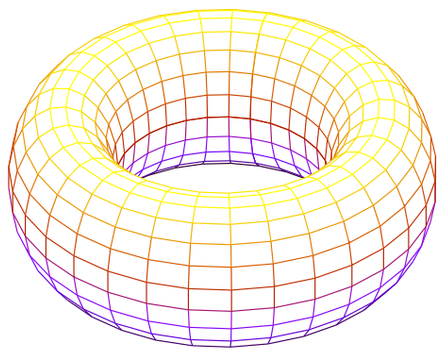
k=4

3D lattices



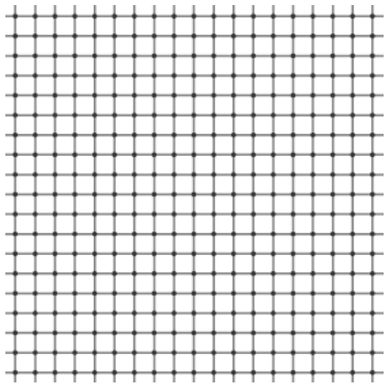
k=6



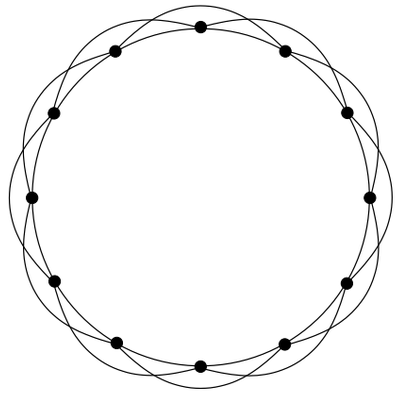


Regular lattices

Clustering coefficient



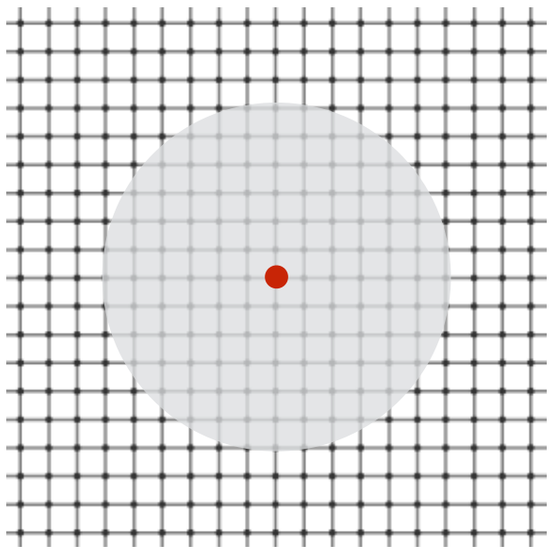
C=0



C=3/6

- Clustering coefficient depends on the structure (can be large or not)
- It is constant for each node

Path length



- Average path length grows quickly with n when k << n
- In a *large* graph with *realistic* average degrees, will be large

Regular lattices

Network	Degree distribution	Path length	Clustering coefficient
Real world networks	broad	short	large
Regular lattices	constant	long	can be large

The Erdős-Rényi Random Graph model (ER)

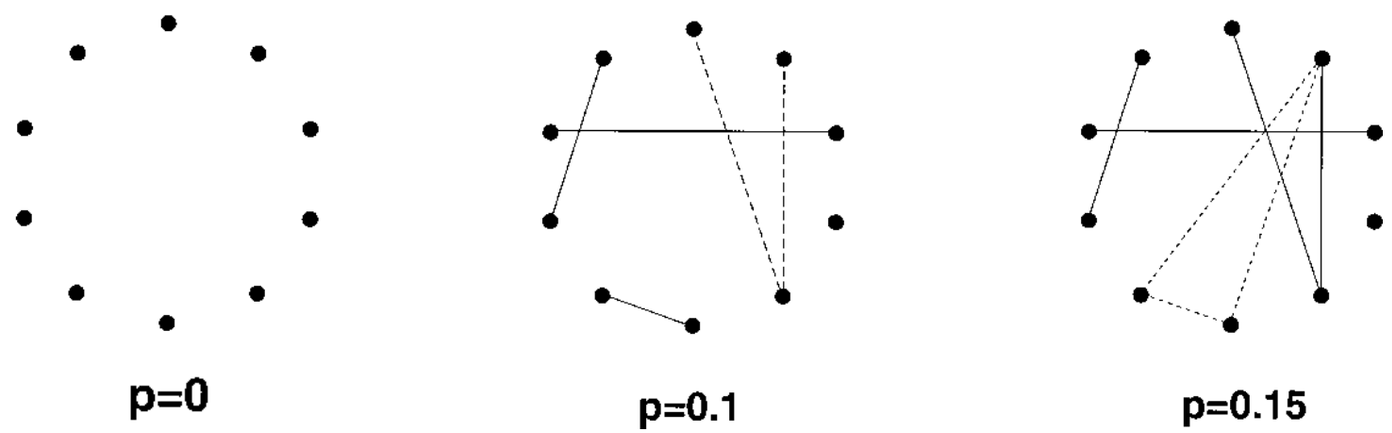
ER Random Graphs

Erdős-Rényi model: simple way to generate random graphs

- The $G(n,L)$ definition
  1. Take n disconnected nodes
  2. Add L edges uniformly at random

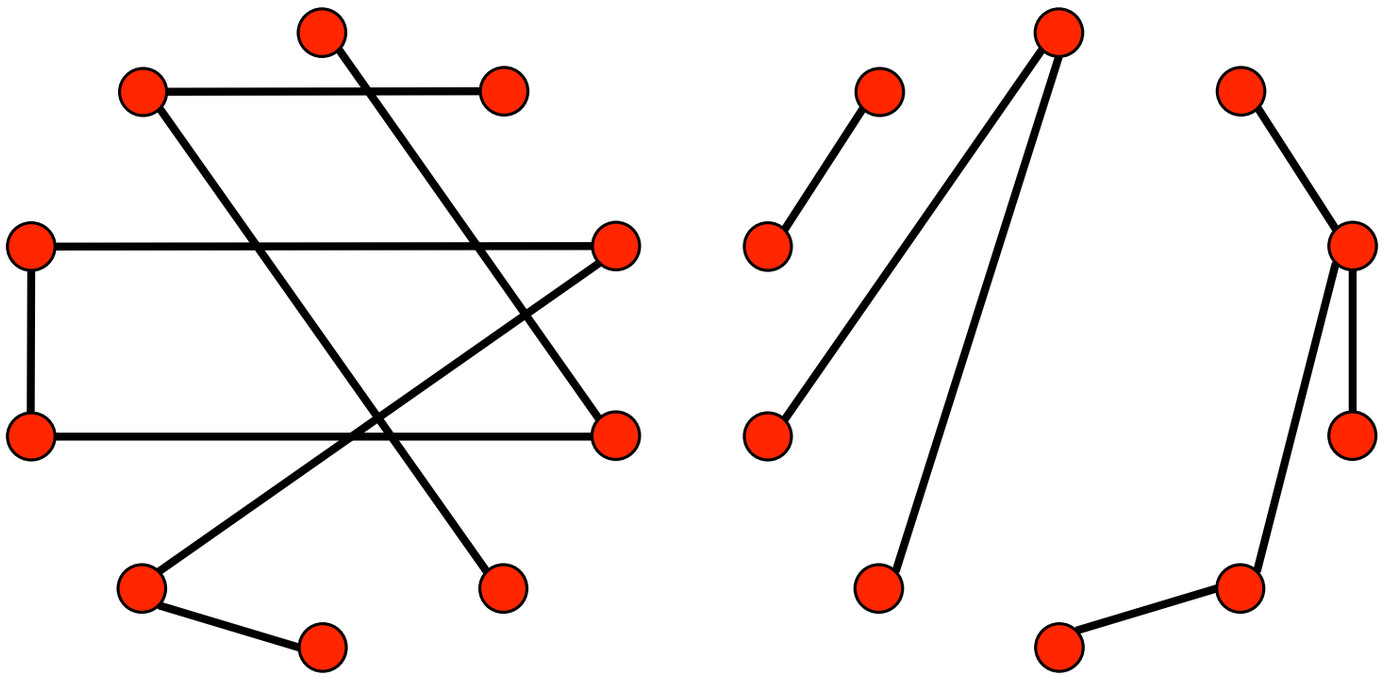
  Alternatively:
  - pick uniformly randomly a graph from the set of all graphs with n nodes and L links

- The $G(n,p)$ definition
  1. Take n disconnected nodes
  2. Add an edge between any of the nodes independently with probability p



Random Graphs

In the G(n,p) variant, the number of edges may vary

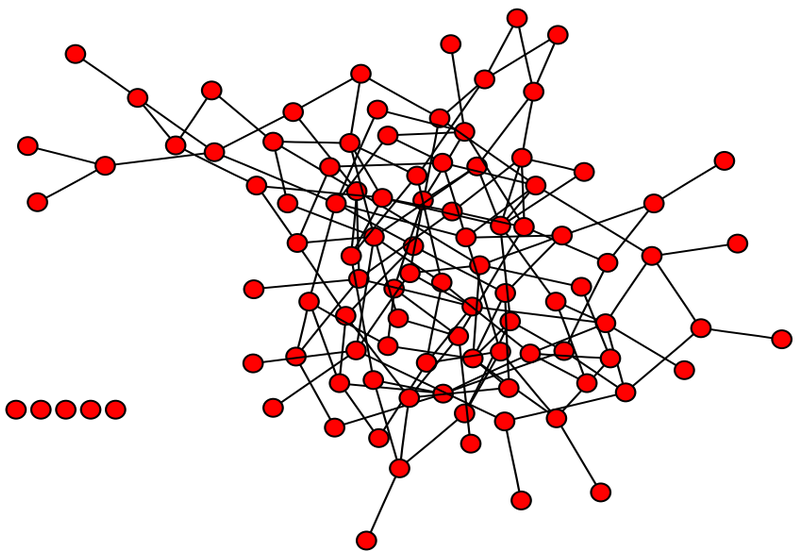


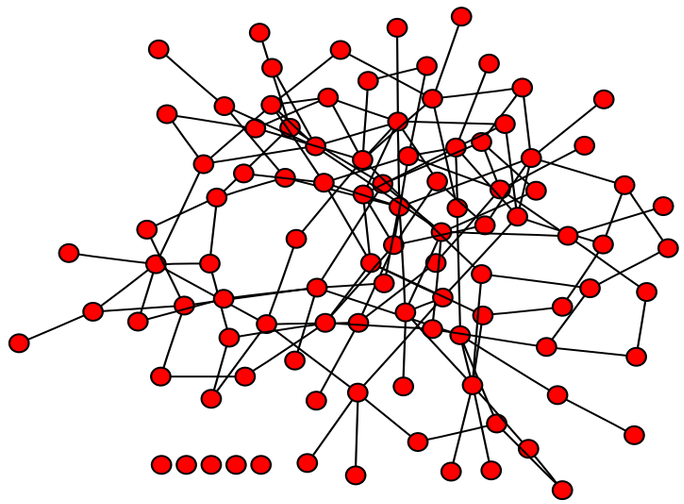
n=10

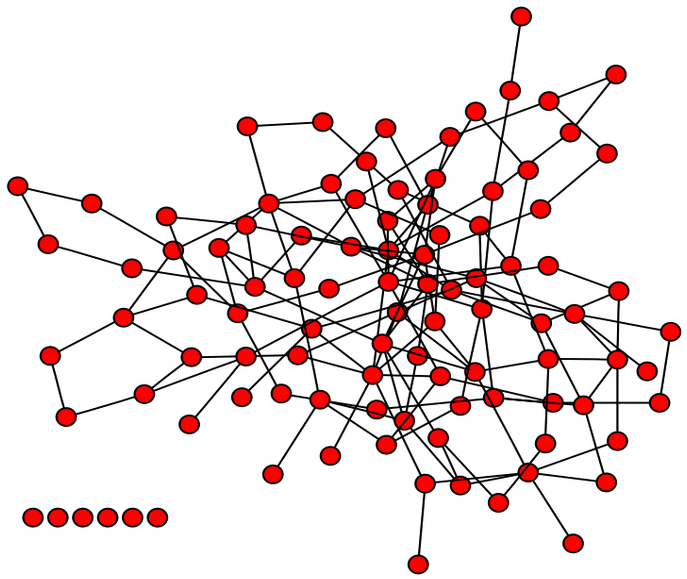
p=1/6

ER Random Graphs

p=0.03
N=100





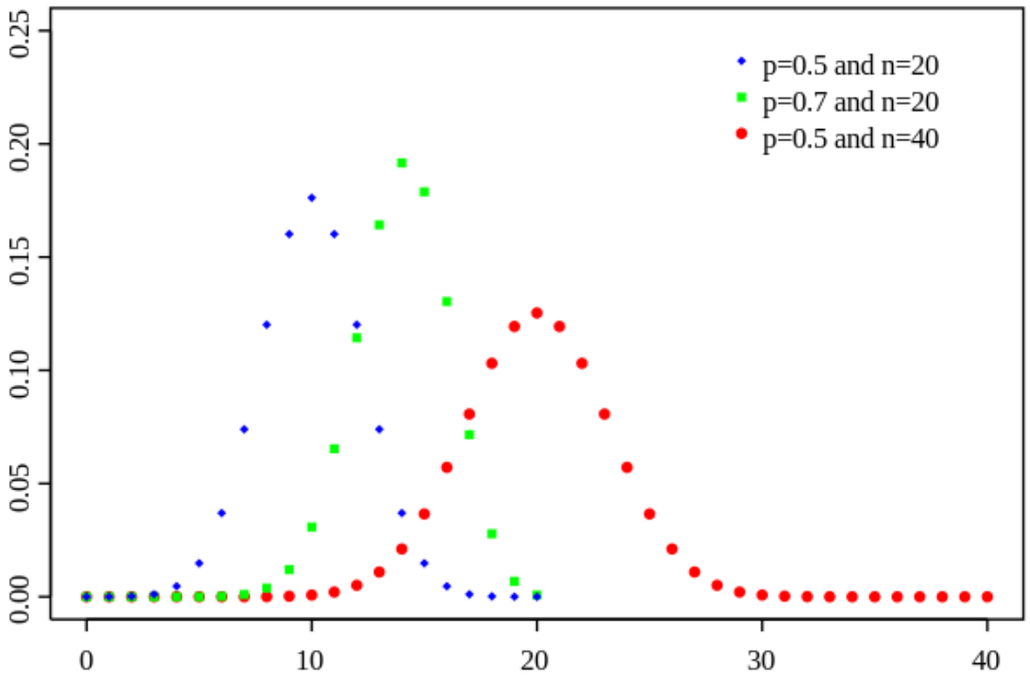


DESCRIBING ER RANDOM GRAPHS

Reminder

Binomial distribution:

Discrete probability distribution of the number of successes(**x**) in a sequence of **N** independent experiments, with success probability **p**



(PMF)

$$P(x) = \binom{N}{x} p^x (1-p)^{N-x}$$

Binomial coefficient:

Number of ways, disregarding order, that **k** objects can be chosen from among **n** objects

$$\binom{n}{k} = \frac{n!}{k!(n-k)!}$$

Reminder

Binomial distribution:

Discrete probability distribution of the number of successes(**x**) in a sequence of **N** independent experiments, with success probability **p**

Properties of Binomial distribution

PMF $$P(x) = \binom{N}{x} p^x (1-p)^{N-x}$$

Mean $$< x > = pN$$

variance $$\sigma^2 = Np(1-p)$$

Distribution of number of edges

We want to compute:

P(L): probability to have exactly **L** links in a network of **n** nodes and probability **p**

Binomial distribution

$$P(x) = \binom{N}{x} p^x (1-p)^{N-x}$$

N: Number of experiments

Pairs of nodes

$$N = \binom{n}{2} = \frac{n(n-1)}{2}$$

P(L): probability to have exactly **L** links in a network of **n** nodes (with **p** the probability to have an edge)

$$P(L) = \binom{\binom{n}{2}}{L} p^L (1-p)^{\binom{n}{2}-L}$$

Distribution of number of edges

- *G(n,p)*

Expected number of links <L>

$$< L > = pN = p\frac{n(n-1)}{2}$$

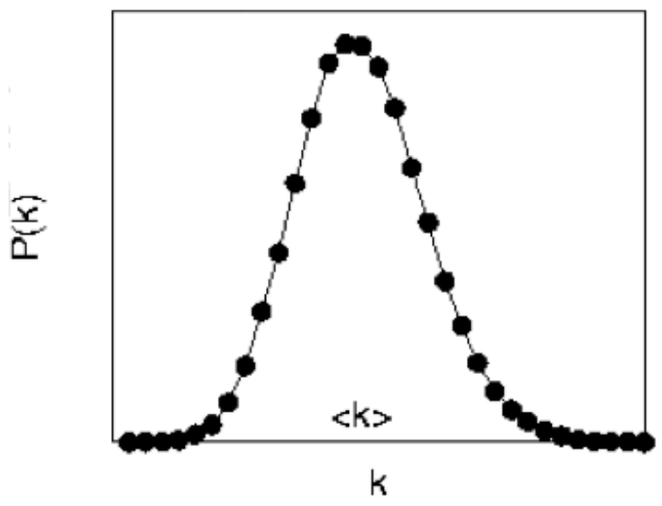
Variance

$$\sigma^2 = Np(1-p) = \frac{n(n-1)}{2}p(1-p)$$

Slide from CCNR course, A. L. Barabási (2012)

Degree distribution - Random Graphs

- *G(n,p)*



P(k): probability to have exactly **k** links among **n-1** (total # of other nodes), with **p** the probability to have an edge

$$P(k) = \binom{n-1}{k} p^k (1-p)^{(n-1)-k}$$

For each node,
independent probabilities to take each neighbor
=> Binomial distribution of degrees

Characteristics:

$$< k > = p(n-1)$$

$$\sigma_k^2 = p(n-1)(1-p)$$

Slide from CCNR course, A. L. Barabási (2012)

Degree distribution - Random Graphs

For large **n** and small **k** (p,L), we can approximate the degree distribution using a poisson distribution of parameter (mean) $\lambda = < k >$

Poisson distribution

$$P(K) = \frac{\lambda^K e^{-\lambda}}{K!}$$

Distribution of degrees

$$P(k) = \frac{< k >^k \, e^{-<k>}}{k!}$$

standard deviation

$$\sigma = \sqrt{< k >}$$

Degree distribution - Random Graphs

standard deviation $\sigma = \sqrt{< k >}$

High confidence to have degrees close to average degrees when degrees increase

Degree distribution - Random Graphs

Conclusion: degree distribution is (approximately) Poisson distributed,

it is **not**

-Heterogeneous

-Long tail

-Scale-free

Slide from CCNR course, A. L. Barabási (2012)

Clustering - Random Graphs

Local clustering of a node

Reminder, clustering coefficient

$$C_i \equiv \frac{2n_i}{k_i(k_i-1)}$$ where n_i is the number of links between the neighbours of node i

- Edges are independent and have the same probability p

$$n_i \cong p\,\frac{k_i(k_i-1)}{2}$$

possible links btw neighbors

$$p = \frac{<k>}{n-1}$$

$$C_i = \frac{2<k>}{n-1}\;\frac{k_i(k_i-1)}{2}\;\frac{1}{k_i(k_i-1)} = \frac{<k>}{n-1} = p$$

- For fixed average degree C is decreasing as N goes large

  ➡ Low clustering coefficient

  ➡ It is vanishing with the system size

Clustering - ER Random Networks

- **Small clustering coefficient** $C_i \equiv \frac{1}{N} < k >= p$

Real-world networks

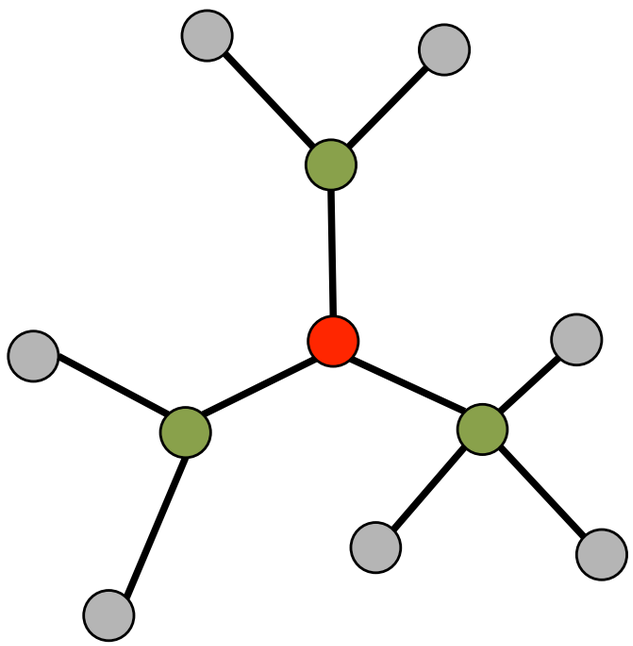
Network	Size	$\langle k \rangle$	ℓ	ℓ_{rand}	C	C_{rand}	Reference
WWW, site level, undir.	153 127	35.21	3.1	3.35	0.1078	0.00023	Adamic, 1999
Internet, domain level	3015–6209	3.52–4.11	3.7–3.76	6.36–6.18	0.18–0.3	0.001	Yook *et al.*, 2001a, Pastor-Satorras *et al.*, 2001
Movie actors	225 226	61	3.65	2.99	0.79	0.00027	Watts and Strogatz, 1998
LANL co-authorship	52 909	9.7	5.9	4.79	0.43	1.8×10^{-4}	Newman, 2001a, 2001b, 2001c
MEDLINE co-authorship	1 520 251	18.1	4.6	4.91	0.066	1.1×10^{-5}	Newman, 2001a, 2001b, 2001c
SPIRES co-authorship	56 627	173	4.0	2.12	0.726	0.003	Newman, 2001a, 2001b, 2001c
NCSTRL co-authorship	11 994	3.59	9.7	7.34	0.496	3×10^{-4}	Newman, 2001a, 2001b, 2001c
Math. co-authorship	70 975	3.9	9.5	8.2	0.59	5.4×10^{-5}	Barabási *et al.*, 2001
Neurosci. co-authorship	209 293	11.5	6	5.01	0.76	5.5×10^{-5}	Barabási *et al.*, 2001
E. coli, substrate graph	282	7.35	2.9	3.04	0.32	0.026	Wagner and Fell, 2000
E. coli, reaction graph	315	28.3	2.62	1.98	0.59	0.09	Wagner and Fell, 2000
Ythan estuary food web	134	8.7	2.43	2.26	0.22	0.06	Montoya and Solé, 2000
Silwood Park food web	154	4.75	3.40	3.23	0.15	0.03	Montoya and Solé, 2000
Words, co-occurrence	460.902	70.13	2.67	3.03	0.437	0.0001	Ferrer i Cancho and Solé, 2001
Words, synonyms	22 311	13.48	4.5	3.84	0.7	0.0006	Yook *et al.*, 2001b
Power grid	4941	2.67	18.7	12.4	0.08	0.005	Watts and Strogatz, 1998
C. Elegans	282	14	2.65	2.25	0.28	0.05	Watts and Strogatz, 1998

Albert, R. et.al. Rev. Mod. Phy. (2002)

Distance - ER Random Graphs - Intuition

low clustering coefficient=>

Random graphs tend to have a tree-like topology with almost constant node degrees.



- nr. of first neighbors: $N(u)_1 = < k >$
- nr. of second neighbors: $N(u)_2 = < k >^2$
- nr. of neighbours at distance d: $N(u)_d = < k >^d$

Intuition: At which distance are all nodes reached?

$$n = < k >^d \Rightarrow \log_{<k>} n = d \Rightarrow d = \frac{\log n}{\log < k >}$$

Diameter, avg. distance in $\mathcal{O}(\log n)$

Slide from CCNR course, A. L. Barabási (2012)

Distance - ER Random Graphs

- **Logarithmically short distance**

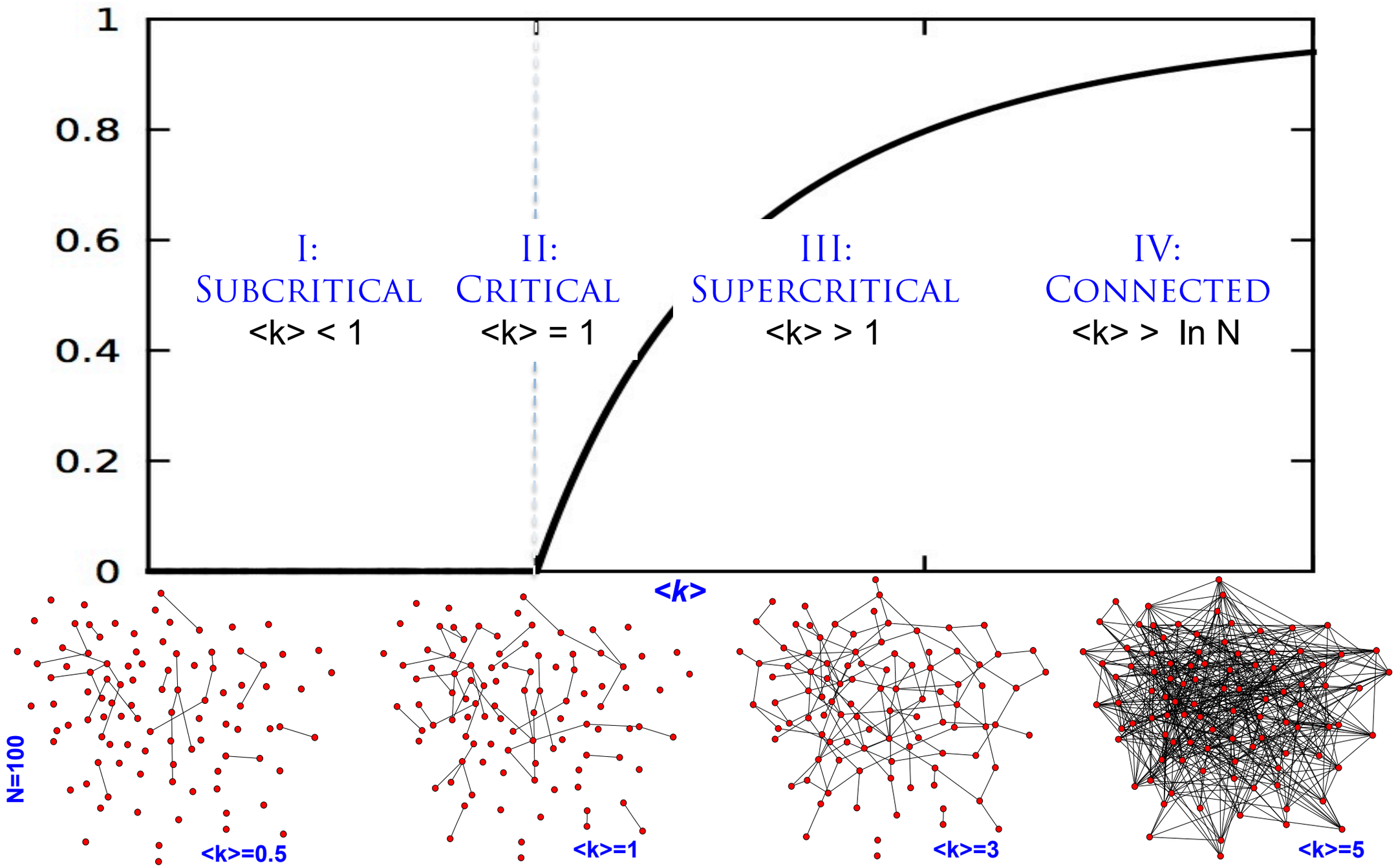
$$d = \frac{\log n}{log\langle k\rangle}$$

Real-world networks

Network	Size	$\langle k\rangle$	ℓ	ℓ_{rand}	C	C_{rand}	Reference
WWW, site level, undir.	153 127	35.21	3.1	3.35	0.1078	0.00023	Adamic, 1999
Internet, domain level	3015–6209	3.52–4.11	3.7–3.76	6.36–6.18	0.18–0.3	0.001	Yook *et al.*, 2001a, Pastor-Satorras *et al.*, 2001
Movie actors	225 226	61	3.65	2.99	0.79	0.00027	Watts and Strogatz, 1998
LANL co-authorship	52 909	9.7	5.9	4.79	0.43	1.8×10^{-4}	Newman, 2001a, 2001b, 2001c
MEDLINE co-authorship	1 520 251	18.1	4.6	4.91	0.066	1.1×10^{-5}	Newman, 2001a, 2001b, 2001c
SPIRES co-authorship	56 627	173	4.0	2.12	0.726	0.003	Newman, 2001a, 2001b, 2001c
NCSTRL co-authorship	11 994	3.59	9.7	7.34	0.496	3×10^{-4}	Newman, 2001a, 2001b, 2001c
Math. co-authorship	70 975	3.9	9.5	8.2	0.59	5.4×10^{-5}	Barabási *et al.*, 2001
Neurosci. co-authorship	209 293	11.5	6	5.01	0.76	5.5×10^{-5}	Barabási *et al.*, 2001
E. coli, substrate graph	282	7.35	2.9	3.04	0.32	0.026	Wagner and Fell, 2000
E. coli, reaction graph	315	28.3	2.62	1.98	0.59	0.09	Wagner and Fell, 2000
Ythan estuary food web	134	8.7	2.43	2.26	0.22	0.06	Montoya and Solé, 2000
Silwood Park food web	154	4.75	3.40	3.23	0.15	0.03	Montoya and Solé, 2000
Words, co-occurrence	460.902	70.13	2.67	3.03	0.437	0.0001	Ferrer i Cancho and Solé, 2001
Words, synonyms	22 311	13.48	4.5	3.84	0.7	0.0006	Yook *et al.*, 2001b
Power grid	4941	2.67	18.7	12.4	0.08	0.005	Watts and Strogatz, 1998
C. Elegans	282	14	2.65	2.25	0.28	0.05	Watts and Strogatz, 1998

Albert, R. et.al. Rev. Mod. Phy. (2002)

Phase transition of giant component



Structural (percolation) phase transition at <k>=1 (or equivalently when p=1/N)

Slide from CCNR course, A. L. Barabási (2012)

ER Random Network - catch up

Network	Degree distribution	Path length	Clustering coefficient
Real world networks	broad	short	large
Regular lattices	constant	long	large
ER random networks	Poissonian	short	small

It is not capturing the properties of any real system

BUT

it serves as a reference system for any other network model

Configuration model

More details at [http://tuvalu.santafe.edu/~aaronc/courses/5352/fall2013/csci5352_2013_L11.pdf]

Random graphs with specified degrees

Problem

- The ER Random Graph model has a Poisson degree distribution
- Most real-world networks have heavy-tailed degree distributions
- **We need to generate networks having pre-determined degrees or degree distribution, but maximally random otherwise**
- The observed properties (clustering coefficient, etc.) might be due *only* to the difference in degree distribution

Random graphs with specified degrees

Configuration model

Based on an observed network

- Defined as $G(n, \vec{k})$ where $\vec{k} = \{k_i\}$ is a degree sequence on n nodes, with k_i being the degree of node i

Ad hoc degree distribution

- **The degree sequence $\vec{k} = \{k_i\}$ can be sampled from a probability distribution**
  - Delta/Dirac function => Random regular graph
  - Poisson => Similar to ER for proper parameters
  - Scale-free => Power-law random graph
- Only global condition to satisfy is: $\sum_i k_i$ **mod** $2 = 0$

  (even degree sum) i.e. each edge has to have ending nodes

Random graphs with specified degrees

Configuration model

How much of some observed pattern is driven by the degrees alone?

Exact or approximate degree distribution

- The model can preserve the **expected** degree sequence, or the **exact** degree sequence
  - Chung-lu (appoximate)
  - Molloy-reed (Exact)

Random graphs with specified degrees

Chung-Lu model for configuration networks = Approximate degree distribution

- Probabilistic model which produce a network with degrees approximating (on average) the original degree
- It is a *"coin-flipping"* process as ER model but **the probability that two nodes i and j are connected depends on the degree k_i and k_j of the ending nodes**
- From the point of view of node i with degree k_i, the probability that one of its edges will connect to j with k_j:

$$k_j/2m$$

- This can happen via k_i links, thus the probability that they are connected:

$$p_{ij} = \frac{k_i k_j}{2m}$$

assuming that: $[\max(k_i)]^2 < 2m$

(/!\ inconsistent probability, it is rather expected number of edges)

Random graphs with specified degrees

Chung-Lu model for configuration networks = Approximate degree distribution

$$\forall_{i>j} \qquad A_{ij} = A_{ji} = \begin{cases} 1 & \text{with probability } p_{ij} \\ 0 & \text{otherwise} \end{cases} \qquad \text{where} \qquad p_{ij} = \frac{k_i k_j}{2m}$$

- Each pairs of nodes are considered once, thus it produces a **simple graph** (without self-loops and multi edges)
- Degree of a node equals only in “expectation” to the originally assigned degree

Complexity:

- $O(n^2)$: We need $n(n-1)$ flips to test all node pairs

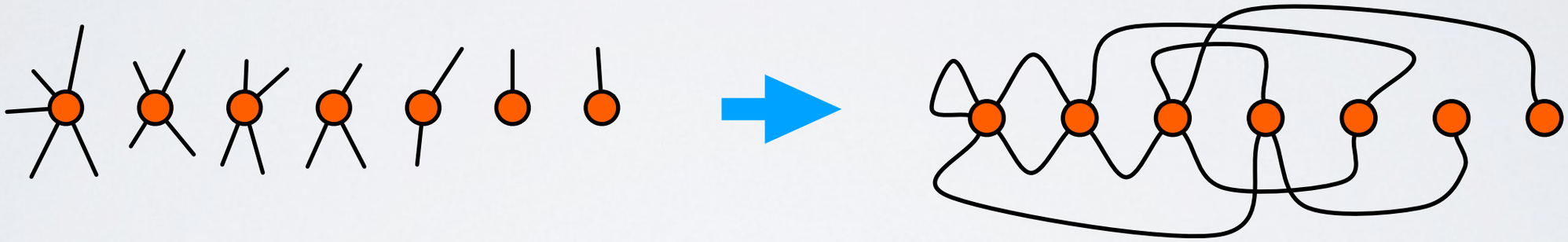
EXPENSIVE!

Random graphs with specified degrees

Molloy-Reed model for configuration networks = exact degree preservation

Original idea:

1. Given a degree sequence $\vec{k} = \{k_1, k_2, \ldots, k_n\}$
2. Assign each node $i \in V$ with k_i number of stubs
3. Select random pairs of unmatched stubs and connect them
4. Repeat 3 while there are unmatched stubs



- This process will produce a configuration model with exact degree sequence
- Possible to select multiple times stubs of the same pair of nodes ➜ **Multilinks**
- Possible to select the stubs of the same node to connect ➜ **Self-links**

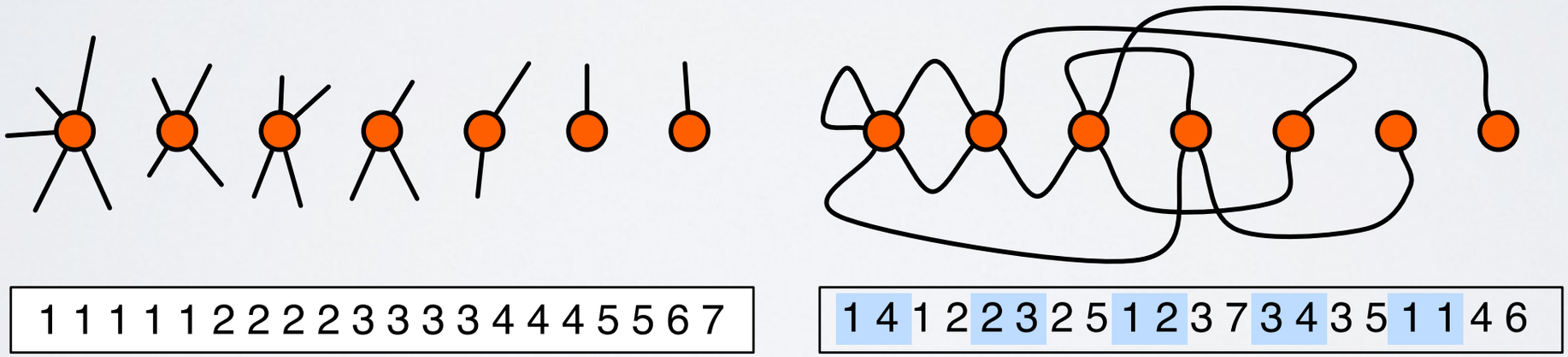
The obtained graph is not simple…but the density of multi and self-links ➜ 0 as N ➜ ∞

Random graphs with specified degrees

Molloy-Reed model for configuration networks = exact degree preservation

An effective algorithm:

1. Take an array $\vec{v}$ with length $2m$ and fill it with exactly k_i indices of each node $i \in V$
2. Make a random permutation of the array $\vec{v}$
3. Read the content of the array in an order and in pairs
4. Pairs of consecutive node indices will assign links in the configuration network



Complexity:

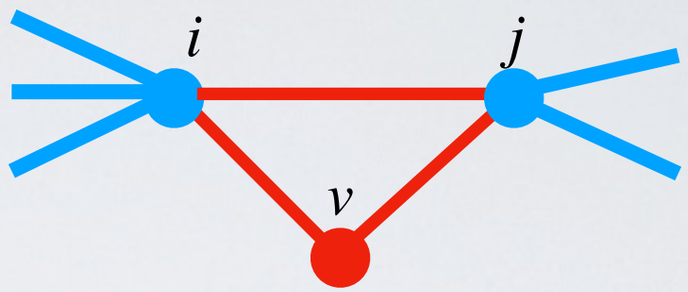
- *O(m)*: Random permutation of an array
- *O(m log m)*: assigning uniformly random variables to indices and quick-sort them

CHEAP!

More details at [http://tuvalu.santafe.edu/~aaronc/courses/5352/fall2013/csci5352_2013_L11.pdf]

Configuration model - mathematical properties

Expected clustering coefficient



It is the average probability that two neighbours of a vertex are connected

- Start at some vertex v (with degree $k \geq 2$)
- Choose a random pair of its neighbours i and j
- The probability that i and j are themselves connected is $k_i k_j / 2m$
- *But degree distributions of neighbors depend on their degree: more complex than simply counting frequencies of degrees (friendship paradox)*

Clustering coefficient

independent of network size

$$C = \ldots = \frac{1}{n} \frac{[\langle k^2 \rangle - \langle k \rangle]^2}{\langle k \rangle^3}$$

- It is a vanishing quantity $O(1/n)$ as long as the second moment is finite

For details, see: http://tuvalu.santafe.edu/~aaronc/courses/5352/fall2013/csci5352_2013_L12.pdf

ER Random Network - catch up

Network	Degree distribution	Path length	Clustering coefficient
Real world networks	broad	short	large
Regular lattices	constant	long	large
ER random networks	Poissonian	short	small
Configuration Model	Custom, can be broad	short	small

Watts-Strogatz model of small-world networks

Small-world networks

- One of the founding papers of Network Science…

D.J. Watts and S. Strogatz,

”Collective dynamics of 'small-world' networks”, Nature 393, 440–442, 1998

Contradiction: Real-world networks have

High clustering coefficient

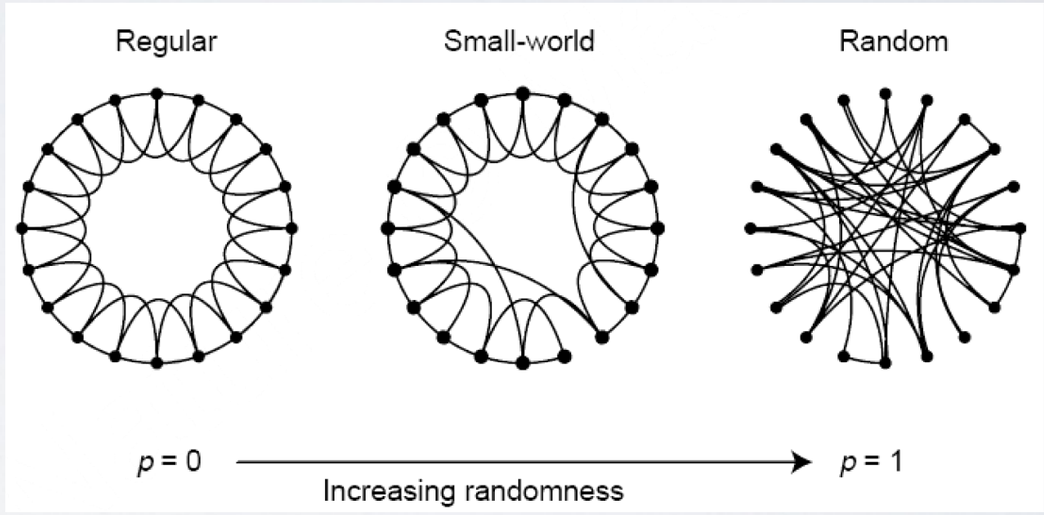
AND

Short distances

The Watts-Strogatz model

A model to capture large clustering coefficient and short distances observed in real networks

- It interpolates between an ordered finite lattice and a random graph
- Fixed parameters:
  - n - system size
  - K - initial coordination number
- Variable parameters:
  - p - rewiring probability



- Algorithm:

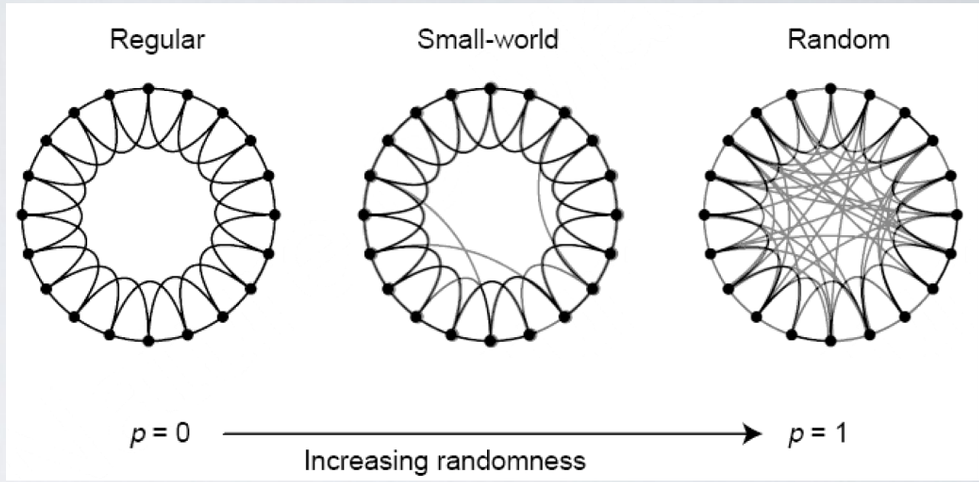
D.J. Watts and S. Strogatz, Nature (1998)

1. Start with a ring lattice with n nodes in which every node is connected to its first K neighbours ($K/2$ on either side).
2. Randomly rewire each edge of the lattice with probability p such that self-connections and duplicate edges are excluded.

By varying p the network can be transformed from a completely ordered ($p=0$) to a completely random ($p=1$) structure

The Watts-Strogatz model

(Global) Clustering coefficient (Definition 2)



- Independent of n
- if p→0 it recovers the ring value
- if p→1 , small

$$C = \frac{\frac{1}{4}NK(\frac{1}{2}K-1)\times 3}{\frac{1}{2}NK(K-1)+NK^2p+\frac{1}{2}NK^2p^2} = \frac{3(K-2)}{4(K-1)+8Kp+4Kp^2}$$

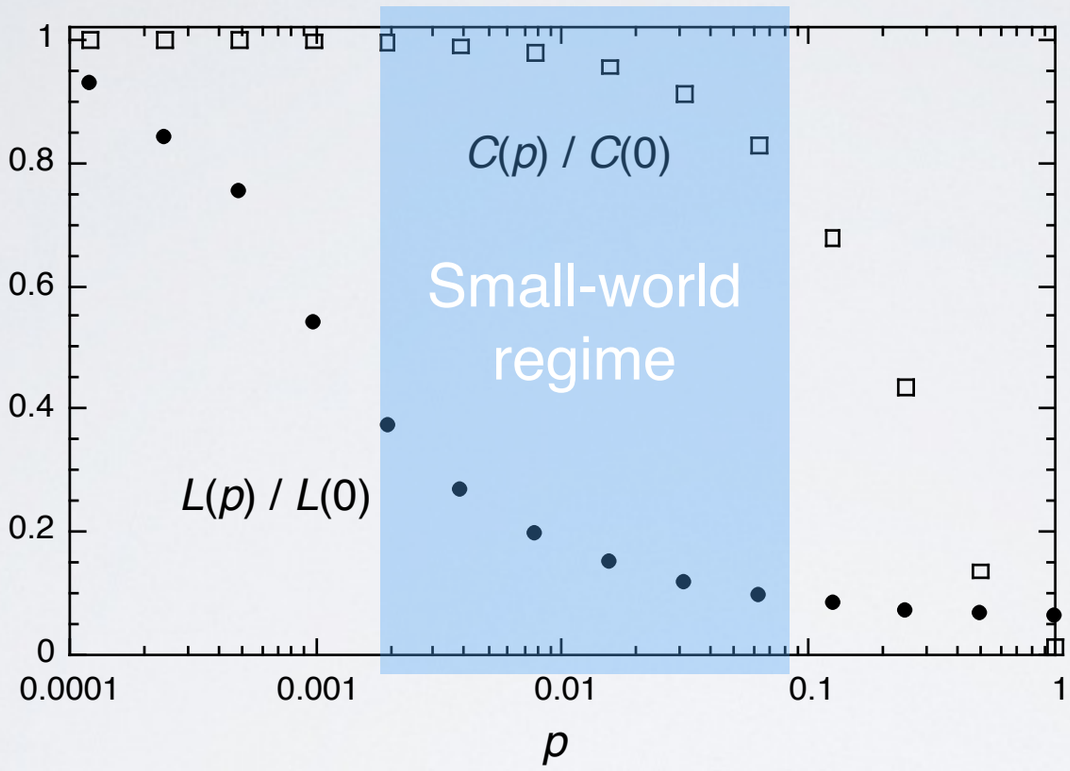
For details, see: http://tuvalu.santafe.edu/~aaronc/courses/5352/fall2013/csci5352_2013_L12.pdf

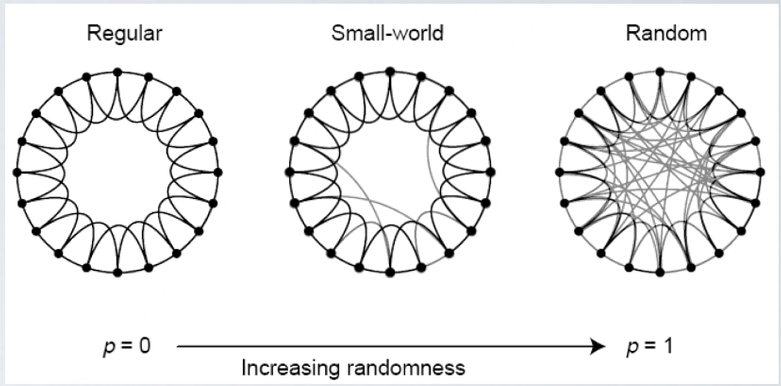
The Watts-Strogatz model

Average path length (Definition 2)

- No closed form solution

  - From numerical simulations:





- See Newman, M. E. (2000). Models of the small world. *Journal of Statistical Physics*, *101*(3-4), 819-841.

for details

L=avg path length

ER Random Network - catch up

Network	Degree distribution	Path length	Clustering coefficient
Real world networks	broad	short	large
Regular lattices	constant	long	large
ER random networks	Poissonian	short	small
Configuration Model	Custom, can be broad	short	small
Watts & Strogatz (in SW regime)	Poissonian	short	large

Scale-free networks

Scale-free networks

A network is called ***Scale-free*** **when its degree distribution follows (to some extent) a Power-law distribution**

Power-law distribution:

(PDF)

$$P(k) \propto k^{-\alpha}$$

α (sometimes γ) called the **exponent** of the distribution

When k (the size of something) increases by a factor c, $P(k)$ (how often it occurs) decreases by that same factor raised to a power α.

Example:

If $\alpha = 2$:

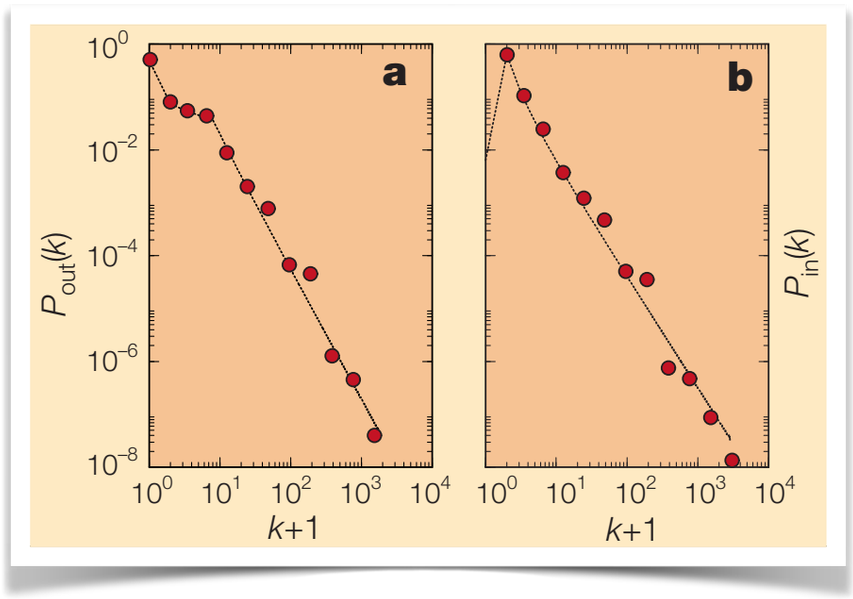
Doubling the size($c = 2$) makes the event $2^2 = 4$ times less frequent

Increasing the size 10x makes it $10^2 = 100$ less frequent

Scale-free networks - first observations

R. Albert, H. Jeong, A-L Barabási, Nature (1999)

Diameter of the world wide web

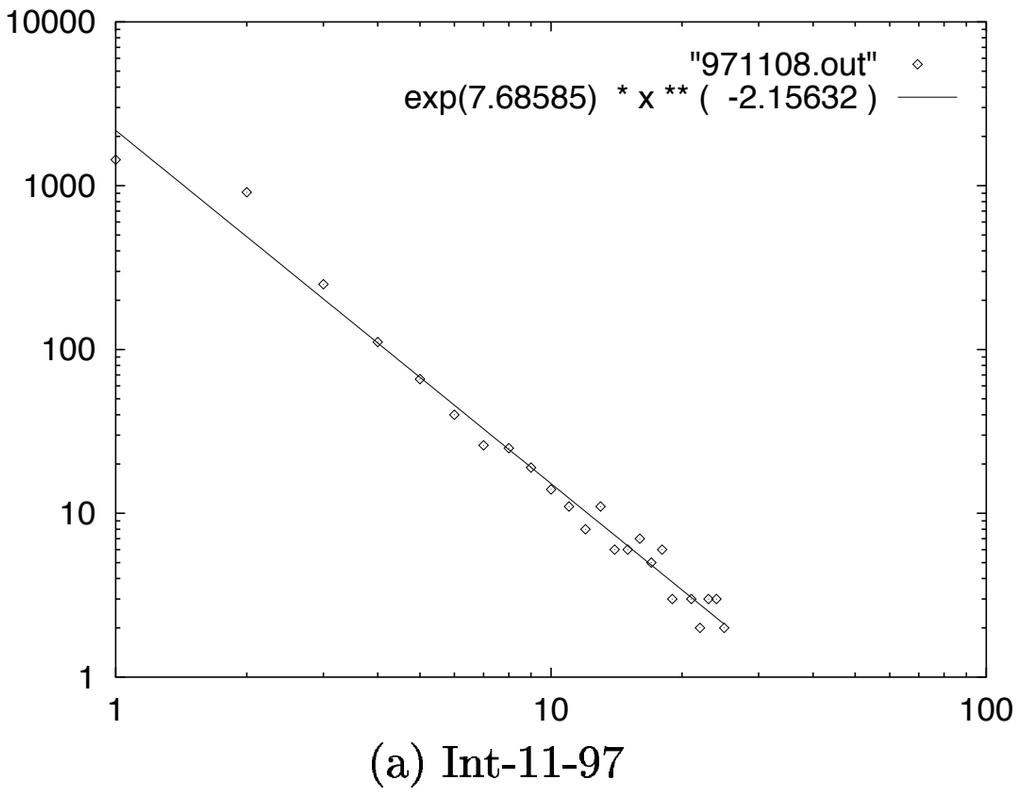


Power law
Appear as a line
On a log-log plot

Scale-free networks - other examples

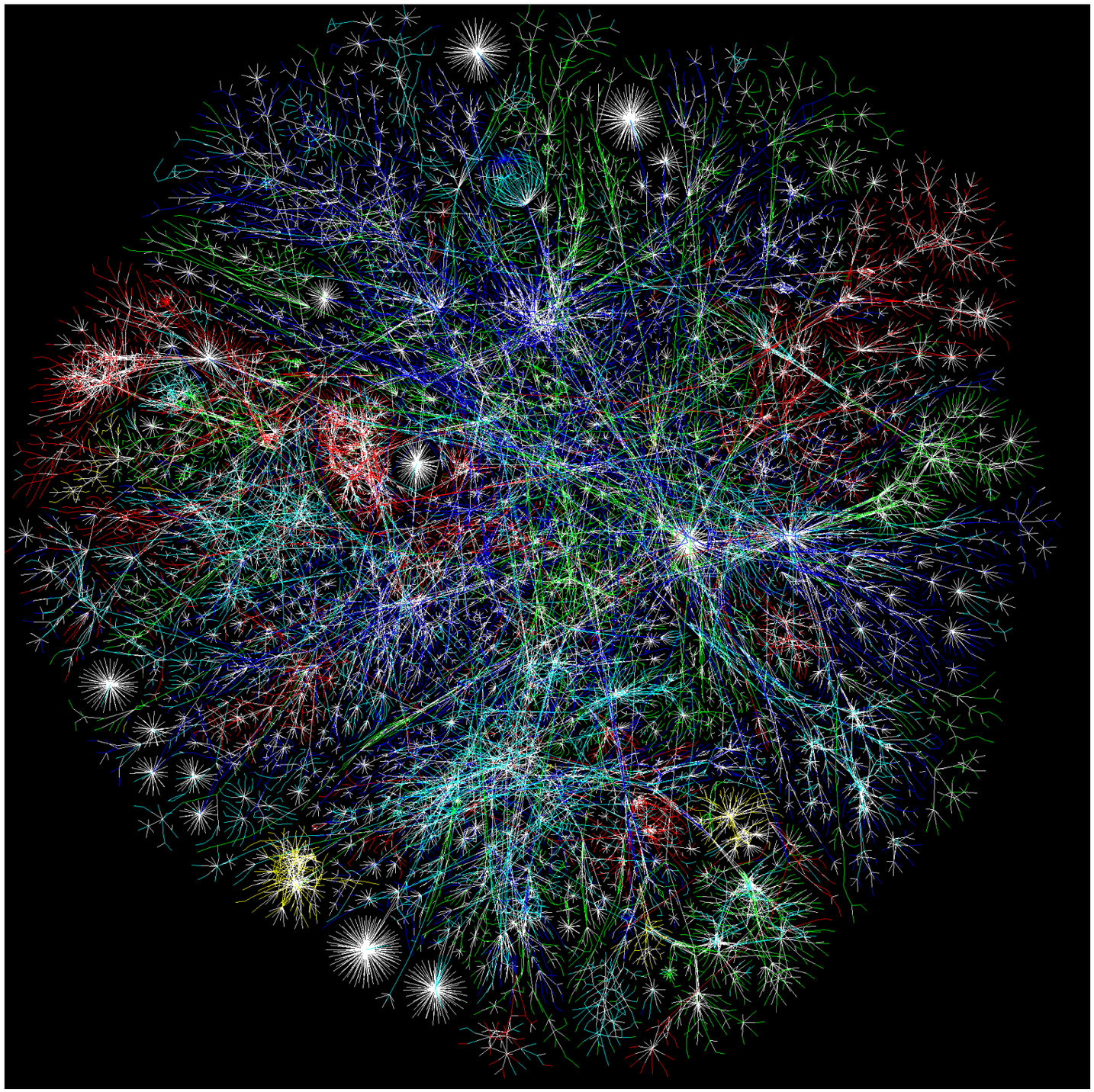
The internet

- Nodes: routers
- Links: Physical wires



(a) Int-11-97

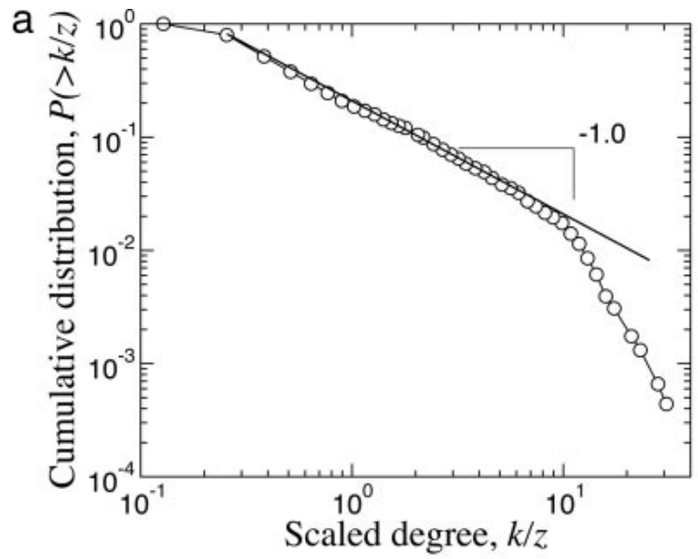
Faloutsos, Faloutsos and Faloutsos (1999)



Scale-free networks - other examples

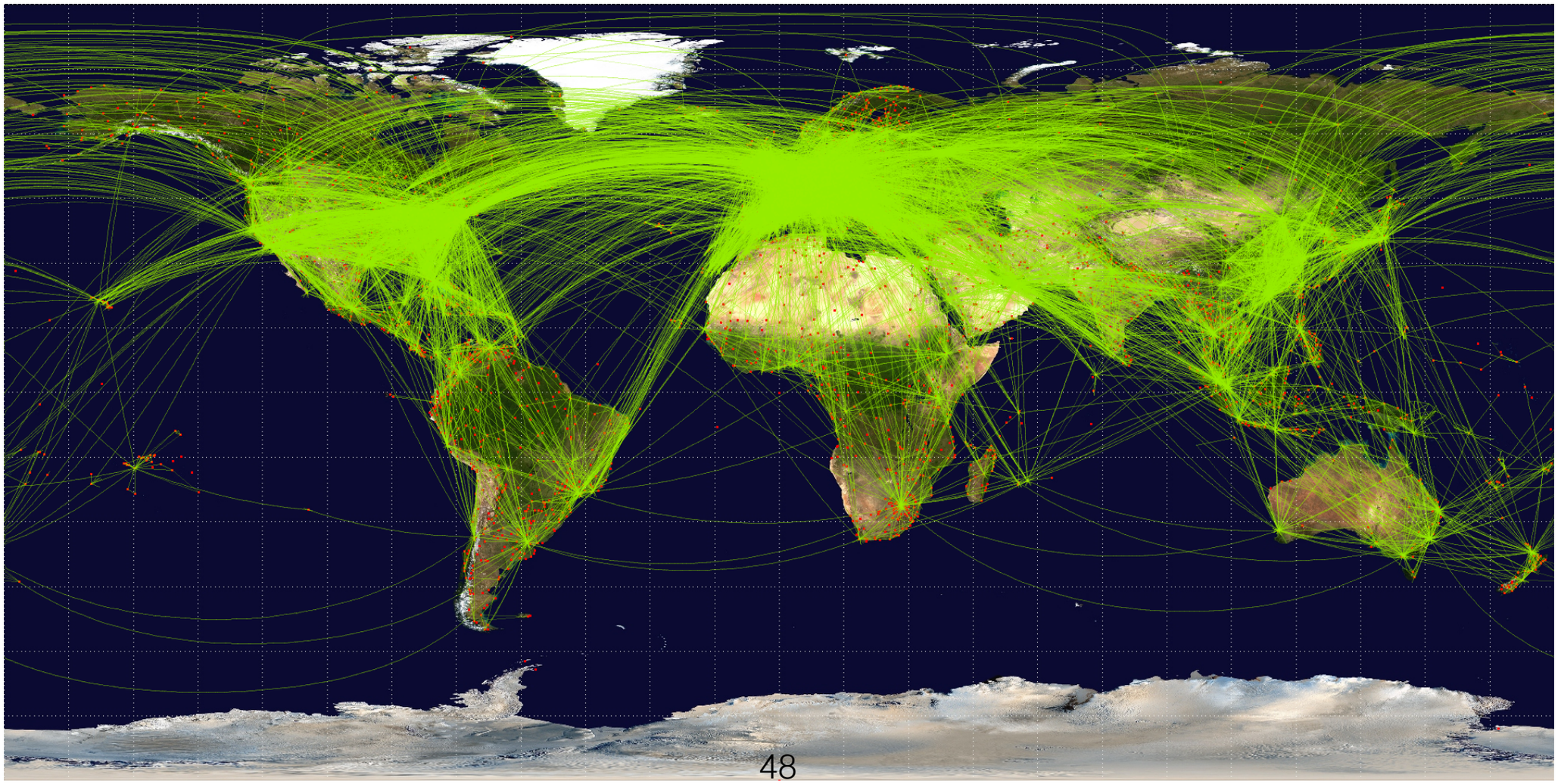
Airline route map network

- Nodes: airports
- Links: airplane connections



Guimera et.al. (2004)

Note: the cumulative distribution of a power law is also a line on a log-log plot

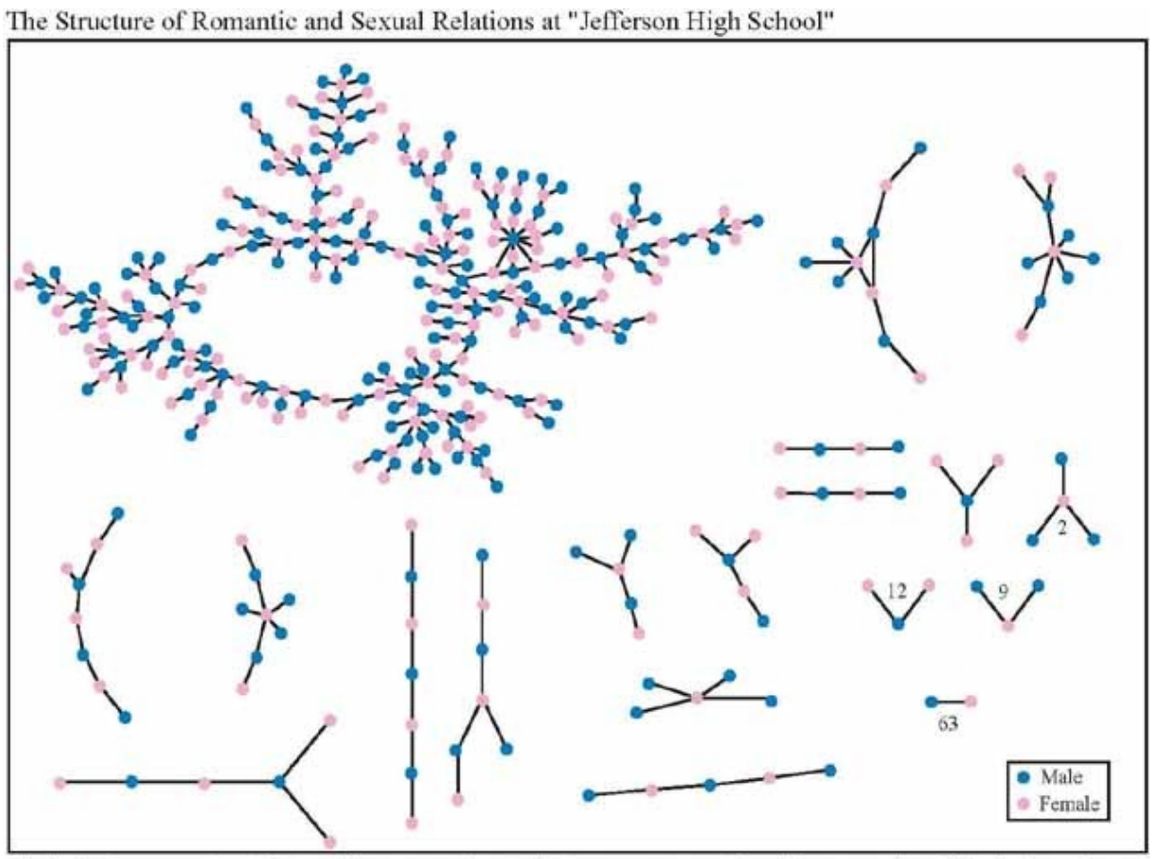


Scale-free networks - other examples

Sexual-interaction networks

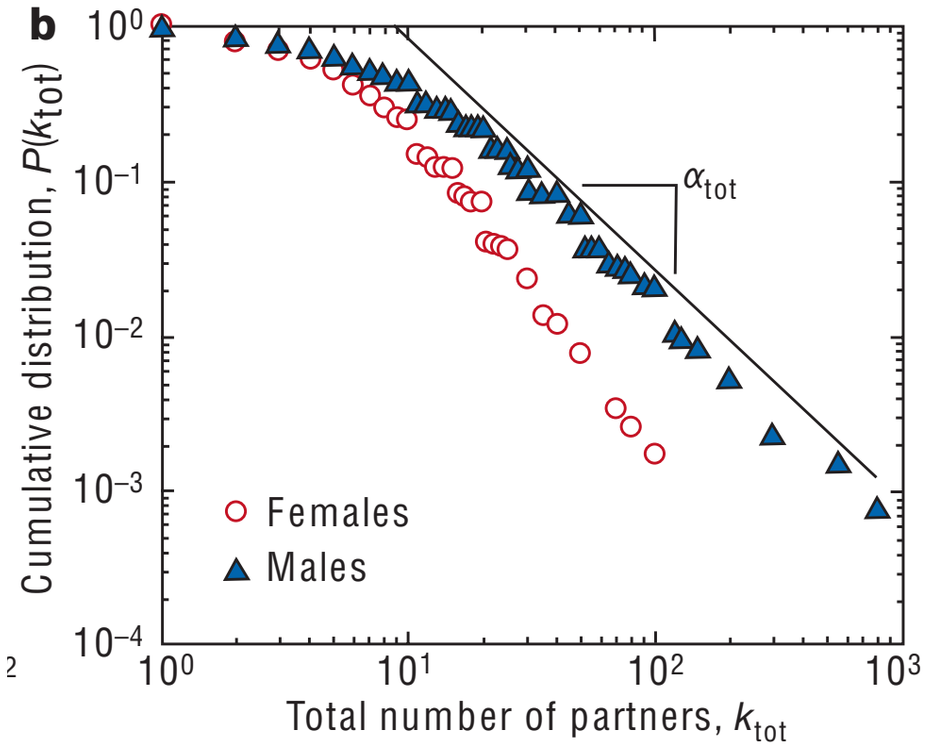
- Nodes: individuals
- Links: sexual incursion

Bearman et.al. (2004)



Each circle represents a student and lines connecting students represent romantic relations occuring within the 6 months preceding the interview. Numbers under the figure count the number of times that pattern was observed (i.e. we found 63 pairs unconnected to anyone else).

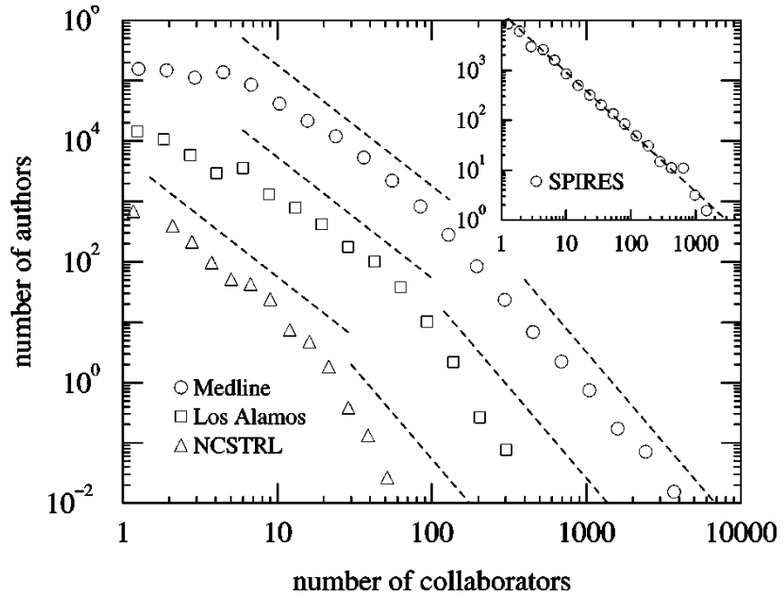
Liljeros et.al. (2001)



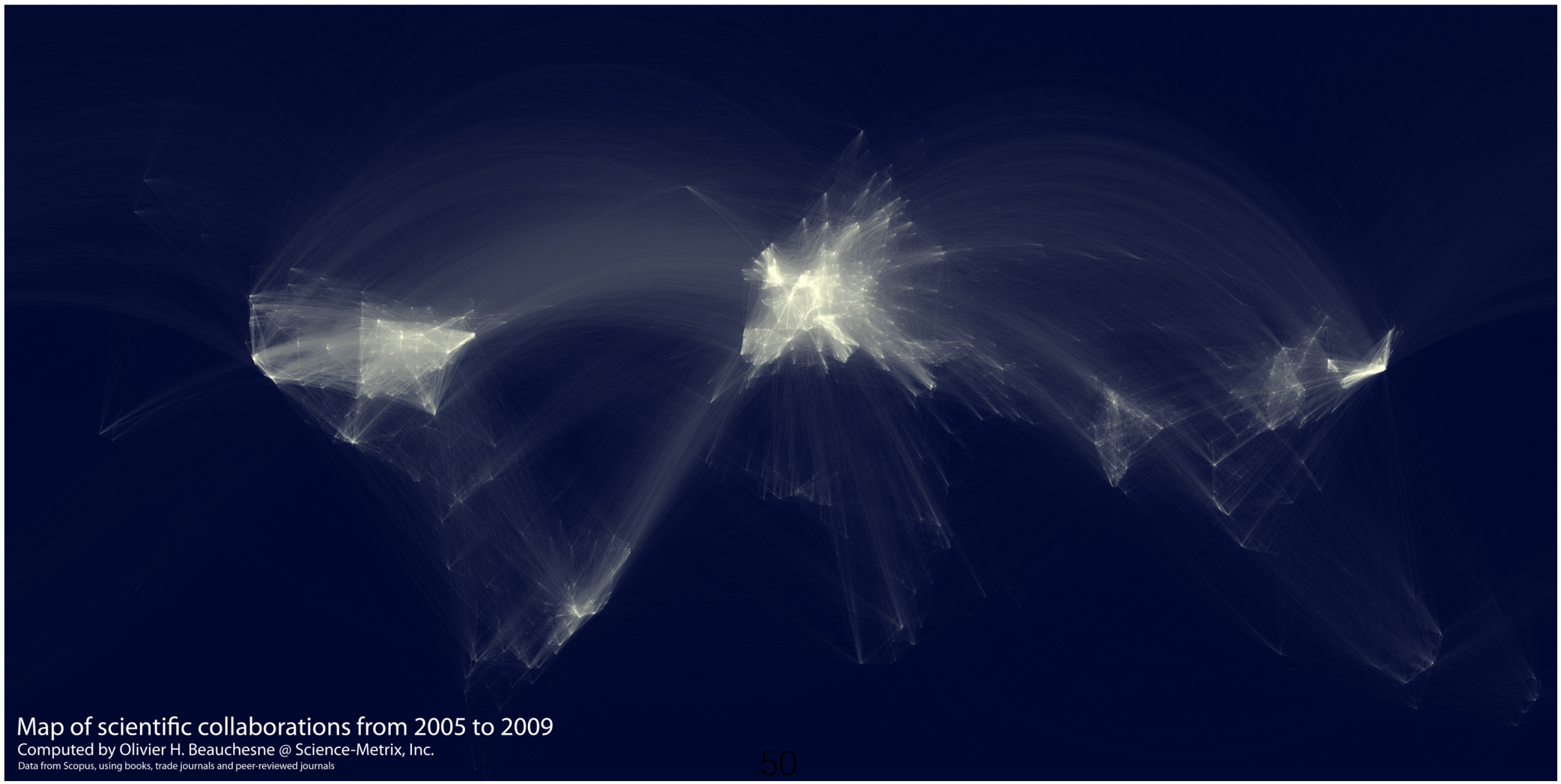
Scale-free networks - other examples

Scientific collaborations

- Nodes: scientists (here geo-localised)
- Links: common papers



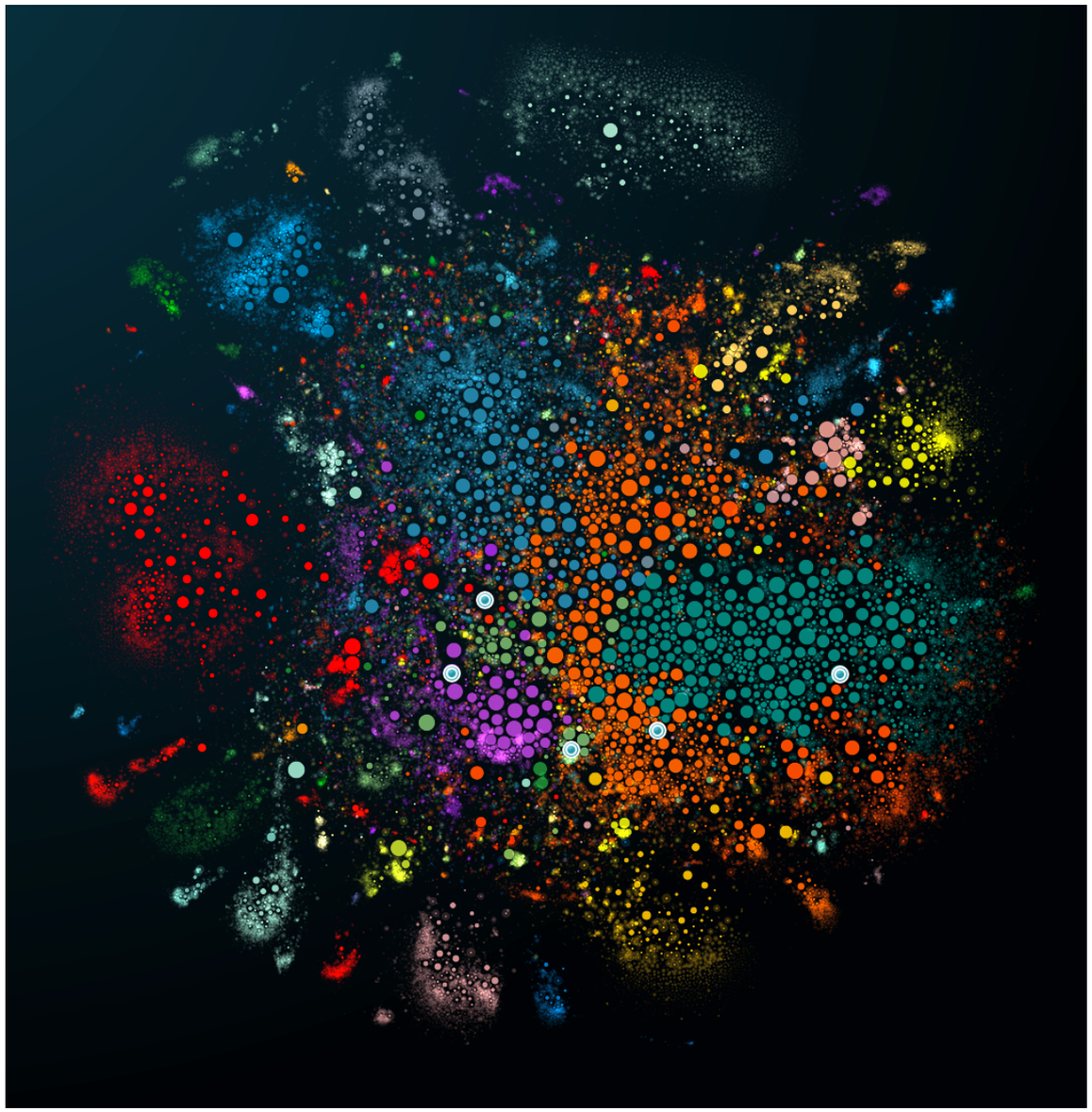
Newman (2001)



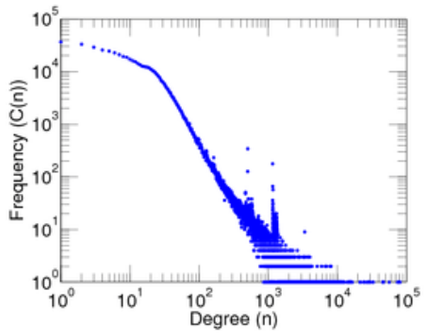
Scale-free networks - other examples

Online social networks

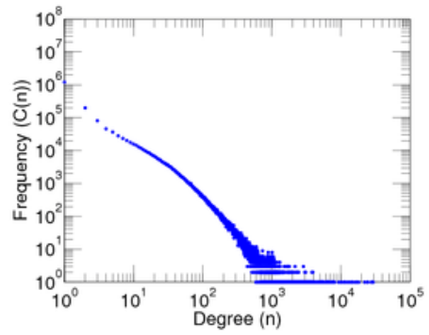
- Nodes: individuals
- Links: online interactions



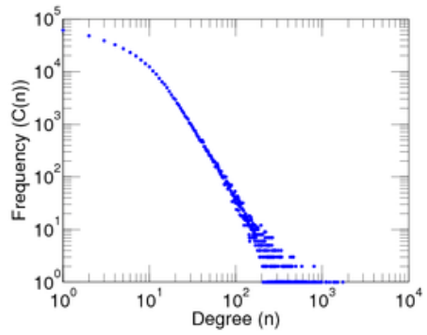
Social network of Steam
http://85.25.226.110/mapper



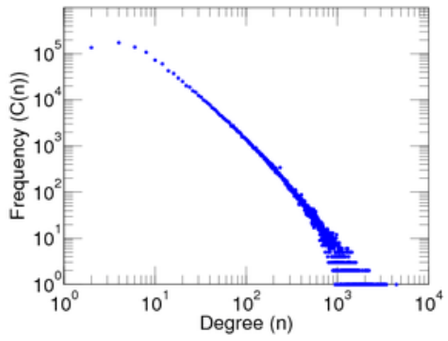
Catster/Dogster Familylinks/Friendships



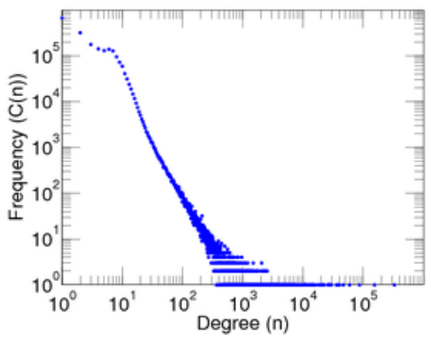
Chinese Wikipedia internal links



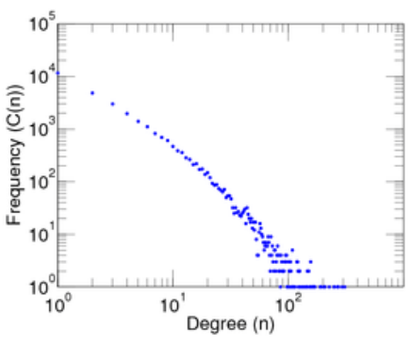
CiteSeer



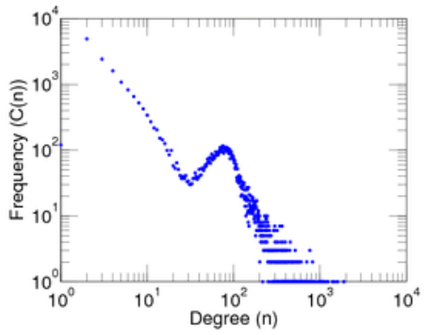
DBLP



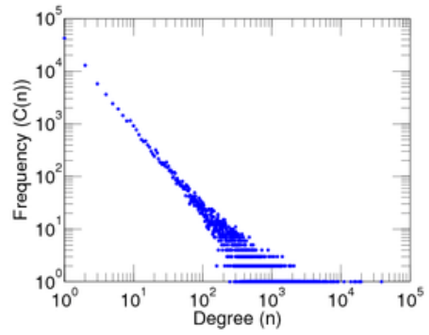
DBpedia



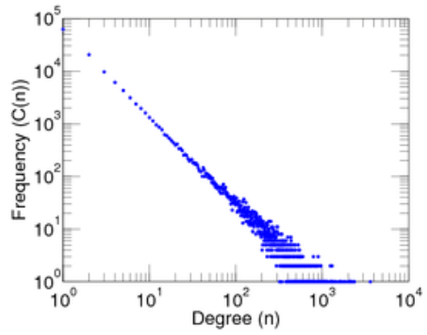
Digg



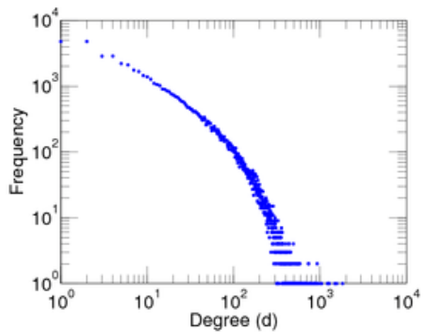
Edinburgh Associative Thesaurus



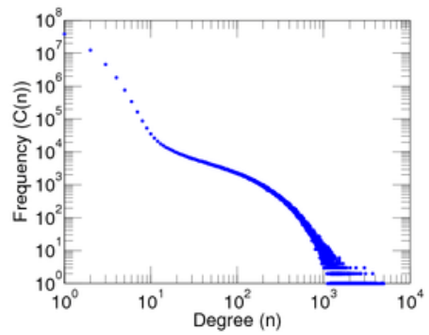
Enron



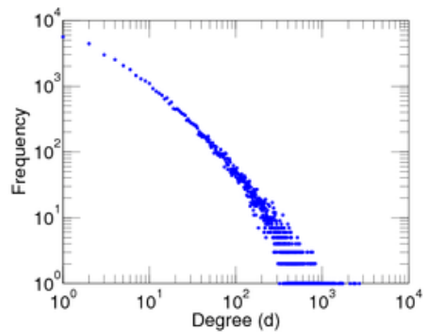
Epinions trust



Facebook friendships



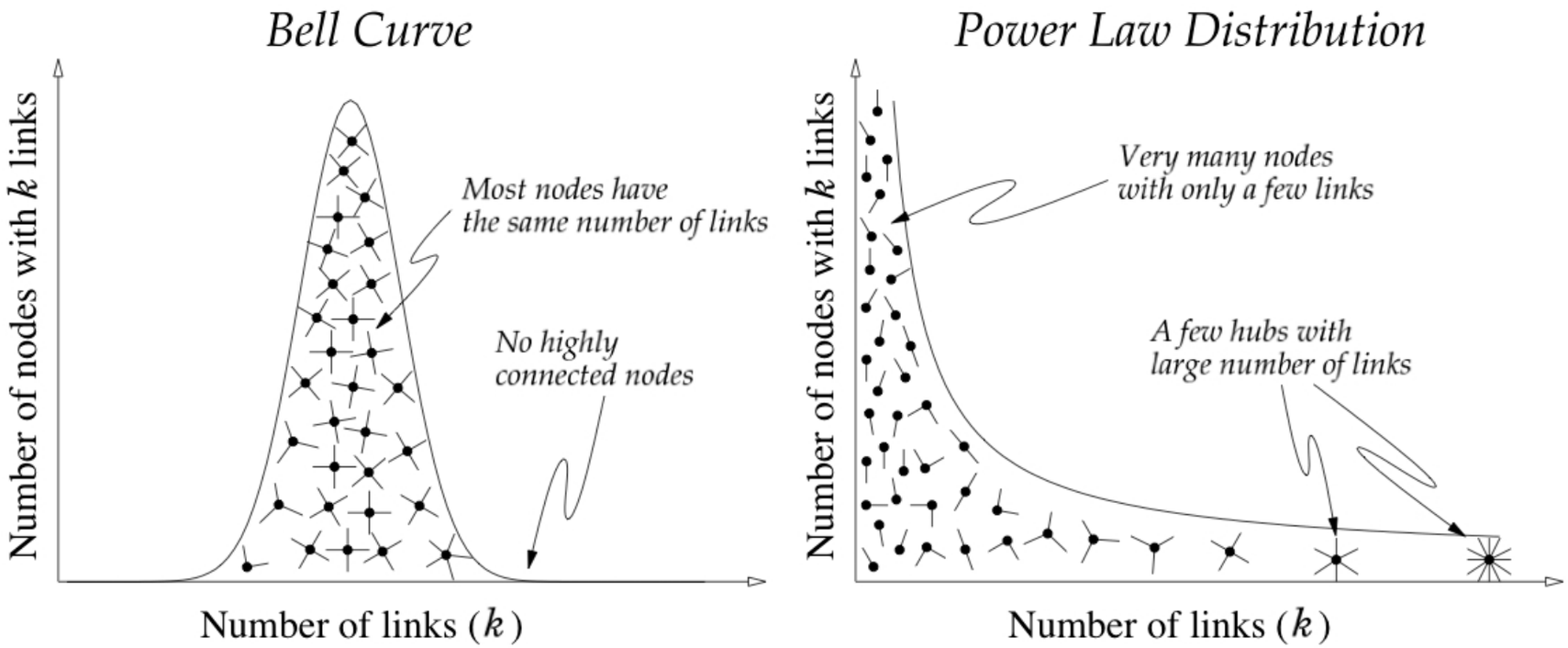
Facebook social graph

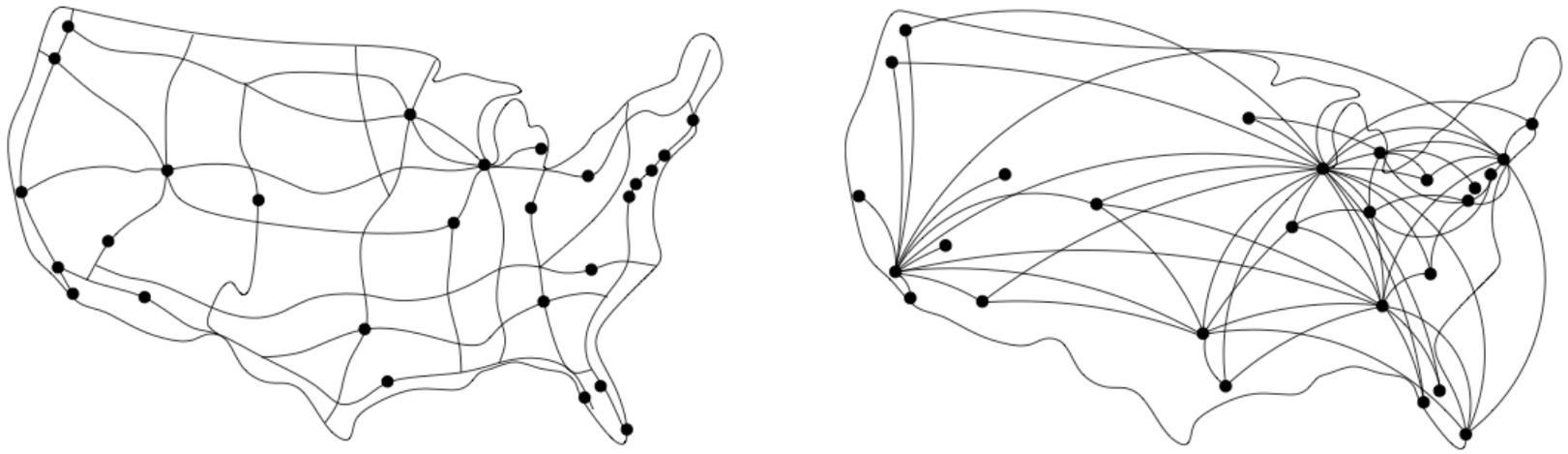


Facebook wall posts

Scale-free distribution

What does it mean?





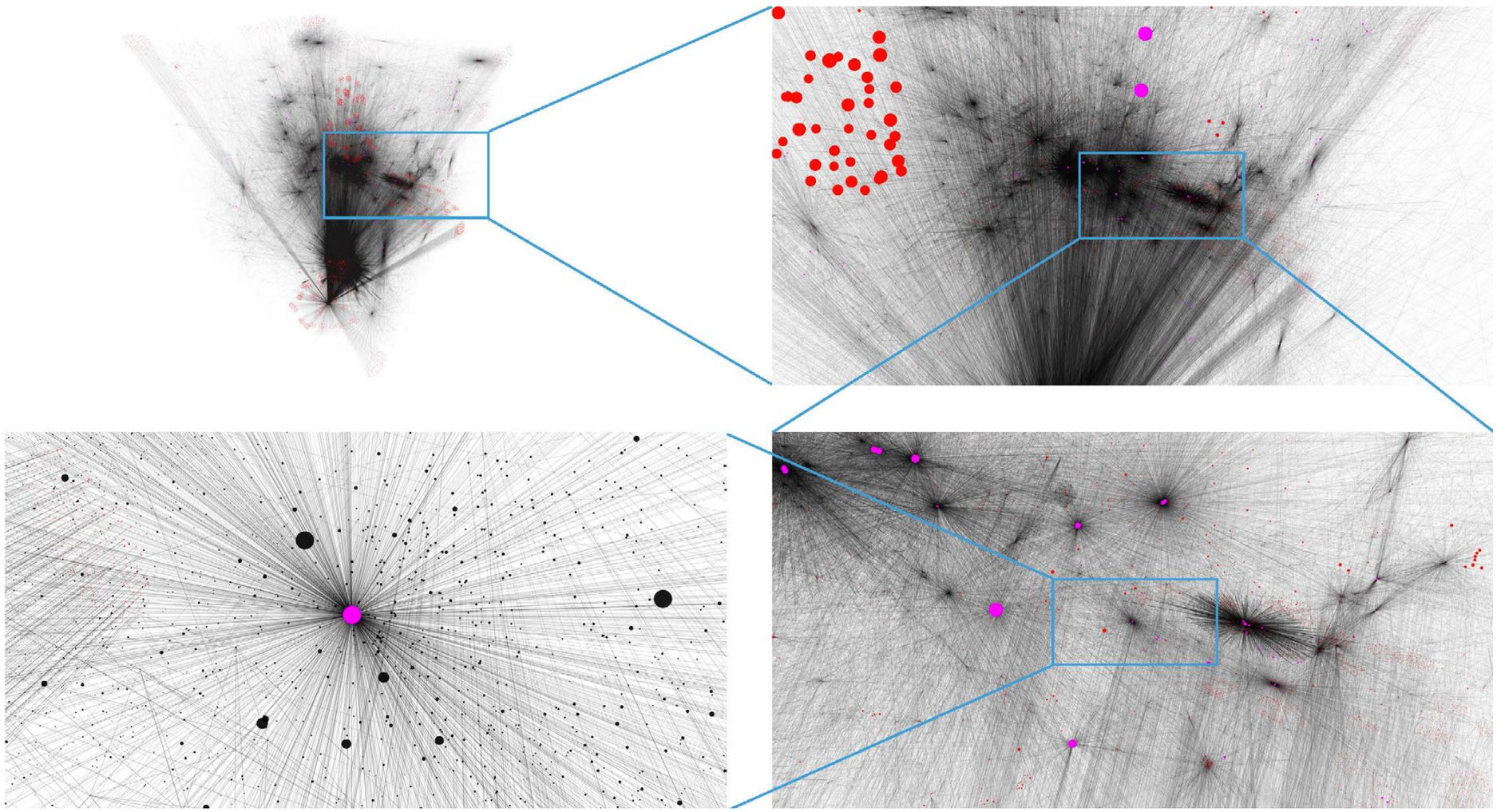
AL. Barabási, Linked (2002)

Degree fluctuations have no characteristic scale (scale invariant)

Scale-free networks

Idea of *scale free*

No typical scale, but also…



AL Barabási, Network Science Book (2013)

Scale-free networks

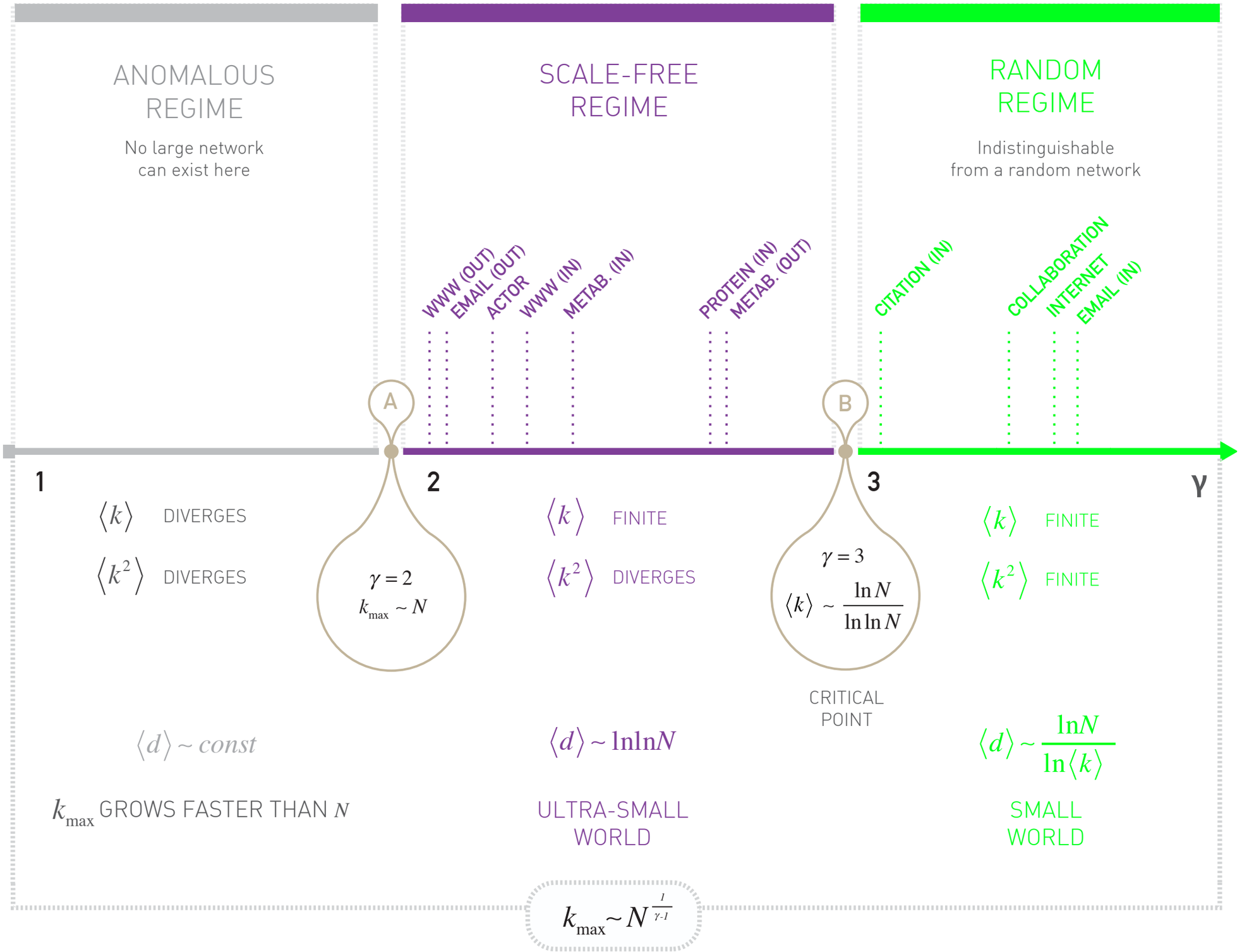
Network	Size	$\langle k \rangle$	κ	Exponent γ_{out}	Exponent γ_{in}
WWW	325 729	4.51	900	2.45	2.1
WWW	4×10^7	7		2.38	2.1
WWW	2×10^8	7.5	4000	2.72	2.1
WWW, site	260 000				1.94
Internet, domain*	3015–4389	3.42–3.76	30–40	2.1–2.2	2.1–2.2
Internet, router*	3888	2.57	30	2.48	2.48
Internet, router*	150 000	2.66	60	2.4	2.4
Movie actors*	212 250	28.78	900	2.3	2.3
Co-authors, SPIRES*	56 627	173	1100	1.2	1.2
Co-authors, neuro.*	209 293	11.54	400	2.1	2.1
Co-authors, math.*	70 975	3.9	120	2.5	2.5
Sexual contacts*	2810			3.4	3.4
Metabolic, *E. coli*	778	7.4	110	2.2	2.2
Protein, *S. cerev.**	1870	2.39		2.4	2.4
Ythan estuary*	134	8.7	35	1.05	1.05
Silwood Park*	154	4.75	27	1.13	1.13
Citation	783 339	8.57			3
Phone call	53×10^6	3.16		2.1	2.1
Words, co-occurrence*	460 902	70.13		2.7	2.7
Words, synonyms*	22 311	13.48		2.8	2.8

Albert, R. et.al. Rev. Mod. Phy. (2002)

- Average values are not reliable since the convergence is very slow
- Furthermore, average values are meaningless since the fluctuations are infinitely large (diverging variance)

Exponents of real-world networks are usually between 2 and 3

Scale-free networks - summary



Slide from CCNR Complex Networks Course
A. L. Barabási 2014

Scale-free networks

- Are real networks really Scale Free ?
- In most real networks, the scale free stands only for *a range* of degrees, i.e., between a minimum degree and maximum degree different than those observed (cut-offs)
- Some other distributions, in particular log-normal distributions, might “look like” power-law



Albert-László Barabási

Emergence of scaling in random networks (1999)



Aaron Clauset

Scale-free networks are rare (2018)

Love is All You Need - Clauset's fruitless search for scale-free networks (2018)



Petter Holme

Rare and everywhere: Perspectives on scale-free networks (2019)

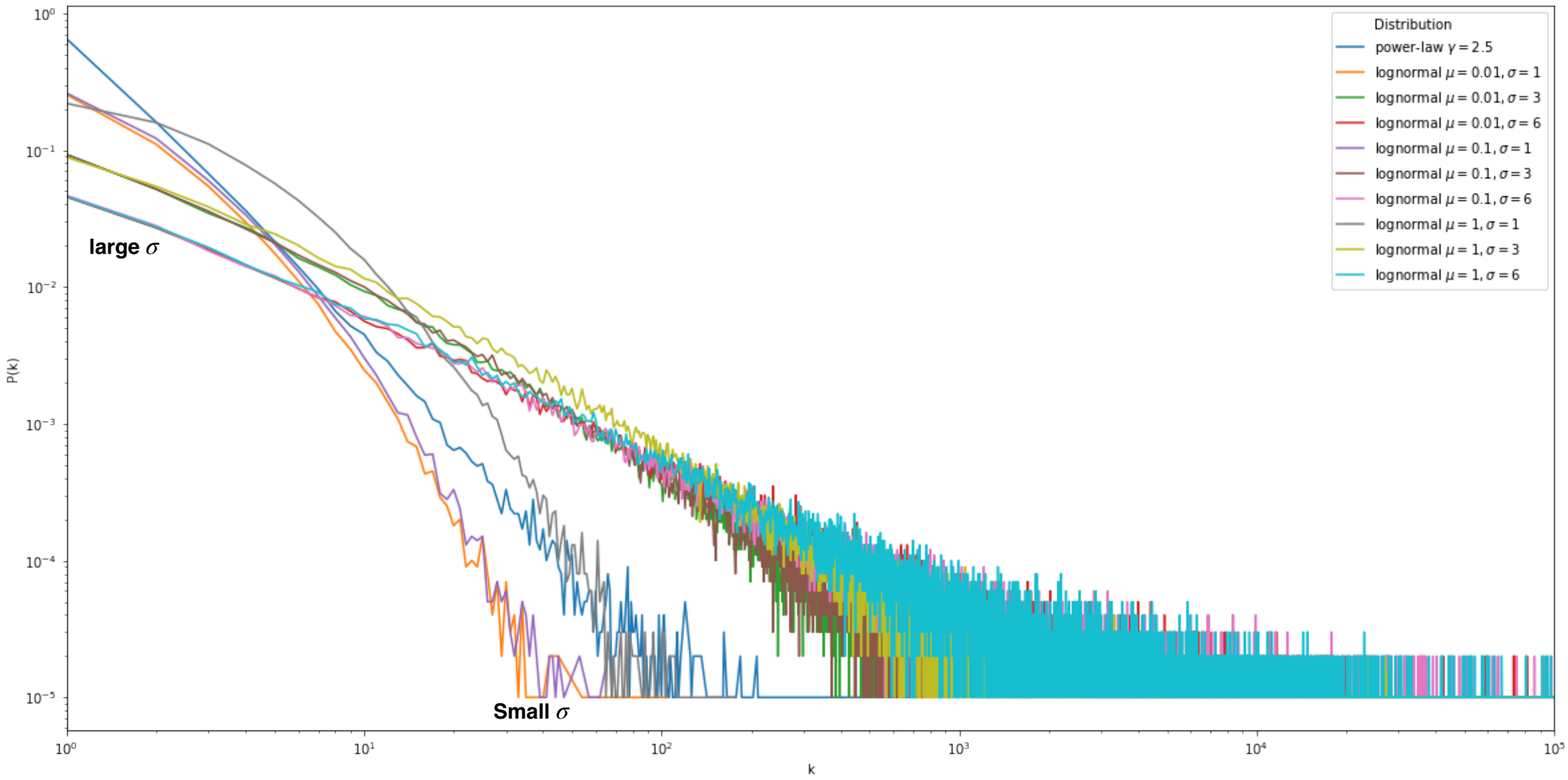
Scale-free networks

Comparing a log-normal distribution and a power law

Log-normal distribution = Probability distribution of a random variable whose logarithm is normally distributed

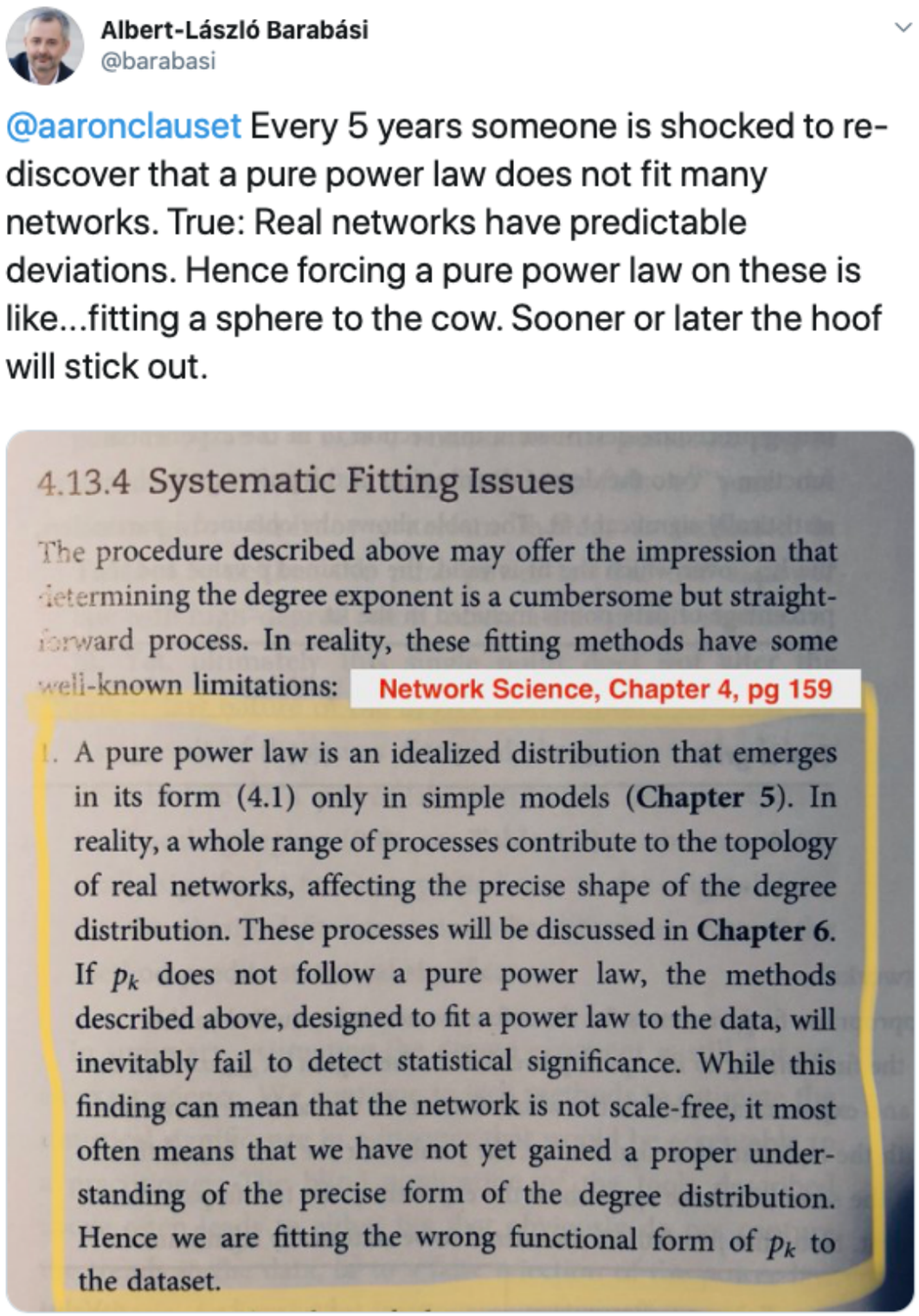
$$\frac{1}{k\sigma\sqrt{2\pi}}\ \exp\left(-\frac{\left(\ln k-\mu\right)^2}{2\sigma^2}\right)$$

$$k^{-\alpha}$$



μ
σ Mean, std of the log of the variable

Scale-free networks



Albert-László Barabási
@barabasi

@aaronclauset Every 5 years someone is shocked to re-discover that a pure power law does not fit many networks. True: Real networks have predictable deviations. Hence forcing a pure power law on these is like...fitting a sphere to the cow. Sooner or later the hoof will stick out.

4.13.4 Systematic Fitting Issues

The procedure described above may offer the impression that determining the degree exponent is a cumbersome but straightforward process. In reality, these fitting methods have some well-known limitations:

Network Science, Chapter 4, pg 159

1. A pure power law is an idealized distribution that emerges in its form (4.1) only in simple models (**Chapter 5**). In reality, a whole range of processes contribute to the topology of real networks, affecting the precise shape of the degree distribution. These processes will be discussed in **Chapter 6**. If p_k does not follow a pure power law, the methods described above, designed to fit a power law to the data, will inevitably fail to detect statistical significance. While this finding can mean that the network is not scale-free, it most often means that we have not yet gained a proper understanding of the precise form of the degree distribution. Hence we are fitting the wrong functional form of p_k to the dataset.

Albert-László Barabási @barabasi · Jan 15, 2018
Replying to @barabasi
Chapter 6 in Network Science networksciencebook.com/chapter/6 discusses what you should be fitting to the degree distribution of *real* scale-free networks. You are right: Pure power laws are predictably rare. Scale-free networks are not.
1 21 45

Aaron Clauset @aaronclauset · Jan 15, 2018
Replying to @barabasi
Yes, science is hard and real data often messy. But it is worrying how criticisms of harsh statistical evaluations can be interpreted as a belief that "disagreement with data" (as Feynman would put it) should not be held against a favored theory or model.
3 5 18

Albert-László Barabási @barabasi · Jan 15, 2018
We are on the same page. The question is, what you test and what you conclude. There are multiple processes that contribute to the degree distribution that modify the power law. Hence testing for power laws only you are ignoring them all, leading to misleading takeway message.
2 4 10

Aaron Clauset @aaronclauset · Jan 15, 2018
Perhaps. I feel good about the accuracy of our conclusions: we used rigorous statistical methods, tested 5 distributions, considered 5 levels of evidence, across nearly 1000 network datasets. The goal was to be thorough and to treat the SF hypothesis as falsifiable.
1 3 14

Albert-László Barabási @barabasi · Jan 15, 2018
The effort is amazing. The conclusions are less so. The feather falls slower than the rock, yet gravitation is not wrong. We add friction. You need to fit for each system the Pk that is right for it. That is hard, I know. Otherwise you ignore 20 year of work by hundreds.
2 4 6

Aaron Clauset @aaronclauset · Jan 15, 2018
It seems easy to get confused here: an empirical power-law degree distribution is evidence for SF structure, but no deviation from the power law can be evidence against SF structure? It is reasonable to believe a fundamental phenomena would require less customized detective work.

Scale-free networks



Aaron Clauset

-Rigorous **statistical tests** show that observed degree distributions are not compatible with a power law distribution (high p-values)

-Compared with different distributions, in particular log-normal, most degree distributions are more likely to be generated by something else than power laws



Albert-László Barabási

-Networks are real objects, not mathematical abstraction, therefore they are sensible to **noise** (real life limits…)



-Power law is a **good, simple model** of degree distributions of a class of networks

-20 years of **fruitful research** based on this model

A whole scientific article dedicated to the controversy:
Jacomy, M. (2020). Epistemic clashes in network science: Mapping the tensions between idiographic and nomothetic subcultures. *Big Data & Society*, *7*(2), 2053951720949577.

The Barabási-Albert model of scale-free networks

Emergence of hubs

What did we miss with the earlier network models?

1. Networks are evolving
   - Networks are not static but growing in time as new nodes are entering the system

2. Preferential attachement
   - Nodes are not connected randomly but tends to link to more attractive nodes

- Pólya urn model (1923)
- Yule process (1925)
- Zipf's law (1941)
- Cumulative advantage (1968)
- Preferential attachement (1999)
- Pareto's law - 80/20 rule
- The rich get richer phenomena
- etc.

AL Barabási, Network Science Book (2013)

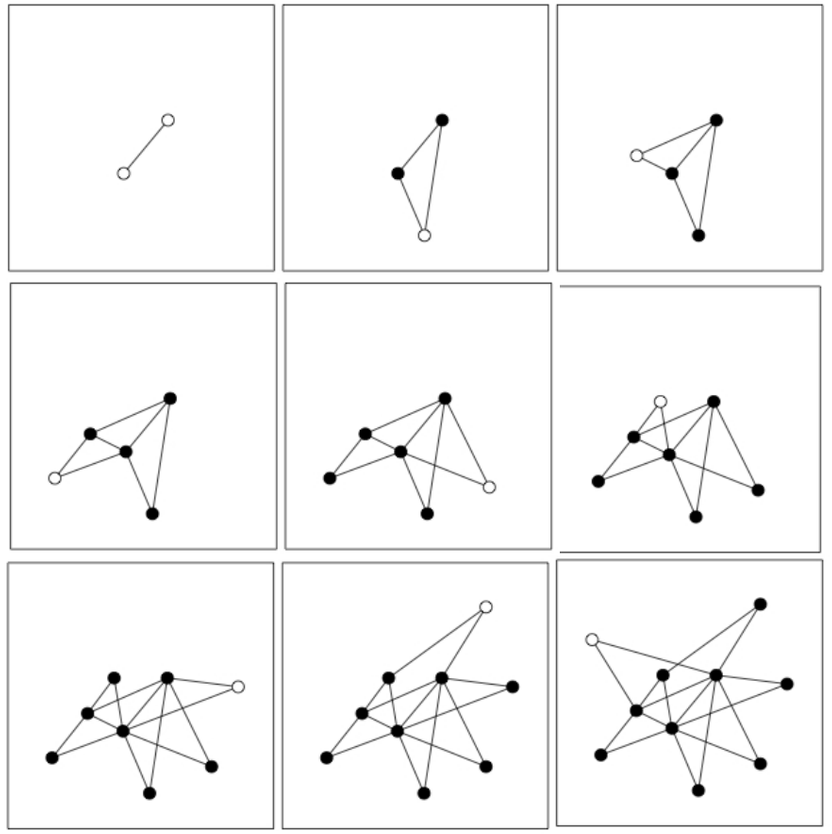
The Barabási-Albert model

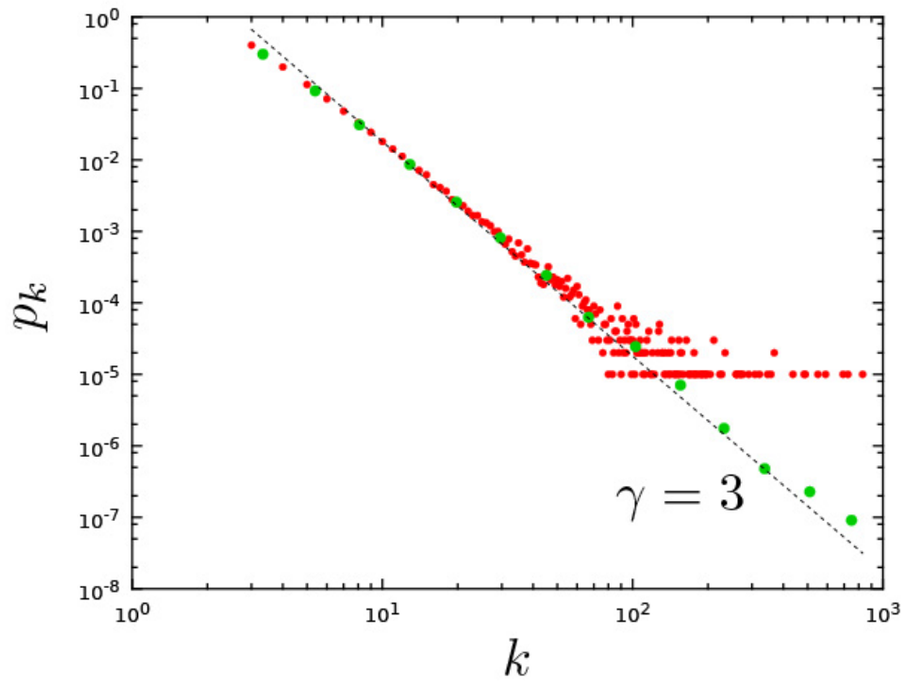
1. Start with m_0 connected nodes

2. At each timestep we add a new node with $m\ (\leq m_0)$ links that connect the new node to m nodes already in the network.

3. The probability $\pi(k)$ that one of the links of the new node connects to node i depends on the degree k_i of node i as

$$\Pi(k_i) = \frac{k_i}{\sum_j k_j}$$





- The emerging network will be scale-free with degree exponent $\gamma=3$ independently from the choice of m_0 and m

AL Barabási, Network Science Book (2013)

ER Random Network - catch up

Network	Degree distribution	Path length	Clustering coefficient
Real world networks	broad	short	large
Regular lattices	constant	long	large
ER random networks	Poissonian	short	small
WS small-world networks	exponential	short	large
BA scale-free networks	power-law	short	Rather small

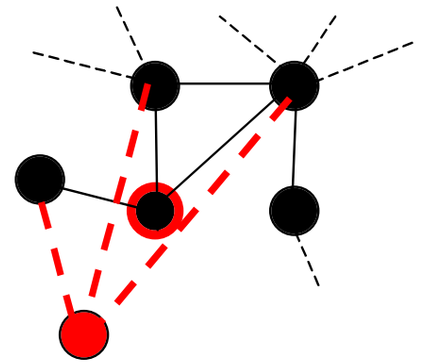
Other scale-free models

The vertex-copying model

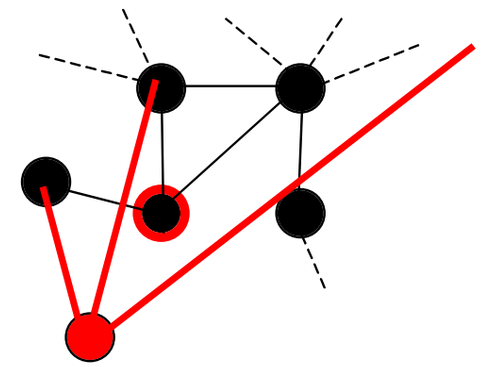
- Motivation:
  - Citations network or WWW where links are often copied
  - Local explanation to preferential attachement

1. Take a small seed network
2. Pick a random vertex
3. Make a copy of it
4. With probability p, move each edge of the copy to point to a random vertex
5. Repeat 2.-4. until the network has grown to desired size of N vertices

1. copy a vertex



2. rewire edges with *p*



- Asymptotically scale-free with exponent $\gamma \geq 3$

ER Random Network - catch up

Network	Degree distribution	Path length	Clustering coefficient
Real world networks	broad	short	large
Regular lattices	constant	long	large
ER random networks	Poissonian	short	small
WS small-world networks	exponential	short	large
BA scale-free networks	power-law	short	Rather small
Other models	power-law	short	Large

Notes

- “All models are wrong, but some are useful”
- ER models, Configuration models are used as reference models in a large number of applications
- WS, BA models are more “making a point” type models: simple processes can explain some non-trivial properties of networks, unfound in random networks.
- Correlation is not causation. Are these simple processes the “cause” ? Maybe, maybe not, sometimes…

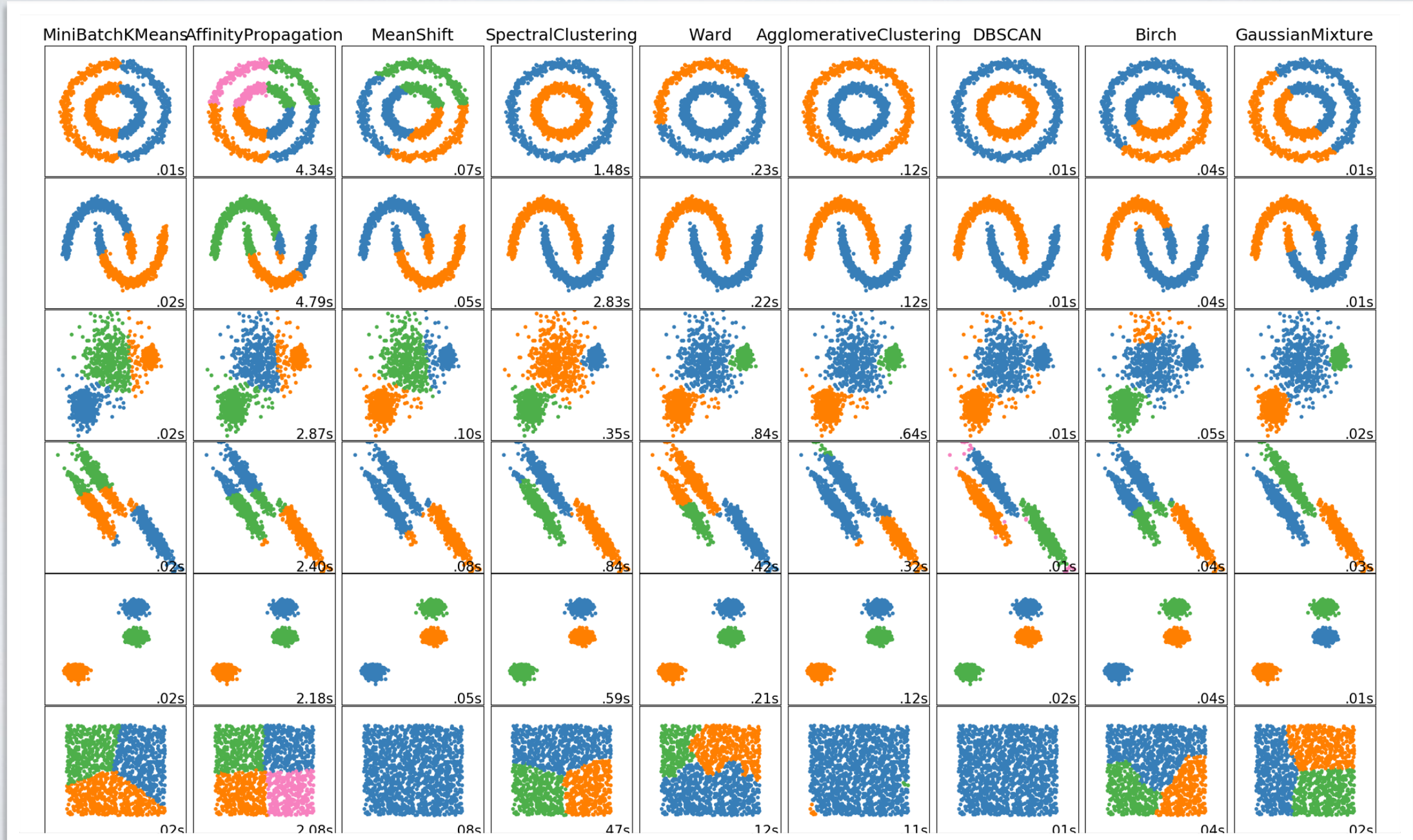
COMMUNITY DETECTION (GRAPH CLUSTERING)

COMMUNITY DETECTION

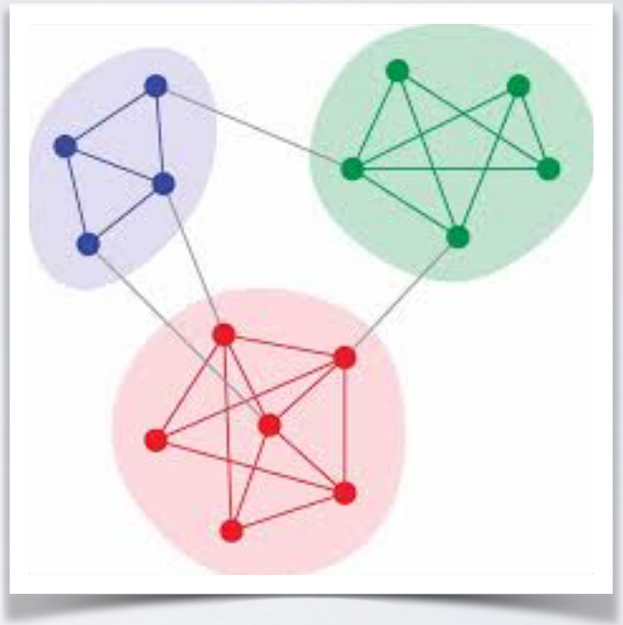
- Community detection is equivalent to "clustering" in unstructured data

- Clustering: unsupervised machine learning
  - Find groups of elements that are similar to each other
    - People based on DNA, apartments based on characteristics, etc.
  - Hundreds of methods published since 1950 (k-means)
  - Problem: what does "similar to each other" means ?

COMMUNITY DETECTION



COMMUNITY DETECTION



- Community detection:
  - Find groups of nodes that are:
    - Strongly connected to each other
    - Weakly connected to the rest of the network
    - Ideal form: each community is 1)A clique, 2) A separate connected component
  - No formal definition
  - Hundreds of methods published since 2003

WHY COMMUNITY DETECTION ?

- One of the key properties of complex networks was
  - High clustering coefficient
  - (friends of my friends are my friends)

- Different from random networks. How to explain it ?
  - Watts strogatz (spatial structure?)

- => In real networks, presence of dense groups: communities
  - Small, dense (random) networks have high density.
  - Large networks could be interpreted as aggregation of smaller, denser networks, with much fewer edges between them

SOME HISTORY

- The *graph partitioning problem* was a classic problem in graph theory

- It goes like this:
  - How to split a network in **k** equal parts such that there is a minimal number of edges between parts.
  - Variants were proposed:
    - What if partitions are not exactly same size ?
    - What if the number of parts is not exactly k ?
    - …

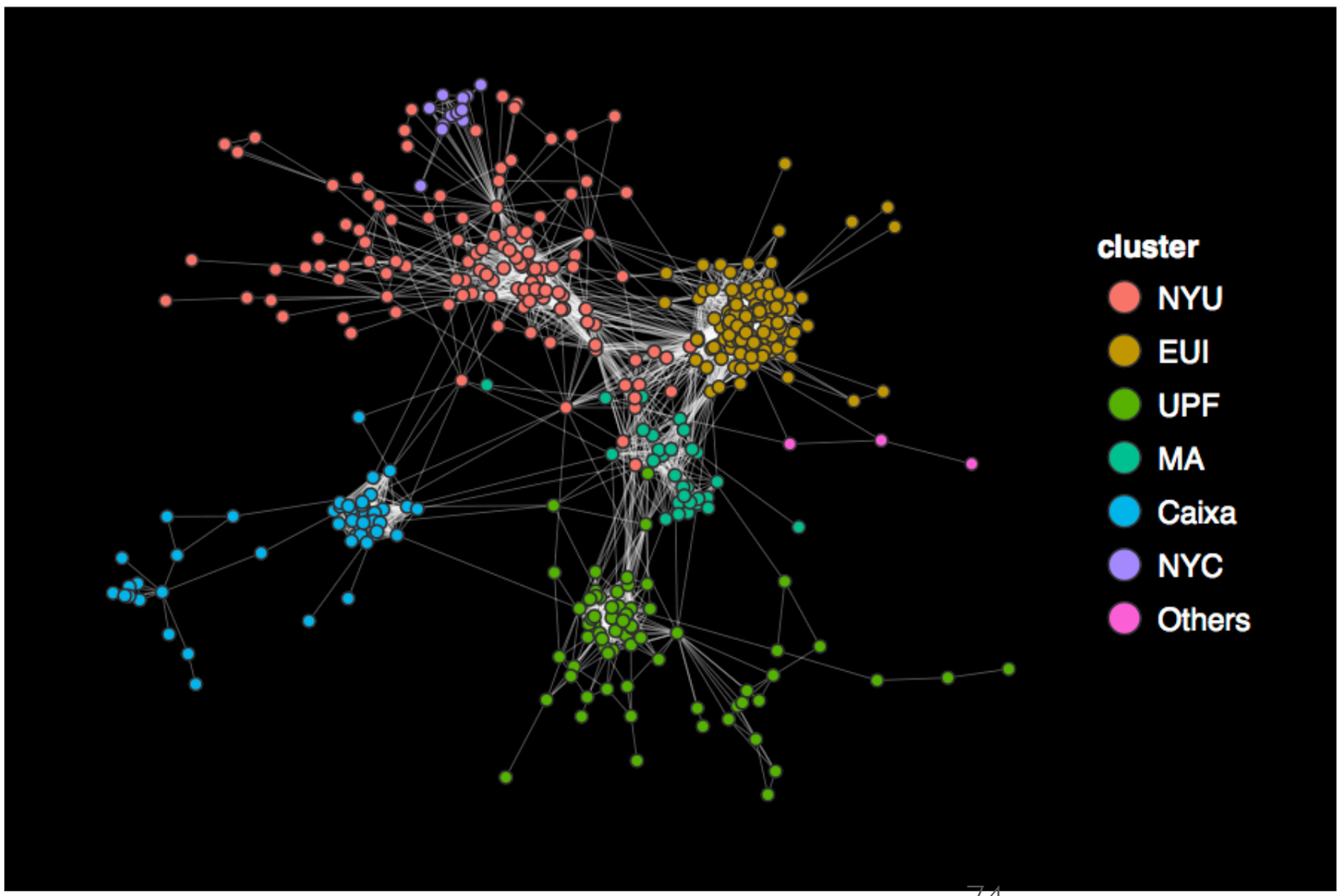
SOME HISTORY

- Then in 2002, [Girvan & Newman 2002], introduction of the problem of "community discovery":
  - Observation that social networks are very often composed of groups
  - The number and the size of these groups is not known in advance
  - Can we design an algorithm to discover automatically those groups ?

Girvan, Michelle, and Mark EJ Newman. "Community structure in social and biological networks." *Proceedings of the national academy of sciences* 99.12 (2002): 7821-7826.

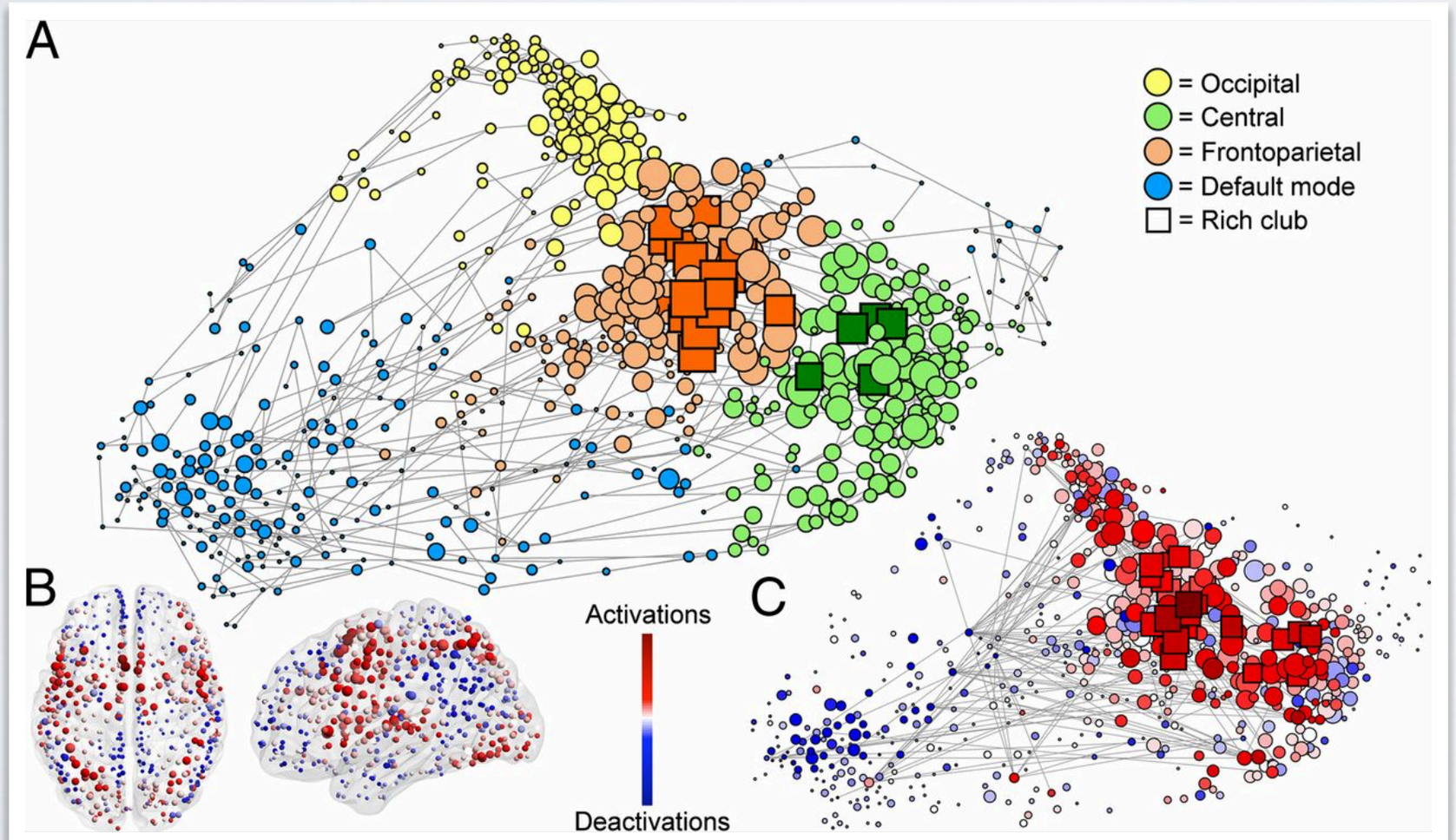
COMMUNITY STRUCTURE IN REAL GRAPHS

- If you plot the graph of your facebook friends, it looks like this



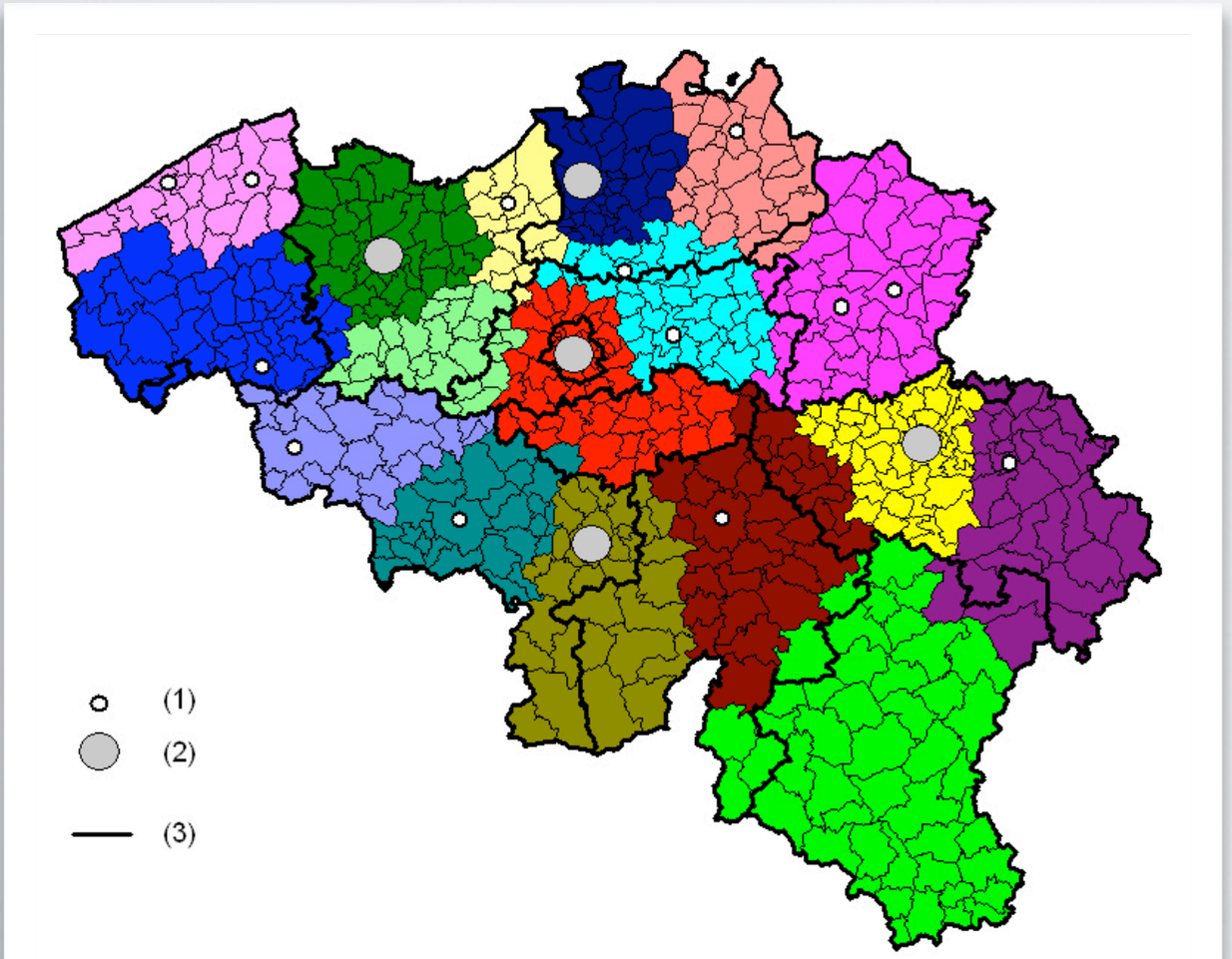
COMMUNITY STRUCTURE IN REAL GRAPHS

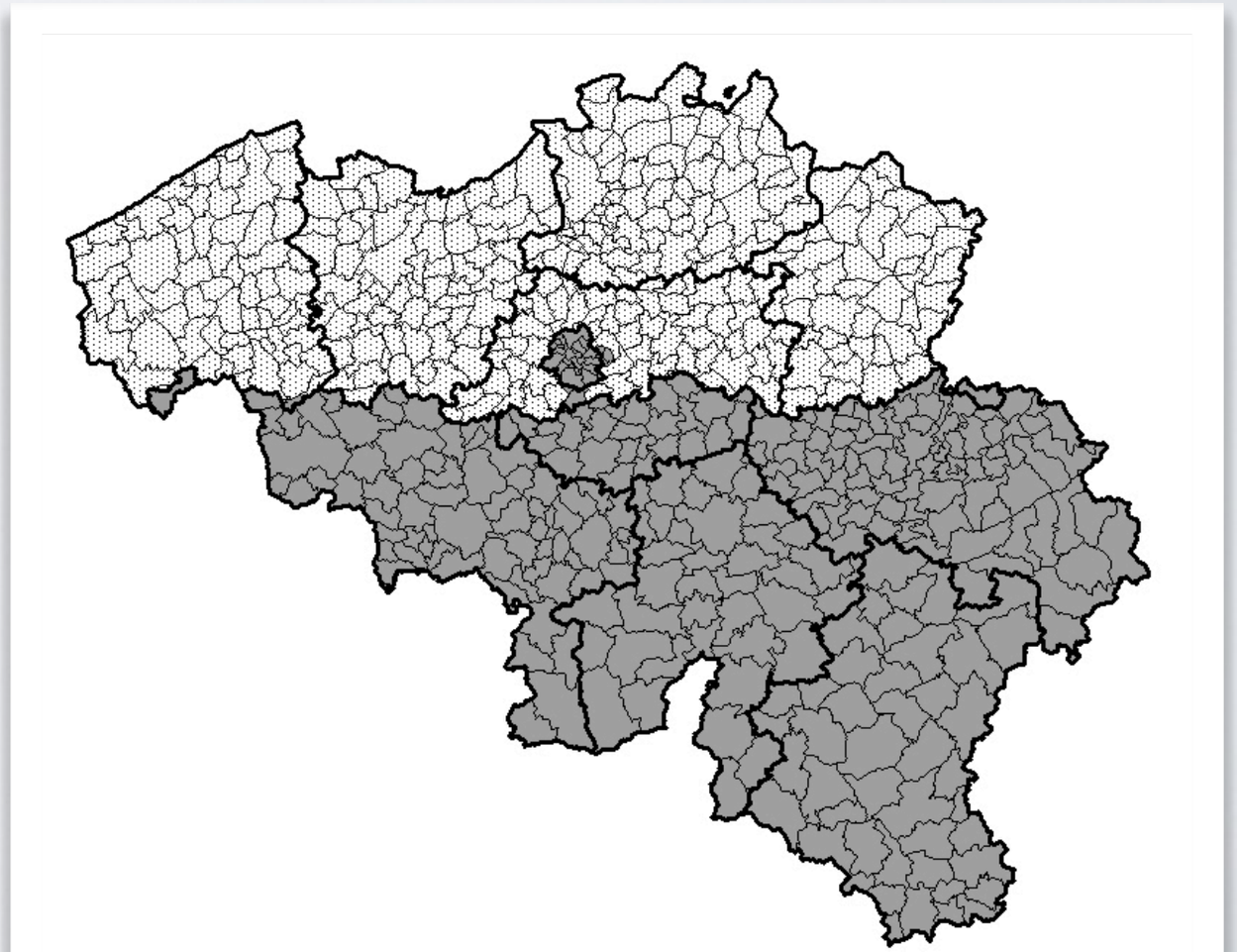
- Connections in the brain ?



COMMUNITY STRUCTURE IN REAL GRAPHS

- Phone call communications in Belgium ?

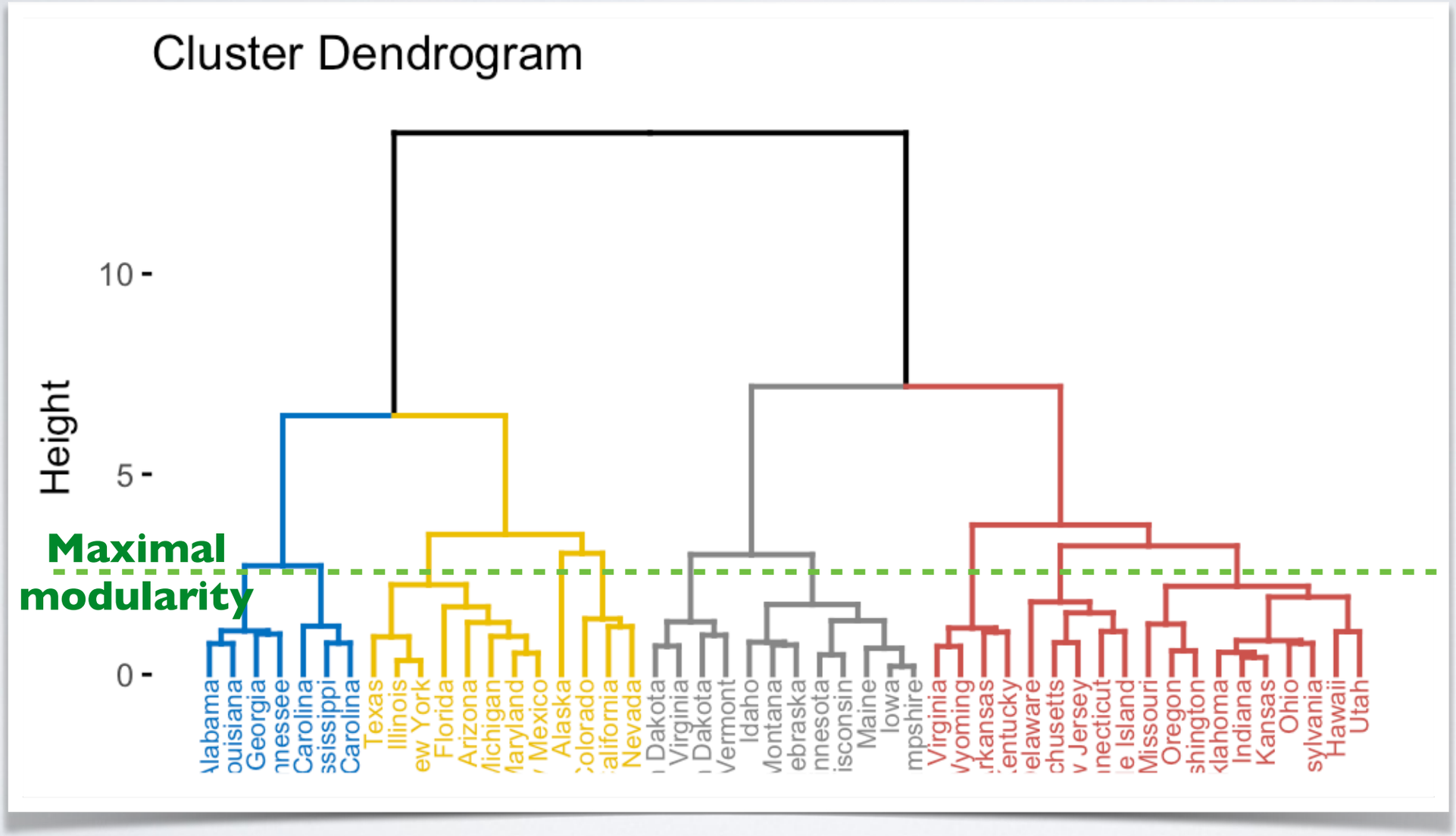




FIRST METHOD BY GIRVAN & NEWMAN

- 1)Compute the betweenness of all edges
- 2)Remove the edge of highest betweenness
- 3)Repeat until all edges have been removed
  - Connected components are communities
- => It is called a *divisive* method
- =>What you obtain is a dendrogram
- How to cut this dendrogram at the *best* level ?

FIRST METHOD BY GIRVAN & NEWMAN



FIRST METHOD BY GIRVAN & NEWMAN

- Introduction of the **Modularity**

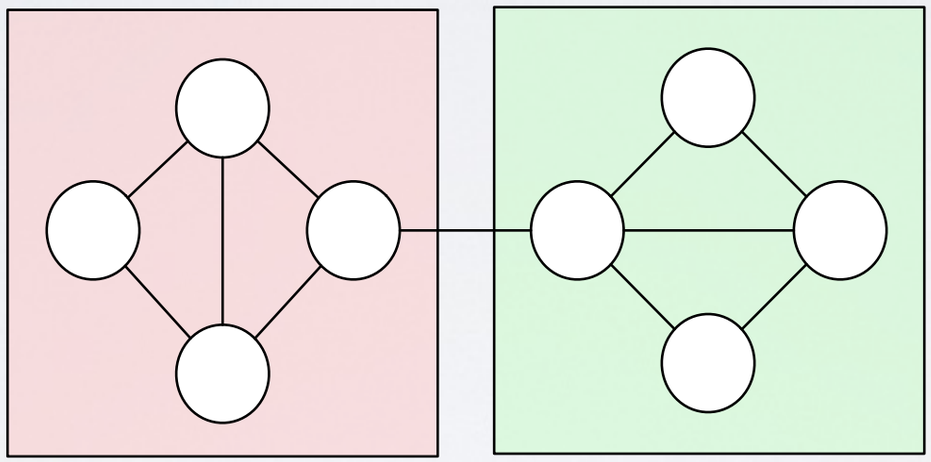
- The modularity is computed for a partition of a graph
  - (each node belongs to one and only one community)

- It compares :
  - The **observed** *fraction of edges inside communities*
  - To the **expected** *fraction of edges inside communities* in a random network
  - Observed - Expected

MODULARITY INTUITION

$n = 8$

$m = 11$



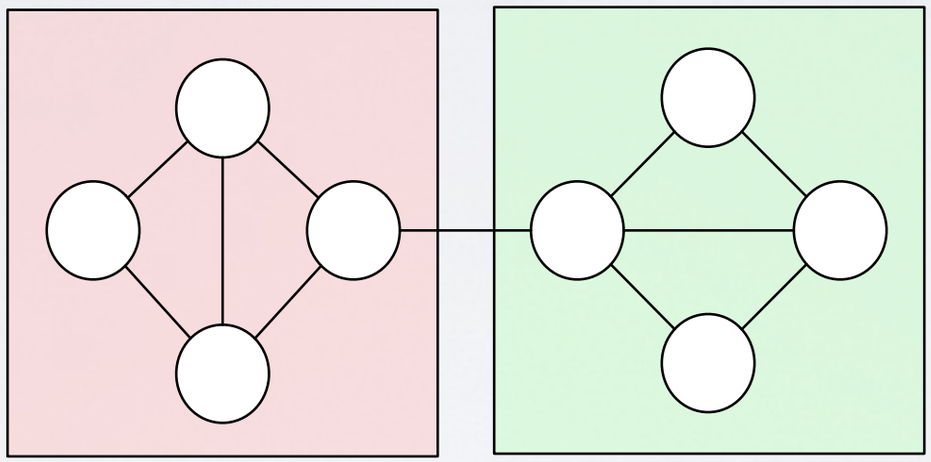
MODULARITY INTUITION

$n = 8$

$m = 11$

$p(u, v) \approx 0.39$

$$d(G) = p(u, v) = \frac{11}{\frac{1}{2}8(8-1)} = \frac{11}{28} \approx 0.39$$

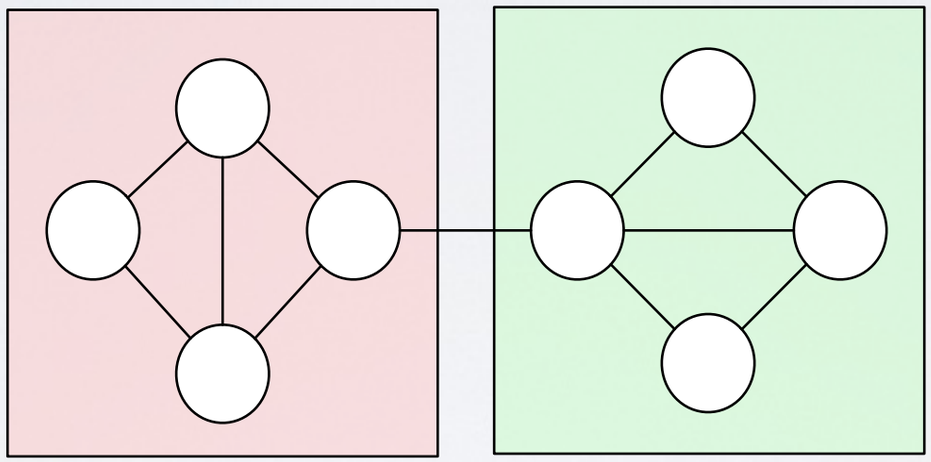


MODULARITY INTUITION

$n = 8$

$m = 11$

$p(u, v) \approx 0.39$



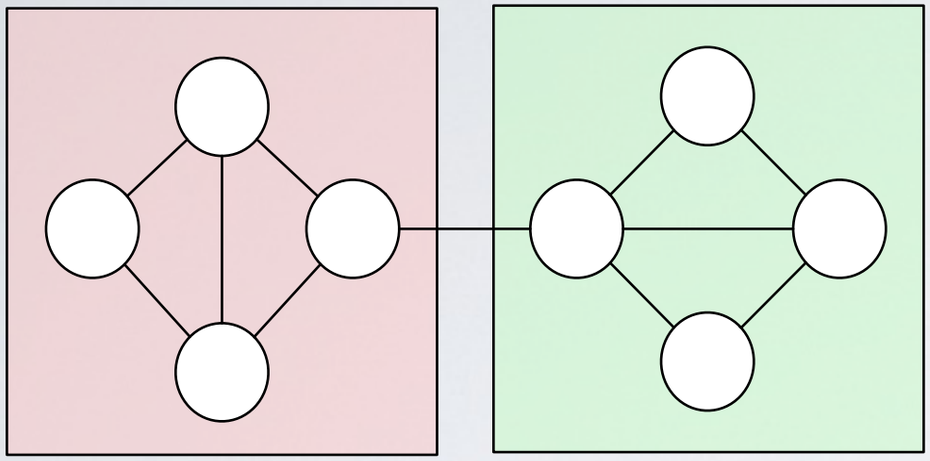
ER random graph

Expected edges inside red (or green)
(#node pairs * prob to observe an edge)

$$\frac{4(4-1)}{2} * p(u, v) = 2.34$$

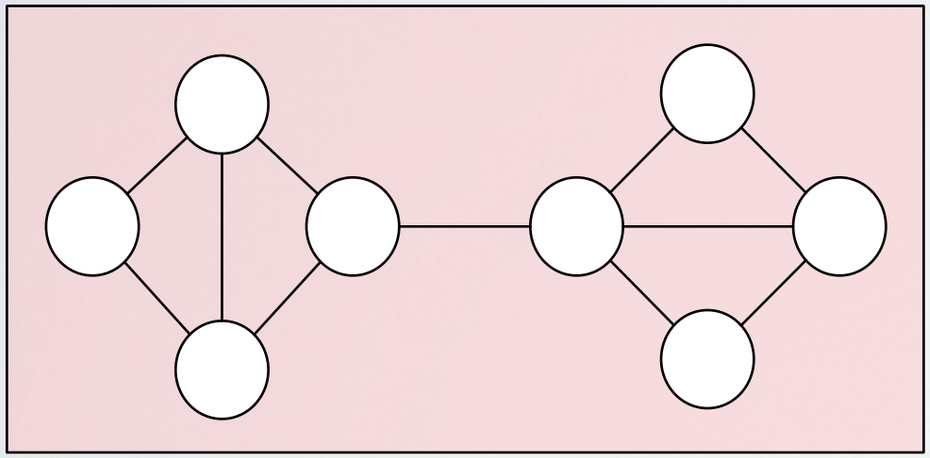
$$\text{Modularity} = \frac{2(5 - 2.34)}{m} = 0.48$$

MODULARITY INTUITION

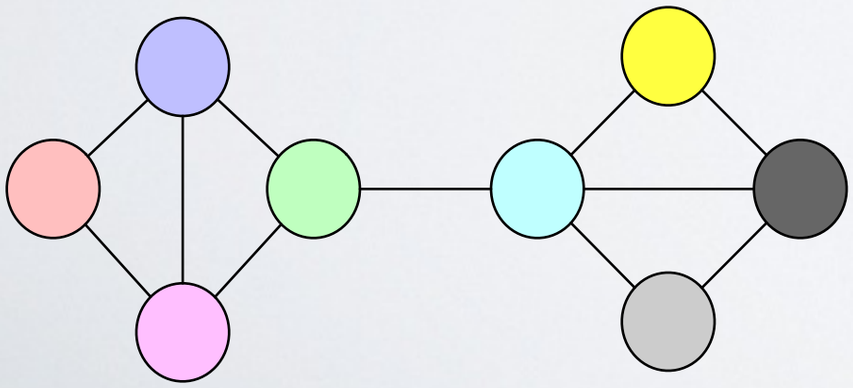


$n = 8$
$m = 11$

Q= 0.48



Q= ?



Q= ?

MODULARITY NULL MODEL

- In previous examples, we used ER as a null model

- Usual approach: configuration model as null model
  - Preserves each node's degree
  - $p(u,v) = \frac{k_u k_v}{2m}$

MODULARITY

$$Q = \frac{1}{(2m)} \sum_{vw} \left[A_{vw} - \frac{k_v k_w}{(2m)} \right] \delta(c_v, c_w)$$

Original formulation

MODULARITY

$$Q = \frac{1}{(2m)} \sum_{vw} \left[A_{vw} - \frac{k_v k_w}{(2m)} \right] \delta(c_v, c_w)$$

Sum over all pairs of nodes

MODULARITY

$$Q = \frac{1}{(2m)} \sum_{vw} \left[A_{vw} - \frac{k_v k_w}{(2m)} \right] \delta(c_v, c_w)$$

1 if in same community

MODULARITY

$$Q = \frac{1}{(2m)} \sum_{vw} \left[A_{vw} - \frac{k_v k_w}{(2m)} \right] \delta(c_v, c_w)$$

1 if there is an edge between them

MODULARITY

$$Q = \frac{1}{(2m)} \sum_{vw} \left[A_{vw} - \frac{k_v k_w}{(2m)} \right] \delta(c_v, c_w)$$

Probability of an edge in a configuration model

MODULARITY

Can also be defined as a sum by community

$$Q = \frac{1}{L} \sum_{i=1}^{|C|} (L_i - \frac{K_i^2}{4L})$$

with $L_i = L(H(c_i))$ the number of edges inside community i and $K_i = \sum_{u \in c_i} k_u$ the sum of degrees of nodes in community i.

FIRST METHOD BY GIRVAN & NEWMAN

- Back to the method:
  - Create a dendrogram by removing edges
  - Cut the dendrogram at the best level using modularity

- =>In the end, your objective is… to optimize the Modularity, right ?

- Why not optimizing it directly !

MODULARITY OPTIMIZATION

- From 2004 to 2008: The golden age of Modularity
- Scores of methods proposed to optimize it
  - Graph spectral approaches
  - Meta-heuristics approches (simulated annealing, multi-agent…)
  - Local/Gloabal approaches…
- => 2008: the Louvain algorithm

LOUVAIN ALGORITHM

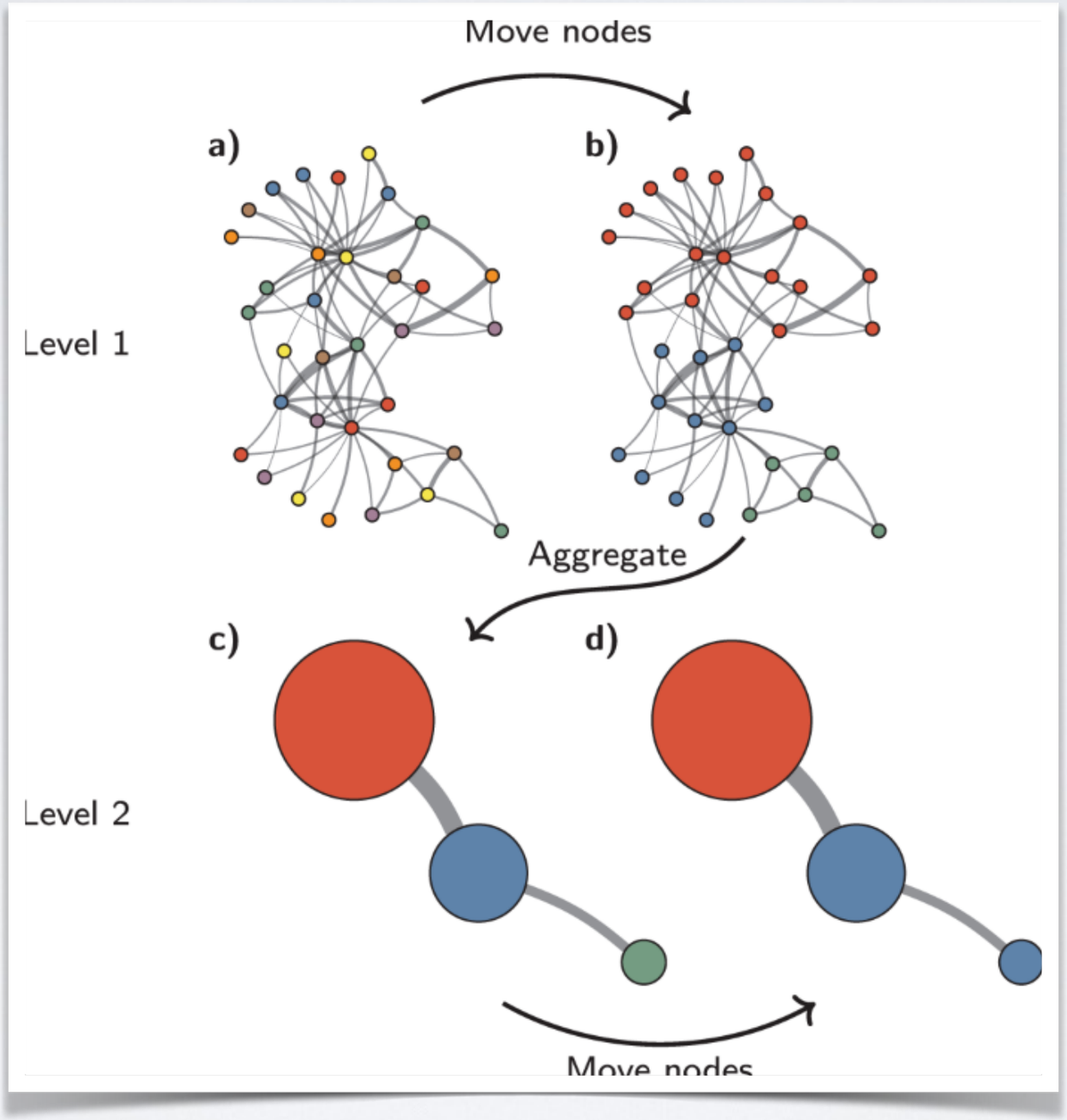
- Simple, greedy approach
  - Easy to implement
  - Fast
- Yields a hierarchical community structure
- Efficient and convenient open-source code

LOUVAIN ALGORITHM

- Each node start in its own community
- Repeat until convergence
  - FOR each node:
    - FOR each neighbor:
      if adding node to its community increase modularity, do it
- When converged, create an *induced network*
  - Each community becomes a node
  - Edge weight is the sum of weights of edges between them
  - /!\: Modularity on the induced graph is equal to Modularity computed on the graph itself, just a "trick" to search a better optimum

Blondel, Vincent D., et al. "Fast unfolding of communities in large networks." *Journal of statistical mechanics: theory and experiment* 2008.10 (2008): P10008.

LOUVAIN ALGORITHM



Blondel, Vincent D., et al. "Fast unfolding of communities in large networks." *Journal of statistical mechanics: theory and experiment* 2008.10 (2008): P10008.

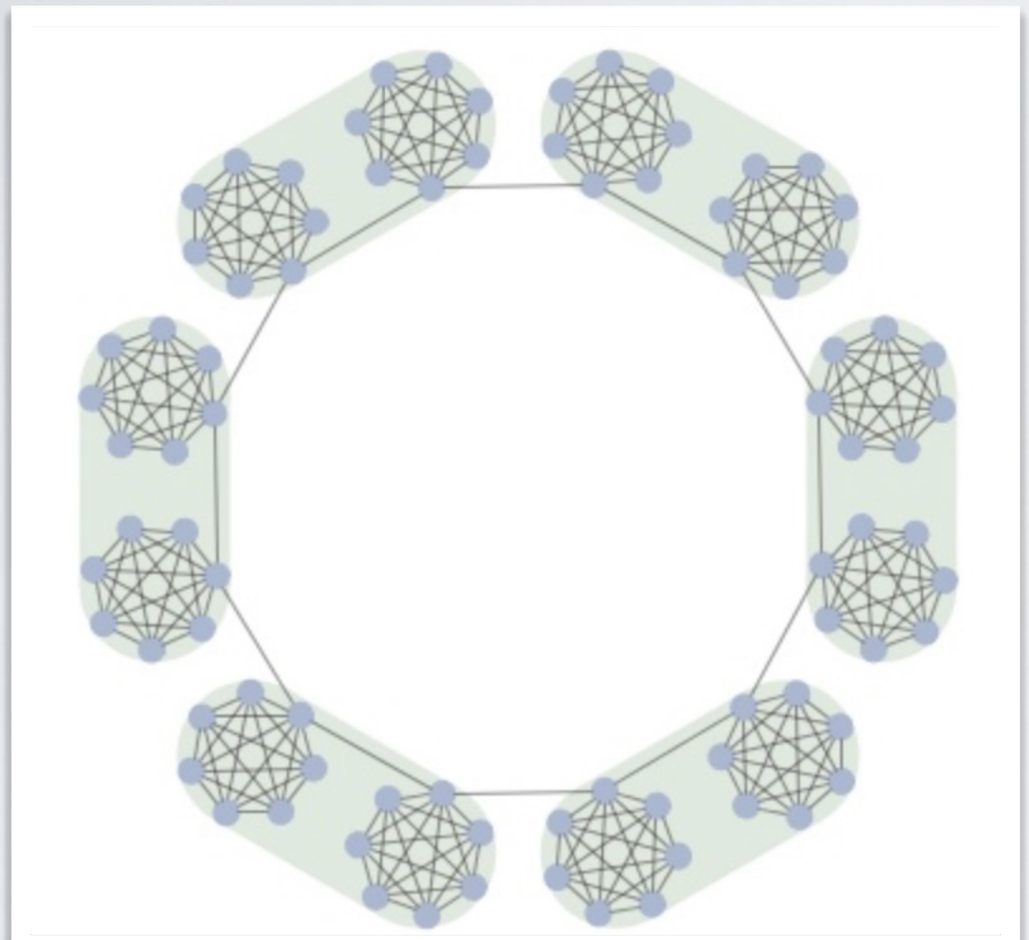
RESOLUTION LIMIT

- Modularity == Definition of good communities ?

- 2006-2008: series of articles [Fortunato,Lancicchinetti,Barthelemy]
  - The **resolution limit** of Modularity

Fortunato, Santo, and Marc Barthelemy. "Resolution limit in community detection." *Proceedings of the national academy of sciences* 104.1 (2007): 36-41.

RESOLUTION LIMIT



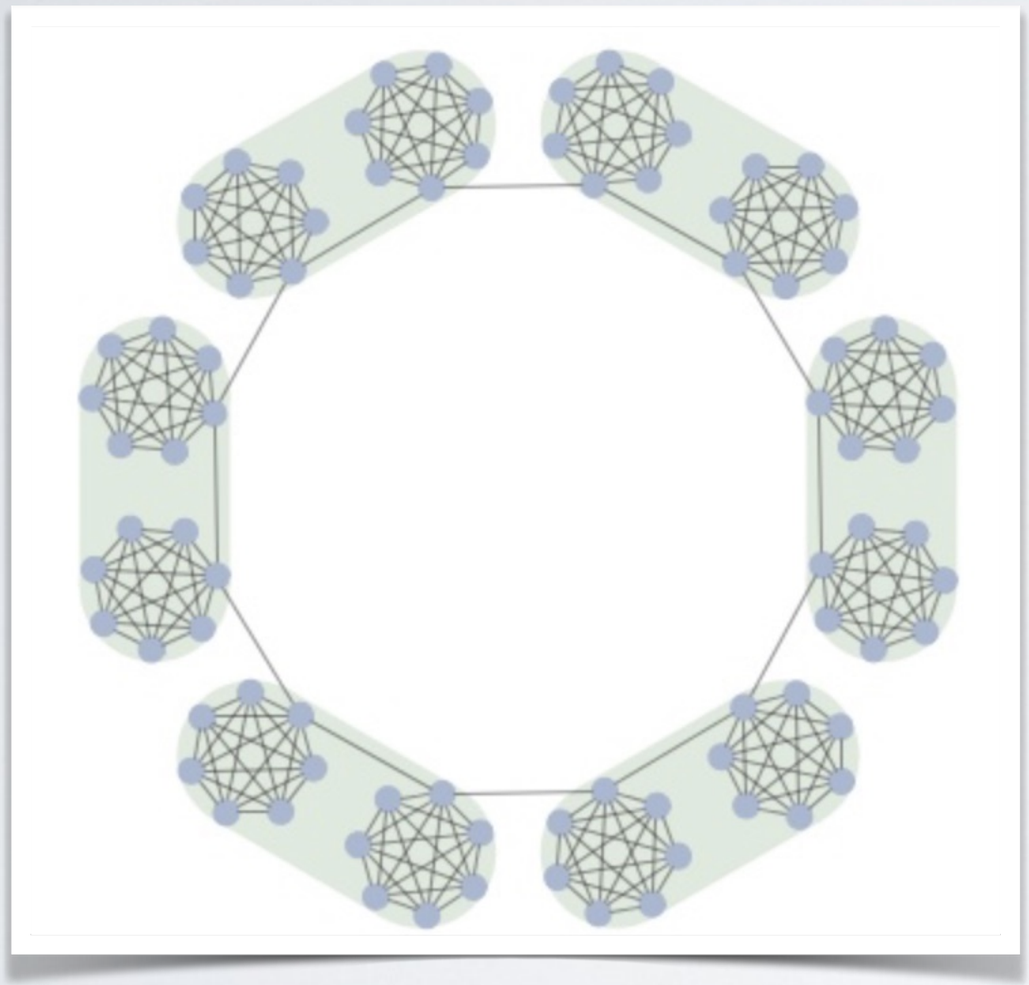
Let's consider a ring of cliques

Cliques are as dense as possible

Single edge between them:
=>As separated as possible

Any acceptable algorithm=>Each clique is a community

RESOLUTION LIMIT



But with modularity:

Small graphs=> OK

Large graphs=>
The max of modularity obtained by merging cliques

RESOLUTION LIMIT

- Discovery that Modularity has a "favorite scale":

- For a graph of given **density** and **size:**
  - Communities cannot be smaller than a fraction of nodes

- Modularity optimization will never discover small communities in large networks

RESOLUTION LIMIT

- Multi-resolution modularity

$$\sum_{i}^{c} e_{ii} - a_i^2 \quad \Rightarrow \quad \sum_{i}^{c} e_{ii} - \lambda a_i^2$$

λ = Resolution parameter

More a patch than a solution…

APPROXIMATION

- Modularity is an exact null model for
  - A multigraph
  - With self-loops
- Applied to simple graphs, it is a reasonable approximation for
  - large, sparse graphs
  - with large communities
  - Without too many giant hubs
  - $\Rightarrow \frac{k_u k_v}{2m} > 1 \Rightarrow$negate modularity on a node pair even if edge present

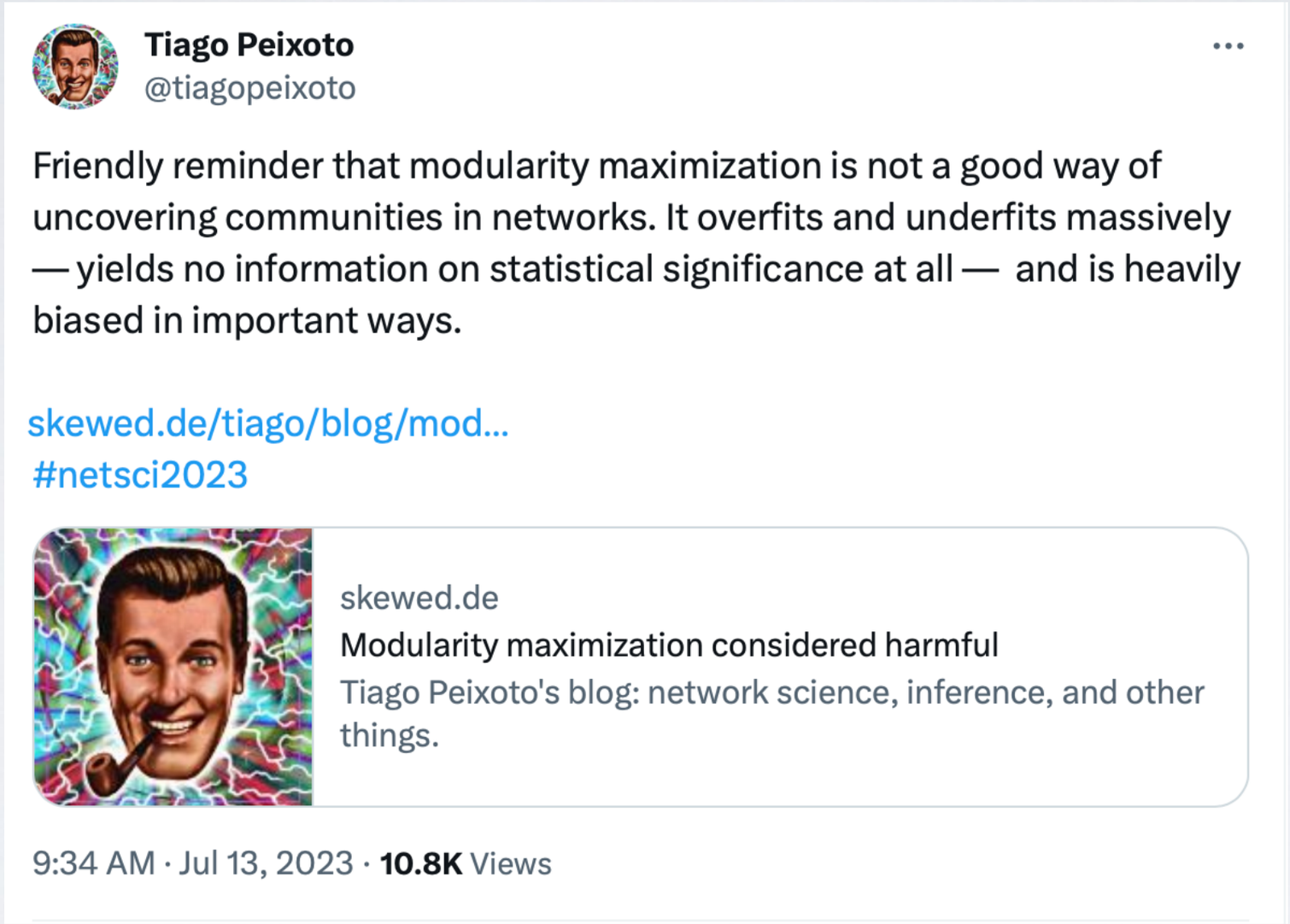
OTHER WEAKNESSES

- Modularity has other controversial/not-intuitive properties:
  - Finding communities in random graphs:
    - Network without community structure: Max modularity for partitions driven by random noise + approximation
  - Global measure => a difference in one side of the network can change communities at the other end (imagine a growing clique ring…)

- To this day, Louvain and modularity remain most used methods
  - Results are usually "good"/useful
  - It's a "standard" tool, like k-means or linear regression
  - Actually more and more challenged[1]

[1] Peixoto, T. P. (2023). *Descriptive vs. inferential community detection in networks: Pitfalls, myths and half-truths*. Cambridge University Press.

OTHER WEAKNESSES



ALTERNATIVES

- 1000+ Algorithms published, and counting
- Common saying:
  - "No algorithm is better than another, it depends on the type of network"(no free-lunch theorem)
  - "The best method depends on the objective"
- Still an open question in the field

ALTERNATIVES

- Most serious alternatives (in my opinion)
  - Infomap (based on information theory —compression)
  - Stochastic block models (bayesian inference)

- These methods have a clear definition of what are good communities. Theoretically grounded

INFOMAP

INFOMAP

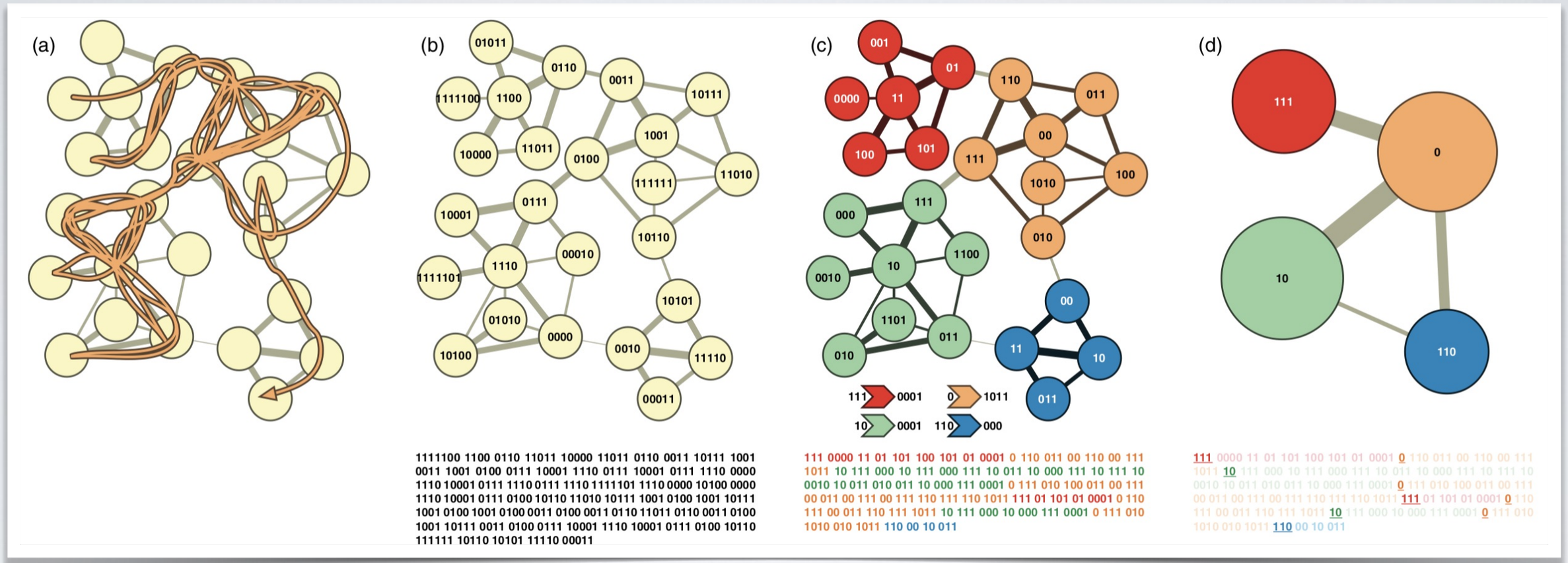
- [Rosvall & Bergstrom 2009]

- Find the partition minimizing the *description* of any *random walk* on the network

- We want to *compress* the description of random walks

Rosvall, Martin, and Carl T. Bergstrom. "Maps of random walks on complex networks reveal community structure." *Proceedings of the National Academy of Sciences* 105.4 (2008): 1118-1123.

INFOMAP



Random walk

Description Without Communities

With communities

Huffman coding: short codes for frequent items

Prefix free: no code is a prefix of another one (avoid fix length/separators)

The Infomap method

Finding the optimal partition M:

- Shannon's entropy

$$L(\mathcal{P}) = H(\mathcal{P}) \equiv -\sum_i p_i \log p_i$$

- Minimise the expected description length of the random walk

Sum of Shannon entropies of multiple codebooks weighted by the rate of usage

probability of between modules movements of a RW, i.e. the rate of usage of the index codebook

probability of within modules movements of a RW, i.e. the rate of usage of the module codebook

$$L(\mathbf{M}) = q_{\curvearrowright} H(\mathcal{Q}) + \sum_{i=1}^{m} p^i_{\circlearrowright} H(\mathcal{P}^i)$$

Expected decryption length of partition M

Entropy of movement between modules, i.e. the frequency weighted average length of codewords

Entropy of movement inside modules, i.e. the frequency weighted average length of codewords in the module codebook

Rosvall, Martin, and Carl T. Bergstrom. "Maps of random walks on complex networks reveal community structure." *Proceedings of the National Academy of Sciences* 105.4 (2008): 1118-1123.

INFOMAP

- To sum up:
  - Infomap defines a *quality function* for a partition different than modularity
  - Any algorithm can be used to optimize it (like Modularity)

- Advantage:
  - Infomap can recognize random networks (no communities)
    - Impossible to compress random noise

STOCHASTIC BLOCK MODELS

STOCHASTIC BLOCK MODELS

- Stochastic Block Models (SBM) are based on statistical generative models of networks

- They are in fact more general than usual communities.

- The model is:
  - Each node belongs to 1 and only 1 community
    - (Relaxed versions exist)
  - To each pair of communities, there is an associated density (probability of each edge to exist)

Stochastic block models

Stochastic Block Models (SBM)

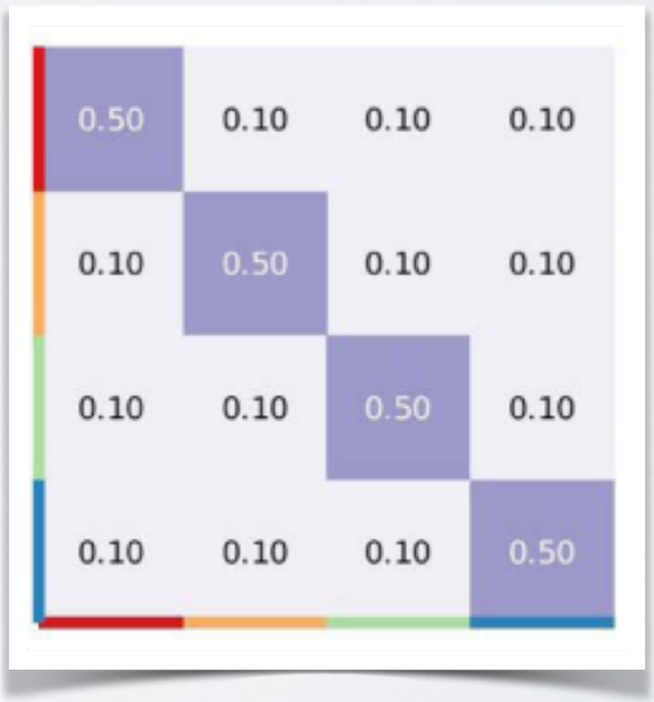
A stochastic block model is a random graph model defined by:

b	$n \times 1$ vector such as b_i describes the index of the block of node i.
E	$k \times k$ **stochastic block matrix**, such as E_{ij} gives the number of edges between blocks i and j (or the probability to observe an edge between any pair of nodes chosen with one node in each of the two blocks).

b



E



Stochastic block models

Stochastic Block Models (SBM)

A stochastic block model is a random graph model defined by:

b	$n \times 1$ vector such as b_i describes the index of the block of node i.
E	$k \times k$ **stochastic block matrix**, such as E_{ij} gives the number of edges between blocks i and j (or the probability to observe an edge between any pair of nodes chosen with one node in each of the two blocks).

Generating networks

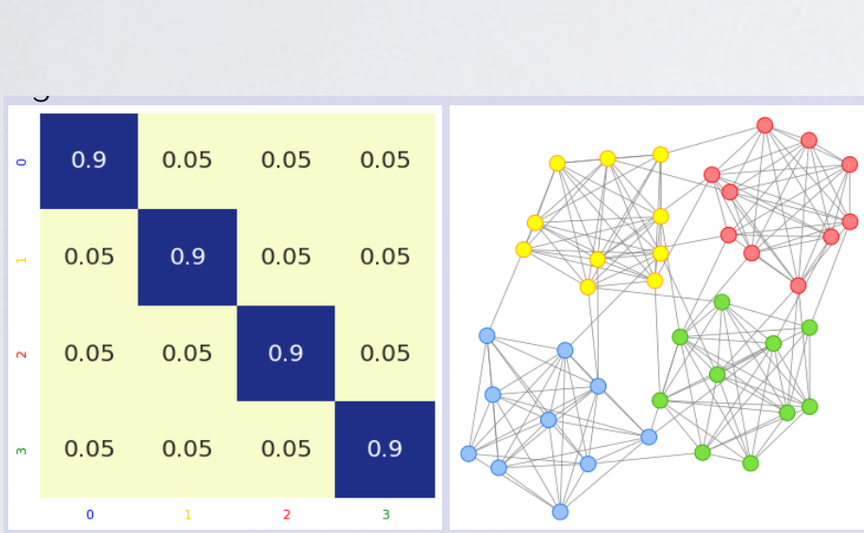
1. Take n disconnected nodes
2. Connect each $u,v \in V$ nodes with probability $E_{b(u),b(v)}$

Properties:

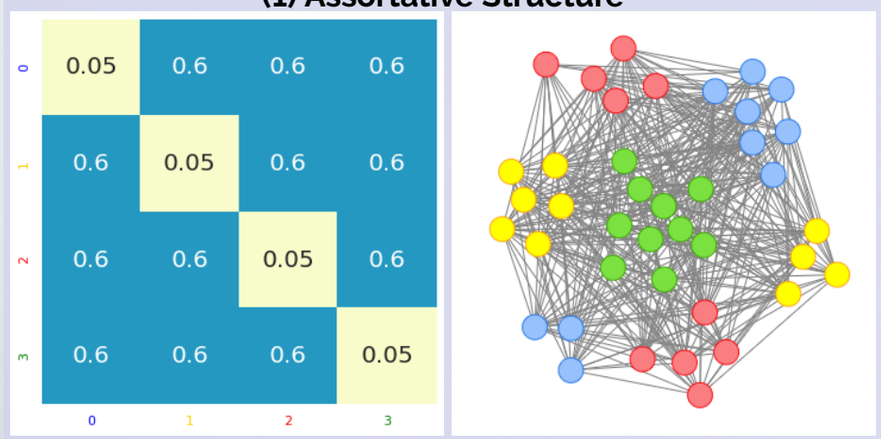
- Every vertices in a same module are statistically equivalent
- Vertices in a module are connected by a random graph
- Emergent degree distribution is a combination of Poisson distributions

STOCHASTIC BLOCK MODELS

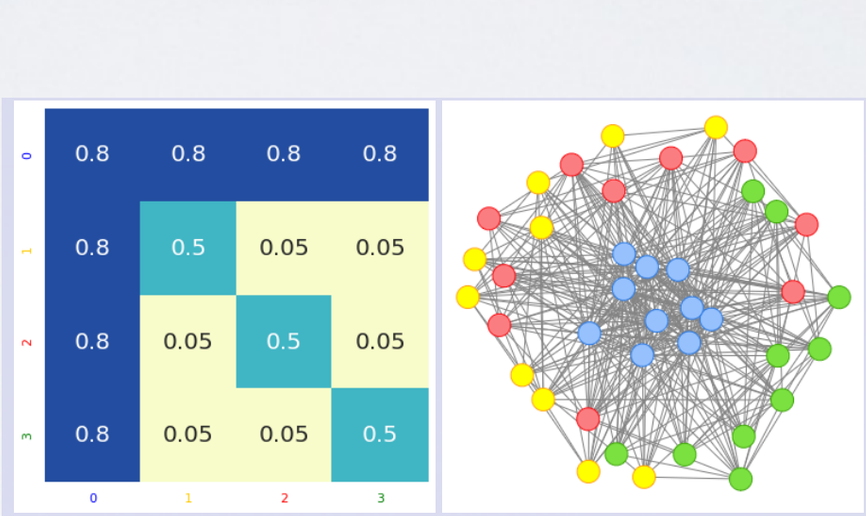
- SBM can represent different things:
  - Associative SBM: density inside nodes of a same communities >> density of pairs belonging to different communities.



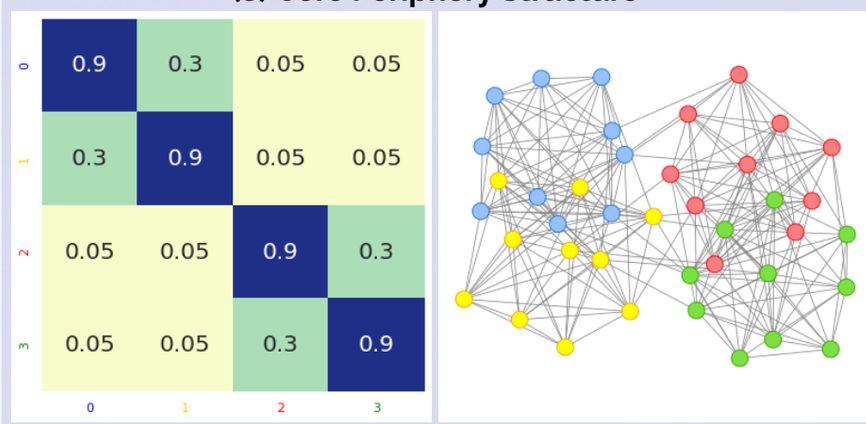
(1) Assortative Structure



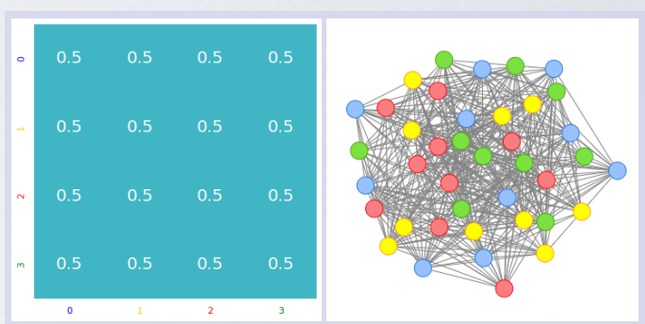
(2) Disassortative Structure



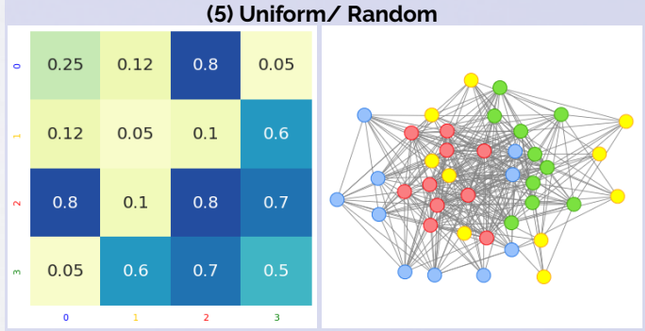
(3) Core Periphery Structure



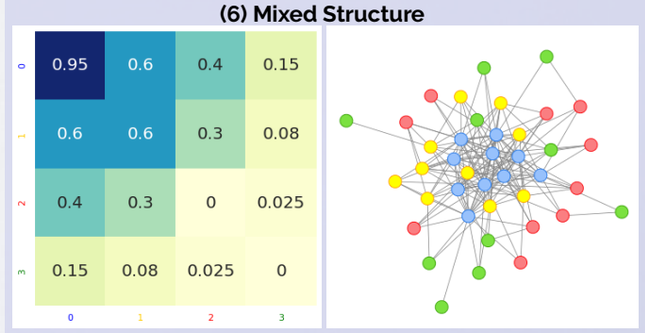
(4) Hierarchical Structure



(5) Uniform/ Random



(6) Mixed Structure



(7) Nested Structure

STOCHASTIC BLOCK MODELS

- SBM can represent different things
- This is very powerful and potentially relevant
- Problem: Often hard to interpret in real situations.
  - SBM can be "constrained":
    - we impose that intra d.>inter d.
    - We impose all Inter d/intra d to be homogeneous

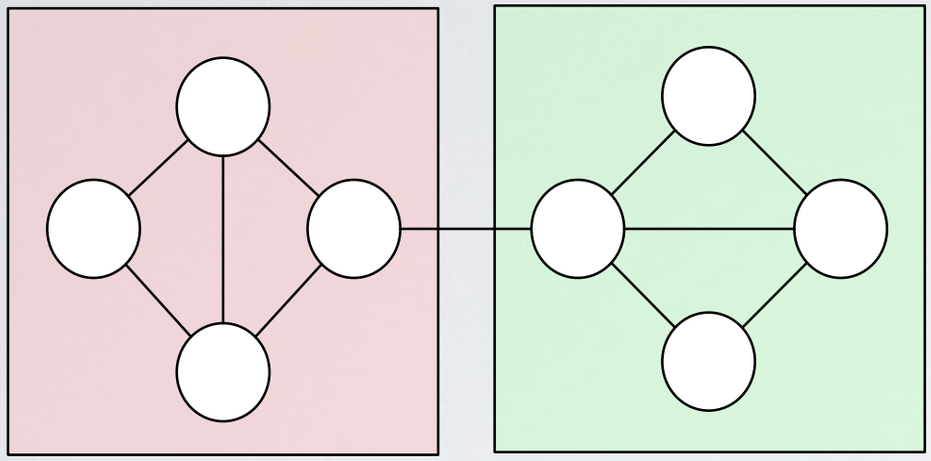
SBM INFERENCE

- Intuition: We want to discover the **parameters** of the model that **maximize the likelihood** of generating the observed graph
  - $\hat{b} = \text{argmax}_b \, p(A \mid b, E)$

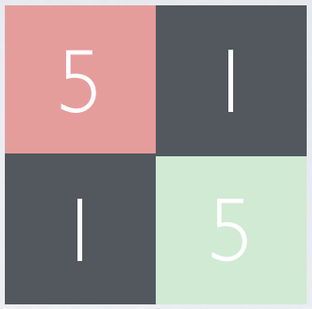
- Can be equivalently formulated using information theory as an information compression problem
  - Minimize the number of **bits** required to **encode the graph**, **given the model**
  - $S = -\, log_2 P(A \mid b, E)$

QUESTIONS ?

INTUITION



$E =$



Microcanonical version

(counting possible graphs/configurations)

block c can generate x possible simple graphs.

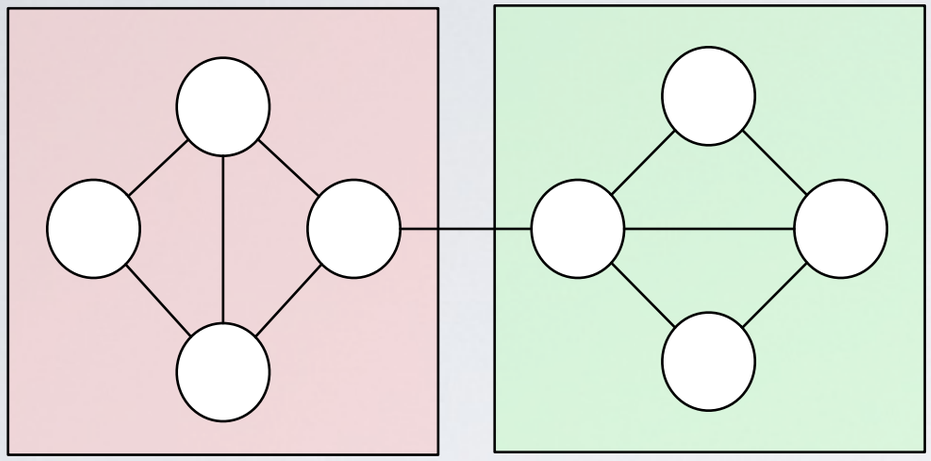
Whatever the graph, its encoding cost is thus $\log_2(x)$

Nb. possible configurations red ?

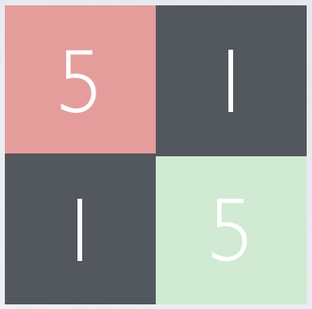
Nb. possible configurations green ?

Nb. possible configurations inter ?

INTUITION



$E =$



$\text{RED} = \log_2(6)$ 6 different configurations for 5 nodes among 6 pairs

$\text{GREEN} = \log_2(6)$

$\text{INTER} = \log_2(4 \times 4)$ 16 different configurations for 1 edge with 4 possibilities on each side

Objective: find RED and GREEN
To minimize this encoding cost

SBM INFERENCE

- Problem: trivial solution
  - Each node in its own cluster
    - $E = A$
    - Probability of 1 to generate the observed network

- Classic solution: fix a number of clusters k

(IN DETAILS)

SBM SIMPLE GRAPHS

- Probability to generate a given graph with a given b, E (likelihood $(A \mid b, E)$)
  - Note that for a given A, b, maximizing E is trivially obtained as the fraction of edges existing between each block pair.

- Assuming a simple graph, Bernouilli distribution
  - Product of probability for each observed edge to be present and for each non existing edge to be not present
  - $$p(A \mid b, E) = \prod_{i<j} \begin{cases} E_{b_i b_j} & \text{if } A_{ij} = 1 \\ 1 - E_{b_i b_j} & \text{if } A_{ij} = 0 \end{cases}$$

SBM POISSON

- Different statistical models can be used
  - The poisson model is a popular choice (simple computation)

- Poisson distribution of edges:
  - Assume possibility of multiple edges between nodes
    - But little difference in practice for sparse graphs

$$P(A \mid b, E) = \prod_{i<j} \frac{(E_{b_i b_j})^{A_{ij}}}{A_{ij}!} e^{-E_{b_i b_j}}$$

    - Since Poisson PMF: $P(k) = \frac{\lambda^k}{k!} e^{-\lambda}$
    - With λ the probability to observe an edge
    - Here I ignore self-loops for simplicity

Karrer, B., & Newman, M. E. (2011). Stochastic blockmodels and community structure in networks. *Physical review E*, *83*(1), 016107.

DC-SBM

- As with modularity, we would like to preserve nodes degrees
  - Else, high-degree nodes tend to end up in a same community, since they are "densely connected" (but expected according to degree)
- $$P(A \mid b, E, \theta) = \prod_{i<j} \frac{(\theta_i \theta_j E_{b_i b_j})^{A_{ij}}}{A_{ij}!} e^{-\theta_i \theta_j E_{b_i b_j}}$$
  - θ is a vector corresponding to nodes degree
  - As E, optimal θ can be deducted from b, A:
    - $\hat{\theta}_i = \frac{k_i}{\kappa_{b_i}}$ with κ_{b_i} the sum of degrees in i's cluster

Karrer, B., & Newman, M. E. (2011). Stochastic blockmodels and community structure in networks. *Physical review E*, *83*(1), 016107.

SBM INFERENCE

- Likelihood maximization:
  - $\hat{b} = \text{argmax}_b \, p(A \,|\, b, E, \theta)$

- Leads to quality functions simple to optimise.
  - Poisson, with self-loops, log and without unnecessary constant:
    - $\mathscr{L}(A \,|\, b) = \sum_{rs} m_{rs} \log \frac{m_{rs}}{n_r n_s}$
  - Same, degree corrected
    - $\mathscr{L}(A \,|\, b) = \sum_{rs} m_{rs} \log \frac{m_{rs}}{\kappa_r \kappa_s}$
  - With
    - m_{rs} : number of edges between blocks r, s
    - n_r : number of nodes in block r
    - κ_r: sum of degrees on block r

Karrer, B., & Newman, M. E. (2011). Stochastic blockmodels and community structure in networks. *Physical review E*, *83*(1), 016107.

MDL SBM

- [2016 Peixoto]
  - Non-parametric SBM
  - Bayesian inference
  - Minimum Description Length (MDL) (Occam's razor)



skewed.de
Modularity maximization considered harmful
Tiago Peixoto's blog: network science, inference, and other things.

9:34 AM · Jul 13, 2023 · **10.8K** Views

Bayesian Formulation

$$P(A \mid b) = P(A \mid \theta, E, b) \overbrace{P(\theta \mid E, b) P(E \mid b) P(b)}^{\text{Priors}}$$

Peixoto, Tiago P. "Bayesian stochastic blockmodeling." *arXiv preprint arXiv:1705.10225* (2017).

STOCHASTIC BLOCK MODELS

Information Theoretic Formulation

$$P(A, \theta, E, b) = 2^{-\Sigma} \qquad \Sigma = S + L$$

$$S = -\log_2 P(A \mid \theta, E, b)$$ # bits necessary to encode the graph knowing the model

$$L = -\log_2 P(\theta, E, b)$$ # bits necessary to encode the model

Objective = maximize the graph compression.
-Too many communities: over-complexifying the model
-Too few communities: More costly to encode the graph, since the model provides few useful information
Occam's razor

Peixoto, Tiago P. "Bayesian stochastic blockmodeling." *arXiv preprint arXiv:1705.10225* (2017).

STOCHASTIC BLOCK MODELS

- To sum up:
  - SBM have a convincing definition of communities
  - Have a richer expression power
  - Can also say if there is no community
  - And also suffer from a form of resolution limit

- Gain popularity since works by Peixoto.
  - Variants:
    - Overlapping (Mixed membership, Fuzzy)
    - Hierarchical
    - …

EVALUATION OF COMMUNITY STRUCTURE

EVALUATION

- Two main approaches:
  - Intrinsic/Internal evaluation
    - Partition quality function
    - Individual Community quality function
  - Comparison of observed communities and expected communities
    - Synthetic networks with community structure
    - Real networks with Ground Truth

INTRINSIC EVALUATION

INTRINSIC EVALUATION

- Partition quality function
  - Already defined: **Modularity**, **graph compression**, etc.

- Quality function for individual community
  - Internal Clustering Coefficient
  - Conductance: $\frac{|E_{out}|}{|E_{out}| + |E_{in}|}$
    - Fraction of external edges

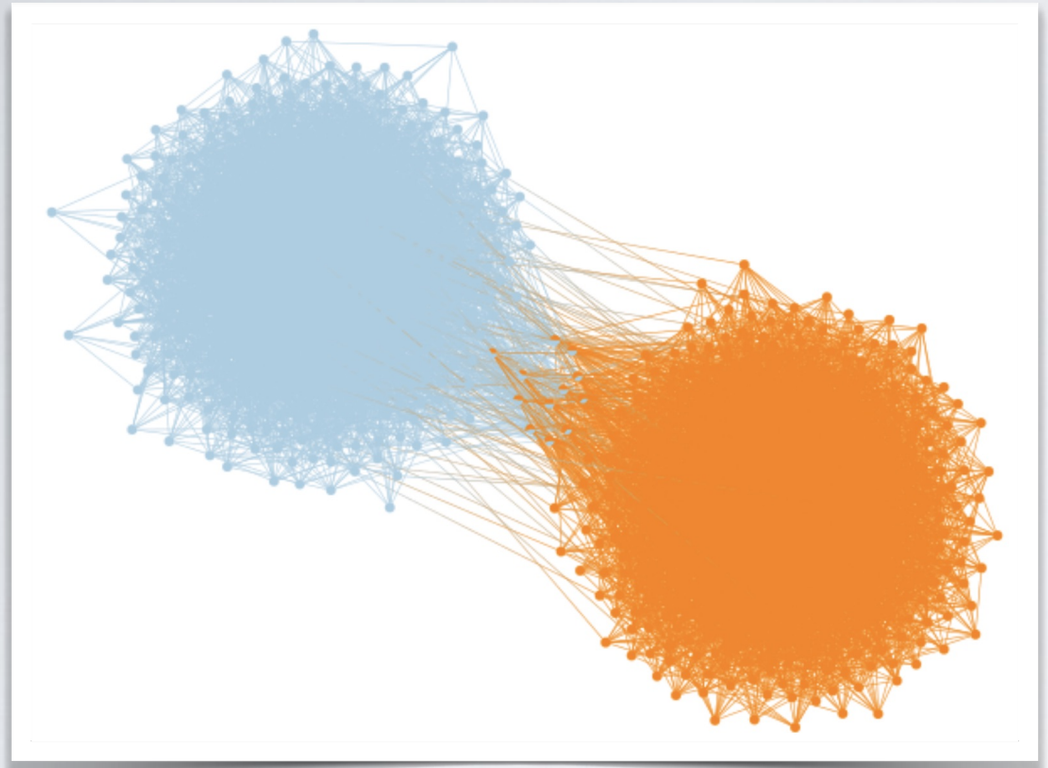
$|E_{in}|, |E_{out}|$:
of links to nodes inside (respectively, outside) the community

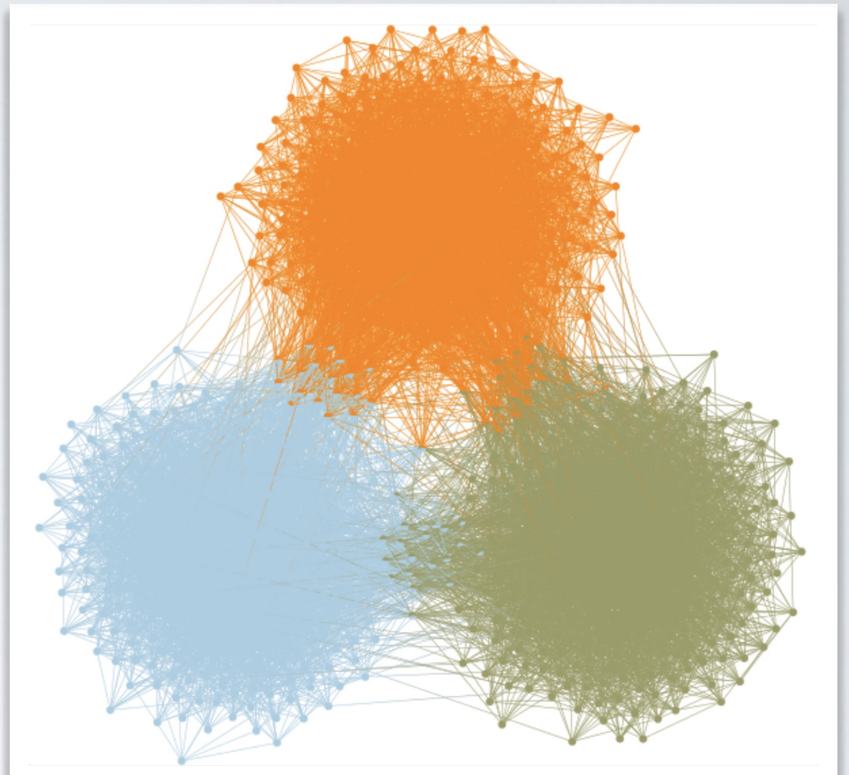
COMPARISON WITH GROUND TRUTH

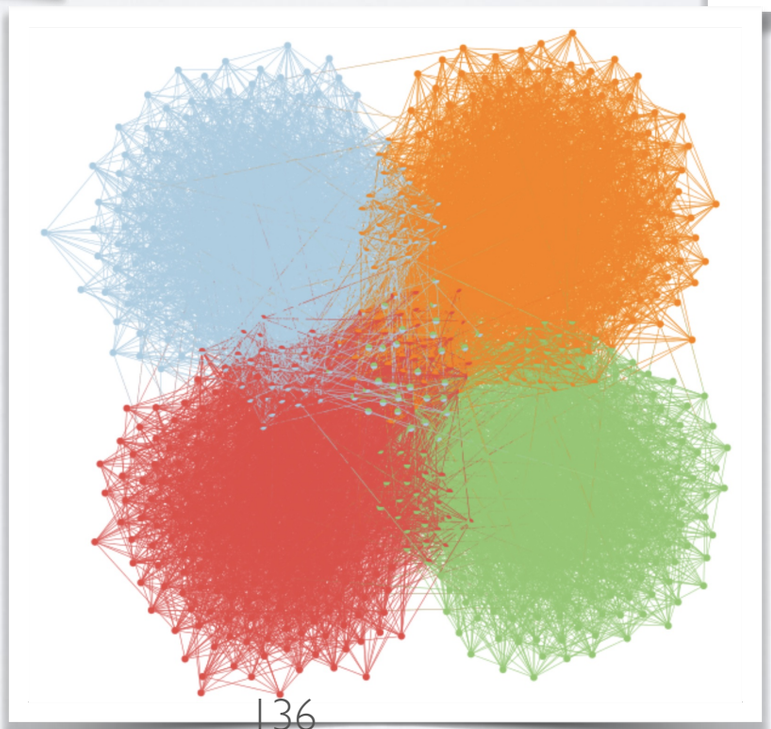
SYNTHETIC NETWORKS

- Planted Partition models:
  - Another name for SBM with manually chosen parameters
    - Assign degrees to nodes
    - Assign nodes to communities
    - Assign density to pairs of communities
    - Attribute randomly edges
  - Problem: how to choose parameters?
    - Either oversimplifying (all nodes same degrees, all communities same #nodes, all intern densities equals…)
    - Or ad-hoc process (sample values from distributions)

SYNTHETIC NETWORKS





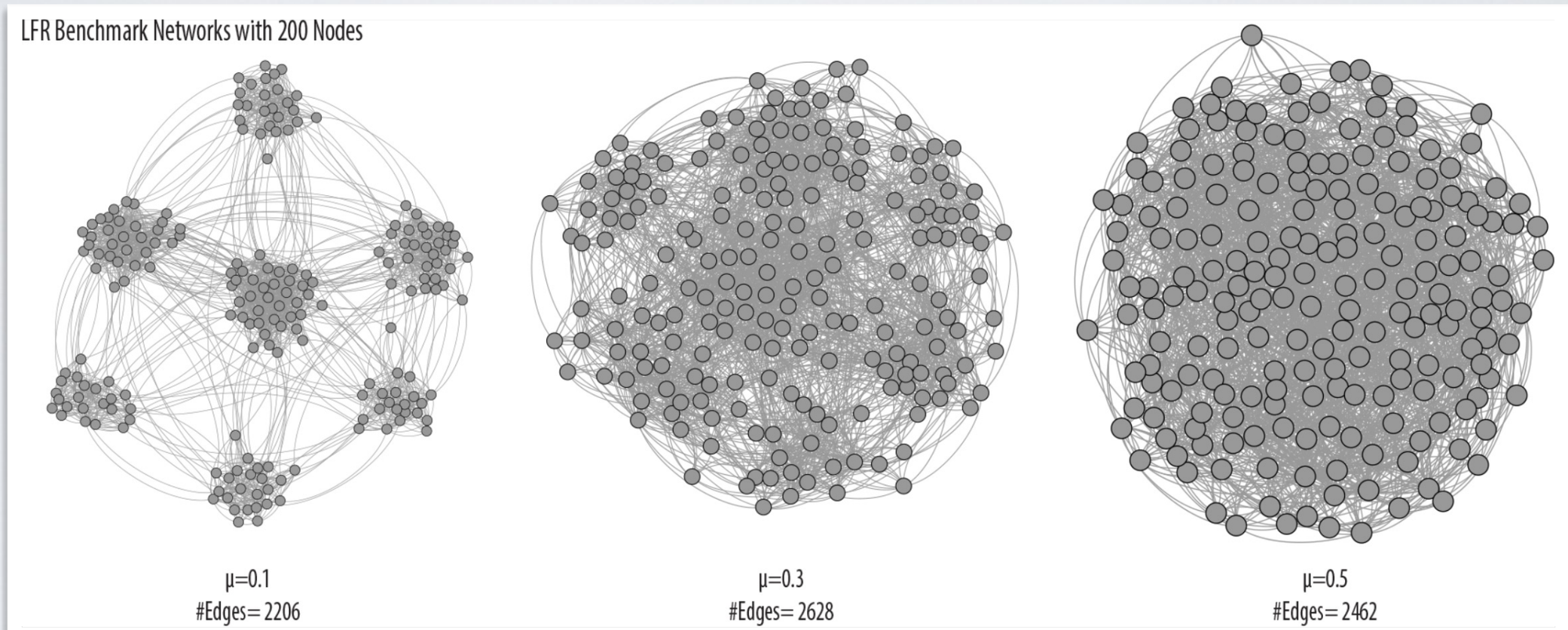


SYNTHETIC NETWORKS

- LFR Benchmark [Lancichinetti 2008]
  - High level parameters:
    - Slope of the power law distribution of degrees/community sizes
    - Avg Degree, Avg community size
    - Mixing parameter: fraction of external edges of each node
  - Varying the mixing parameter makes community more or less well defined

- Currently the most popular

SYNTHETIC NETWORKS



SYNTHETIC NETWORKS

- Pros of synthetic generators:
  - We know for sure the communities we should find
  - We can control finely the parameters to check robustness of methods
    - For instance, resolution limit…

- Cons:
  - Generated networks are not realistic: simpler than real networks
    - LFR: High CC, scale free, but all nodes have the same mixing coefficient, no overlap, …
    - SBM: depend a lot on parameters, random generation might lead to unexpected ground truth (it is *possible* to have a node with no connections to other nodes of its own community…)

REAL NETWORKS WITH GT

- In some networks, **ground truth** communities are known:
  - Social networks, people belong to groups (Facebook, Friendsters, Orkut, students in classes…)
  - Products, belonging to categories (Amazon, music…)
  - Other resources with defined groups (Wikipedia articles, Political groups for vote data…)

- Some websites have collected such datasets, e.g.
  - http://snap.stanford.edu/data/index.html

REAL NETWORKS WITH GT

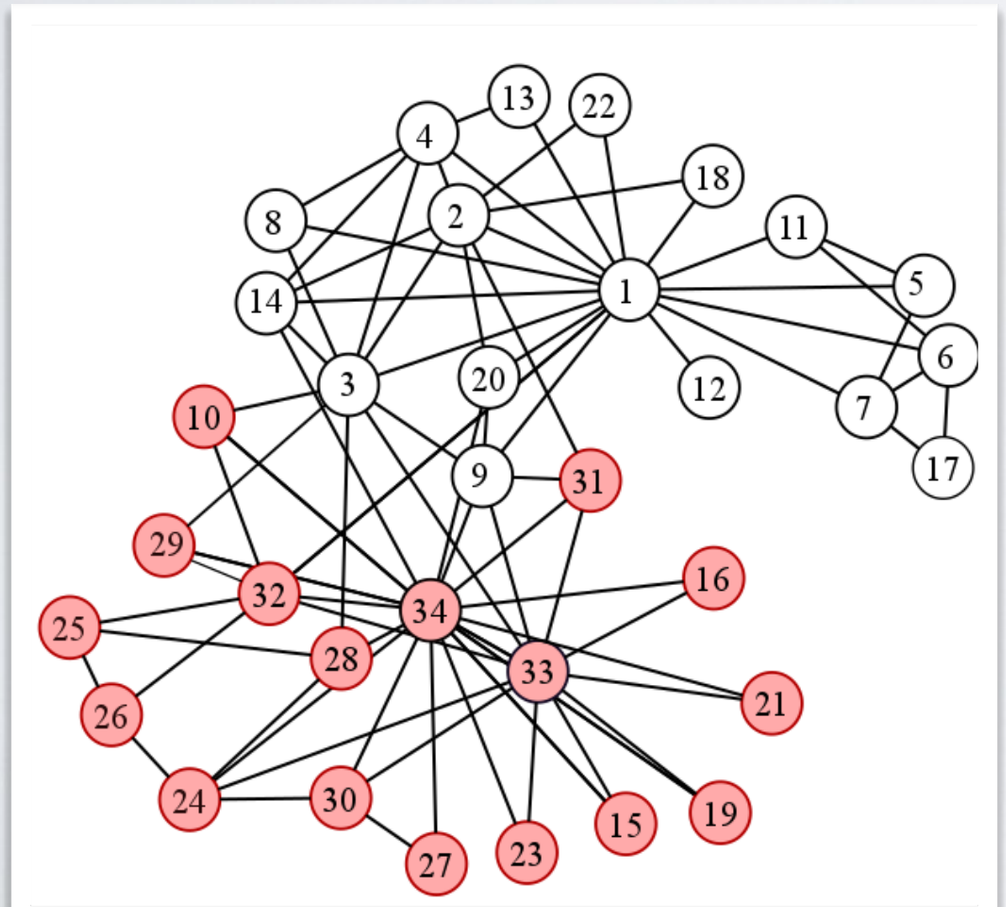
- Pros of GT communities:
  - Retain the full complexity of networks and communities

- Cons:
  - No guarantee that communities are **topological** communities.
  - In fact, they are not: some GT communities are not even a single connected component…

- Currently, controversial topic
  - Some authors say it is non-sense to use them for validation
  - Some others consider it necessary

REAL NETWORKS WITH GT

- Example: the most famous of all networks: Zackary Karate Club



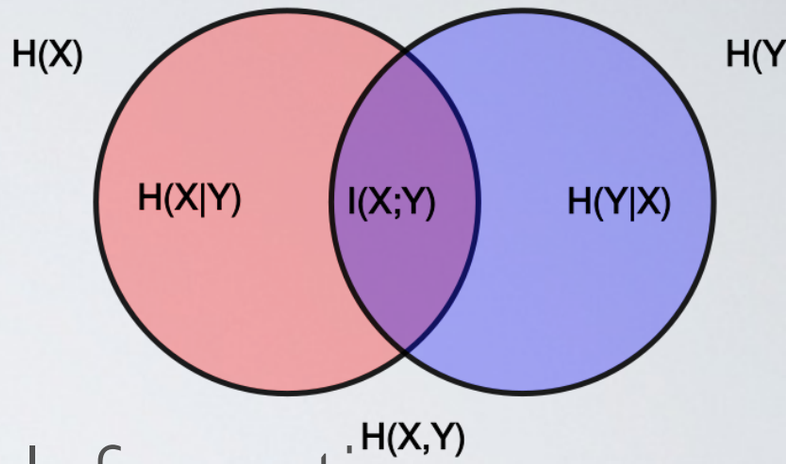
If your algorithm find the right communities,
Then it is wrong…

MEASURING PARTITION SIMILARITIES

- Synthetic or GT, we get:
  - Reference communities
  - Communities found by algorithms

- How to measure their similarity?
  - Same as for the clustering task in unsupervised machine learning
  - NMI => AMI
  - ARI
  - …

MEASURING PARTITION SIMILARITIES



- NMI: Normalized Mutual Information
- Classic notion of Information Theory: Mutual Information
  - How much knowing one variable reduces uncertainty about the other
  - Or how much in common between the two variables

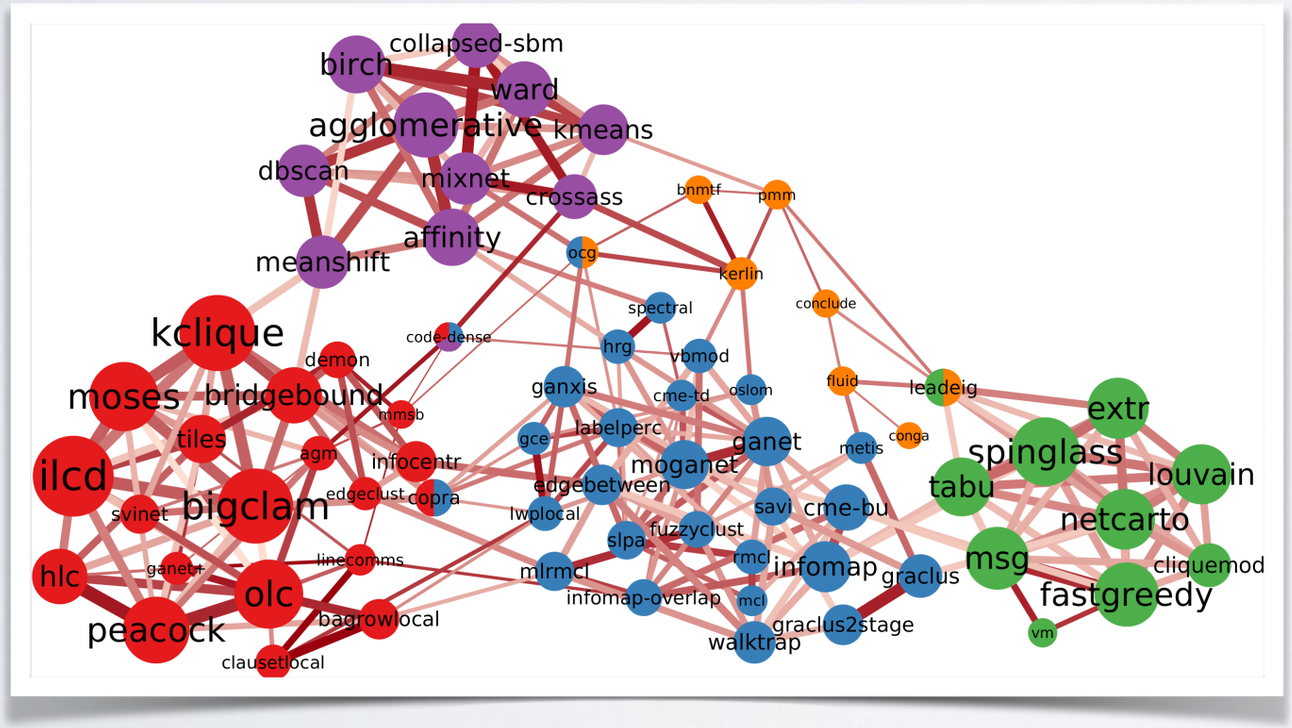
$$I(X;Y) = \sum_{y \in Y} \sum_{x \in X} p(x,y) \log \left(\frac{p(x,y)}{p(x)\,p(y)} \right)$$

- Normalized version: NMI
  - 0: independent, 1: identical
- Adjusted for chance: aNMI

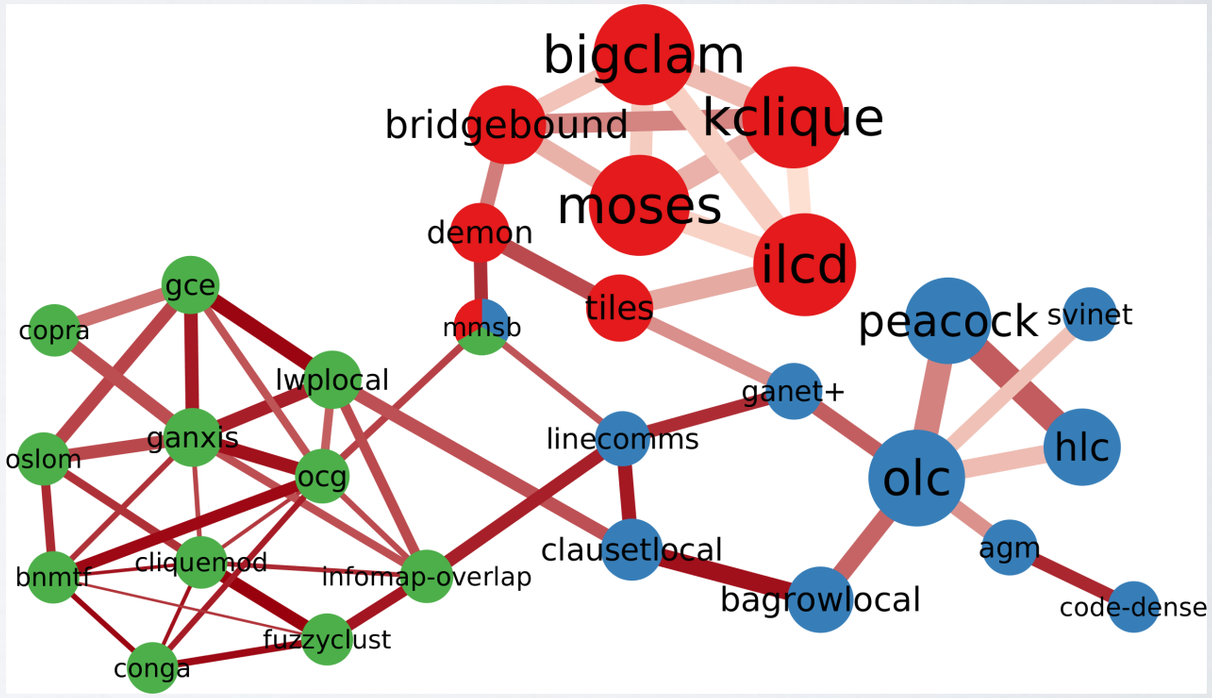
$$AMI(U,V) = \frac{MI(U,V) - E\{MI(U,V)\}}{\max\{H(U), H(V)\} - E\{MI(U,V)\}}$$

ALGORITHMS COMPARATIVE ANALYSIS

Rank	Algorithm	oNMI MAX
1	linecomms	165
2	oslom	73
3	infomap-overlap	64
4	savi	62
5	labelperc	57
6	rmcl	54
7	edgebetween	41
7	leadeig	41
7	vbmod	41
10	gce	32



All methods



Overlapping only

Coscia, Michele. "Discovering Communities of Community Discovery." *arXiv preprint arXiv:1907.02277* (2019).

OTHER MESO-SCALE ORGANIZATIONS

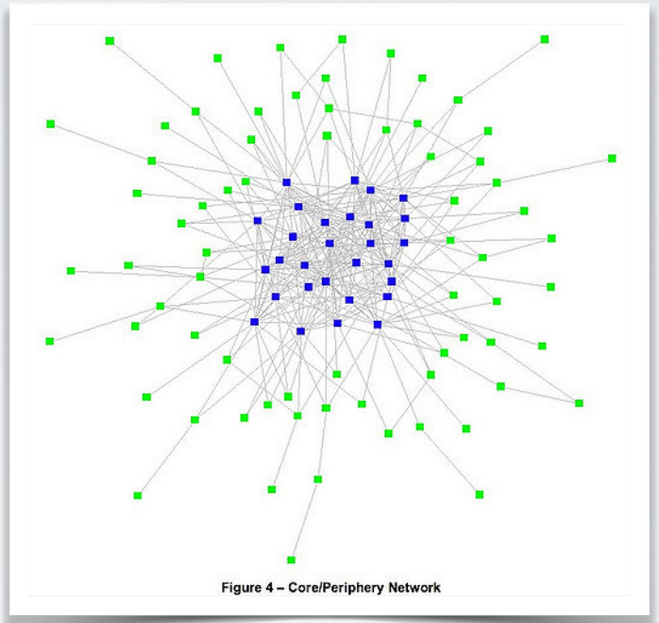
MESO-SCALE

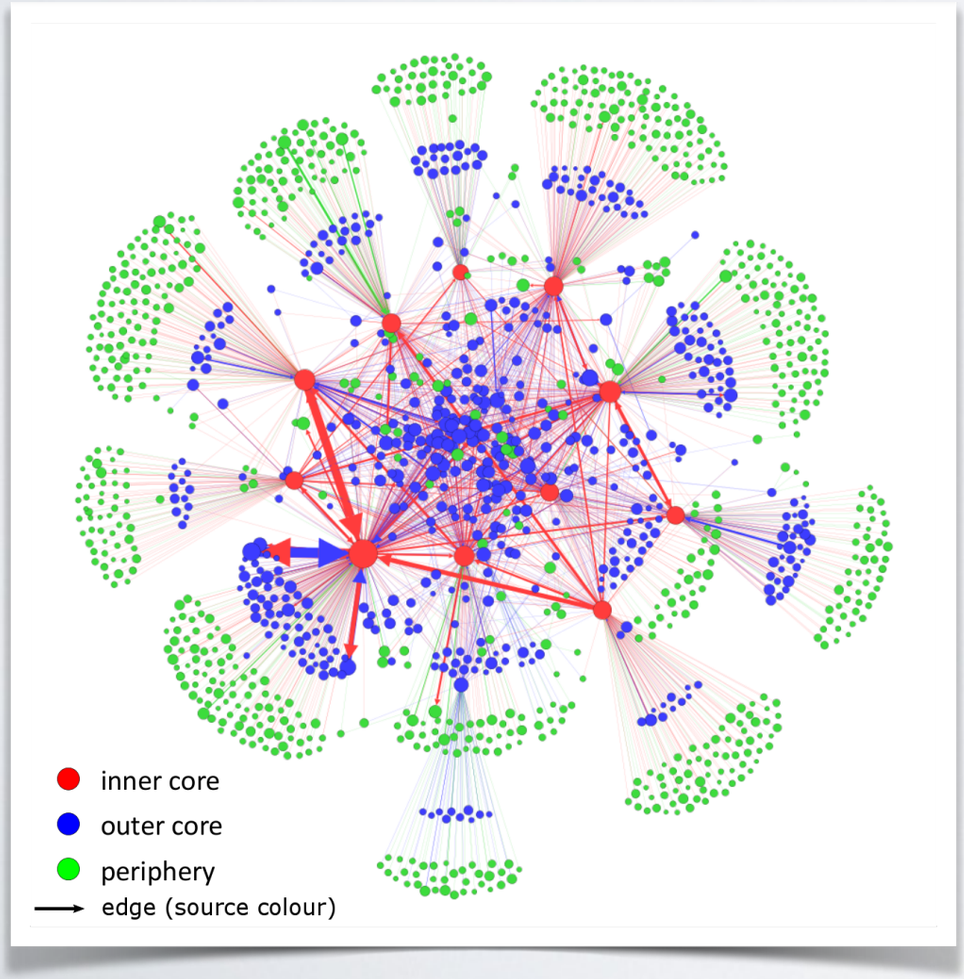
- MACRO properties of networks:
  - degree distribution, density, average shortest path…

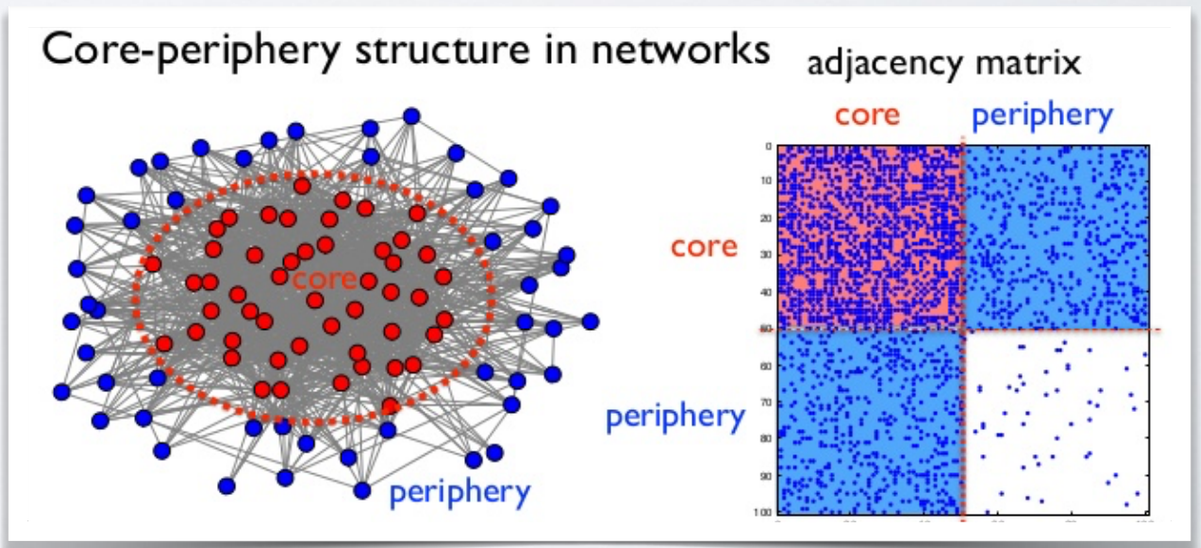
- MICRO properties of networks:
  - Centralities

- MESO-scale: what is in-between
  - Community structure
  - Overlapping Community Structure
  - Core-Periphery
  - Spatial Organization

CORE-PERIPHERY







OVERLAPPING COMMUNITIES

- In real networks, communities are often overlapping
  - Some of your High-School friends might be also University Friends
  - A colleague might be a member of your family
  - …
- Overlapping community detection is considered much harder
  - And is not well defined

OVERLAPPING COMMUNITIES

- Many algorithms
  - Adaptations of modularity, random walks, label propagations, SBM…
  - Original methods
  - Many local methods (local criterium), unlike global optimization for non-overlapping methods.

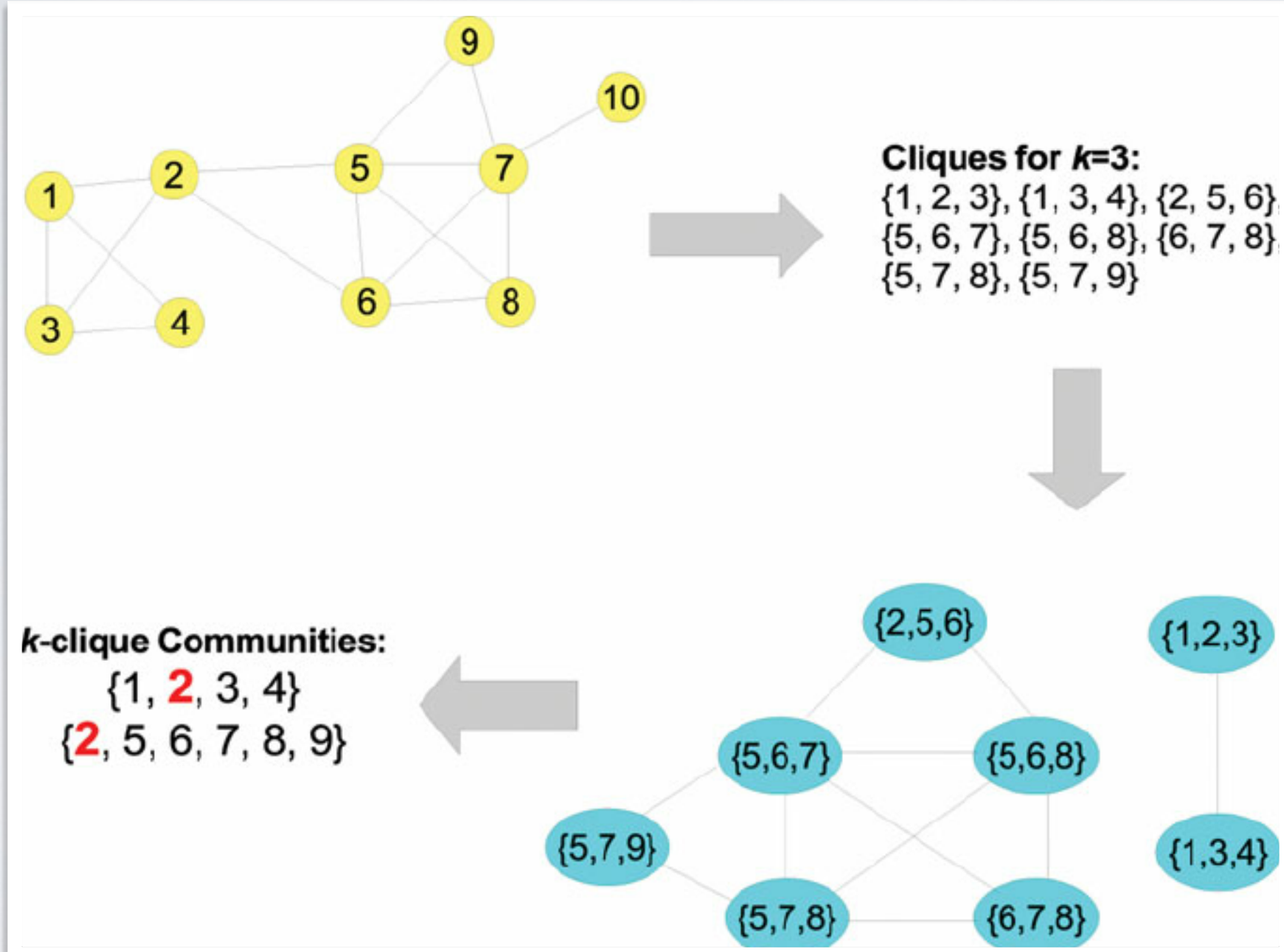
OVERLAPPING COMMUNITIES

- Motif-based definitions:
  - Cliques
    - Of a given size
    - Maximal cliques
  - N-cliques
    - Set of nodes such as there is at least a path of length <=N between them
    - Generalization of cliques for N>1
    - Computationally expensive

K-CLIQUE PERCOLATION

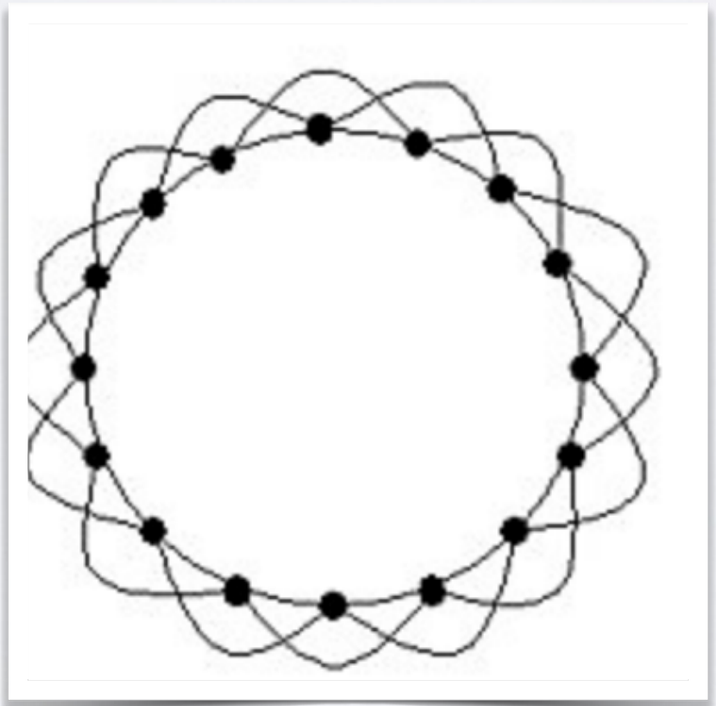
- (Other name: CPM, C-finder)
- Parameter: size k of atomic cliques
- 1)Find all cliques of size k
- 2)merge iteratively all cliques having k-1 nodes in common

K-CLIQUE PERCOLATION



K-CLIQUE PERCOLATION

- Weakness: communities can be very far from random networks



HIERARCHICAL COMMUNITIES

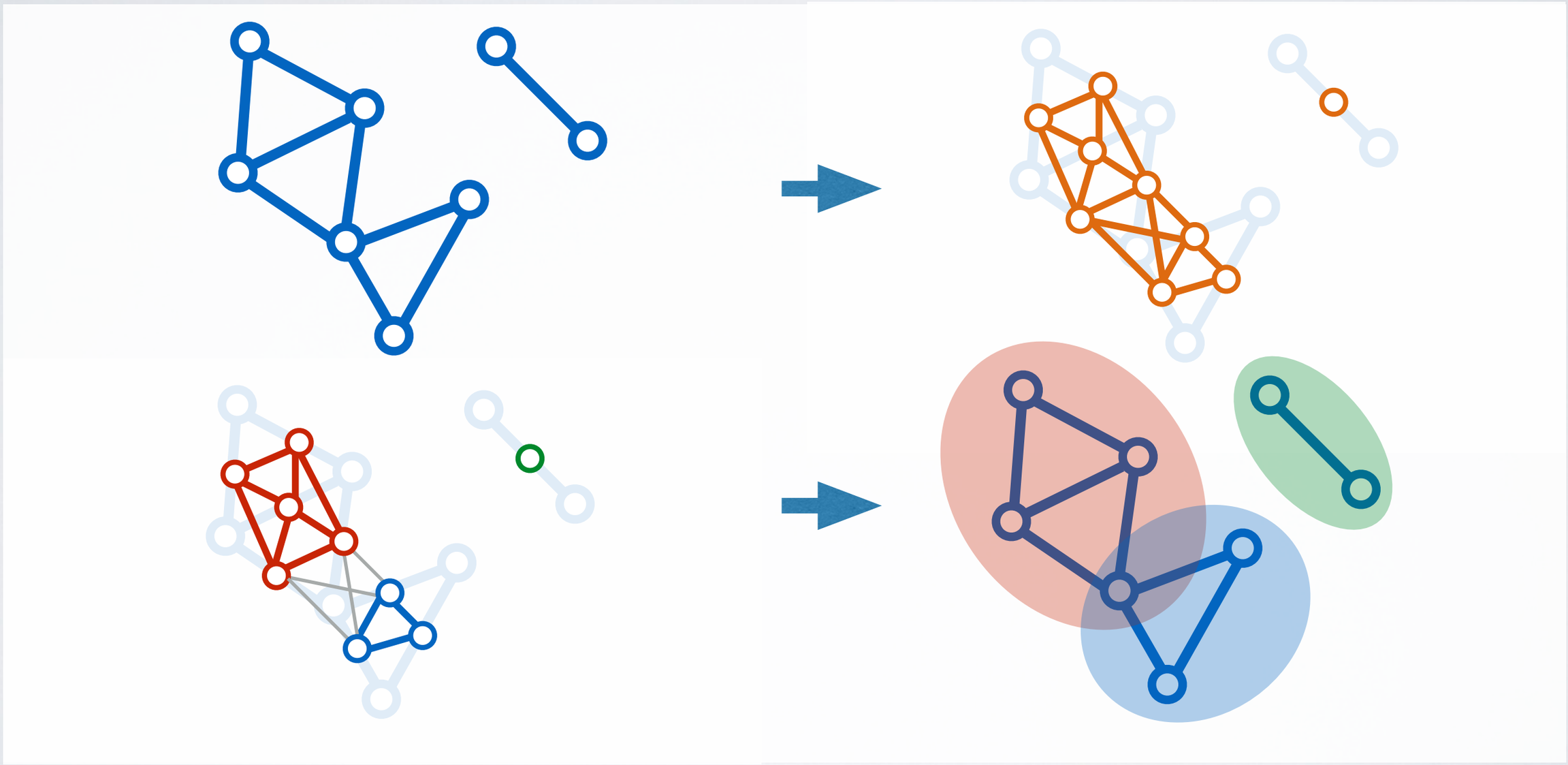


Lancichinetti, Andrea, et al. "Finding statistically significant communities in networks." *PloS one* 6.4 (2011): e18961.

Link clustering - overlapping communities

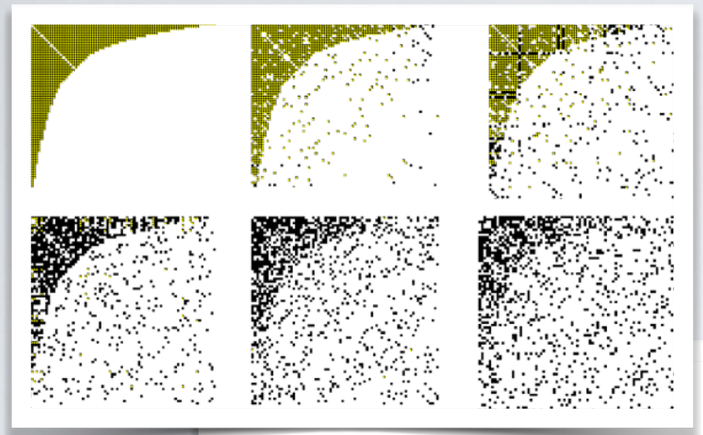
Link graphs

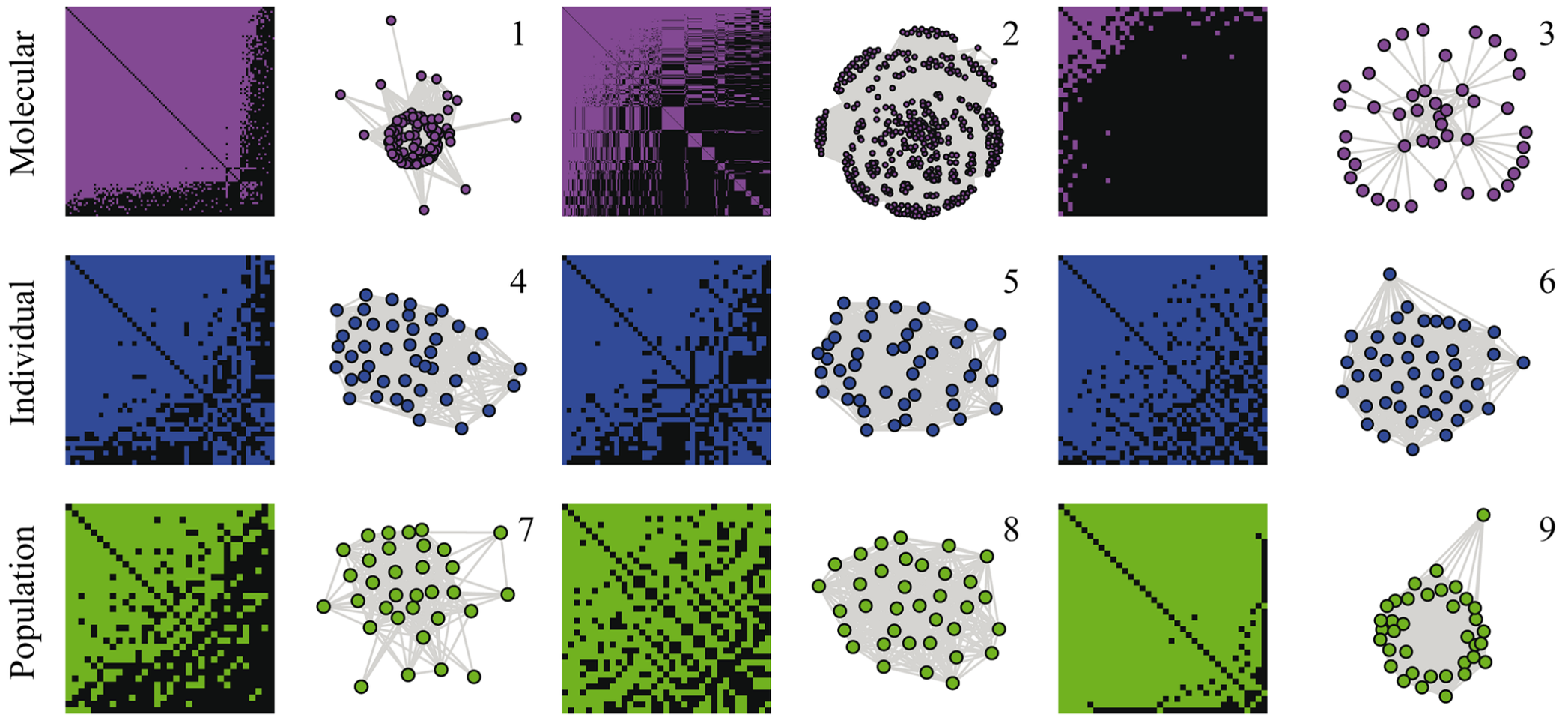
- Links are replaced by nodes which are connected if the original links share a node



- Community detection on link graphs allows for overlapping communities

NESTEDNESS





LATENT POSITIONS

- Random Dot Product Graphs
  - Similar approach than SBM
  - But nodes have a position in a latent space of a certain nb of dimensions
    - Edge probability depends on the dot product between latent positions
  - Likelihood maximization can be used for inference

- Graph embedding
  - Shallow neural networks (DeepWalk, Node2Vec)
  - Deep Neural Networks (GNN, Variational Graph Auto Encoders etc.)

ERGM

- Exponential Random Graph Models
  - Constraint on local patterns and properties
    - Triangle counts
    - Mutual edges in directed graphes
    - Homophily
    - …
- Estimated by likelihood maximization
  - But very costly, limited to small graphs
  - More used in social sciences on manually collected datasets