Random networks

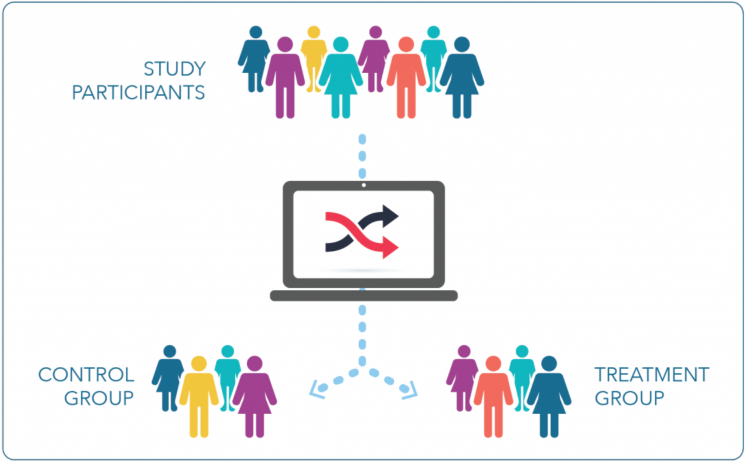
M. Mitrović Dankulov and A. Alorić

December 28, 2022

Outline

- Randomized experiments allow researchers to scientifically measure the impact of an intervention on a particular outcome of interest
- Clinical trials, biological experiments, social science experiments - any type of empirical measurment that does not allow controlled experiment
- Purpose - prevention of selection bias and insurance against the accidental bias

Clinical trials



Source: https://mrctcenter.org/clinical-research-glossary/glossary-words/randomization/

What if we cannot make an experiment?

- Problem - we have measured some effect in a system; we cannot make a controlled experiment nor sample several groups and measure the effect on several different subsets

- We use base model to test our hypothesis

- Base or null model contains some assumptions about the system and is used as a representative of a general population

Example: fair coin

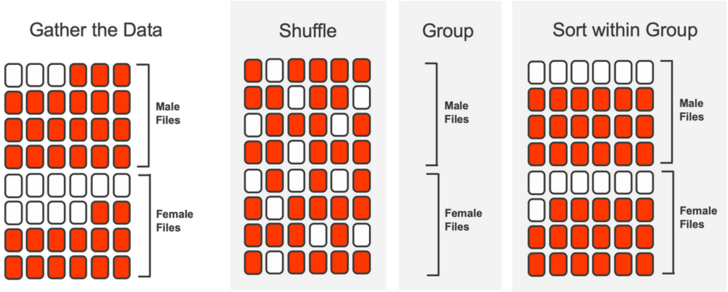
- We are trowing a coin and get: $\{T, T, T, H, T, H, T, T, T, H\}$
- Two hypothesis:
    - Ho: Our coin is fair - 50% : 50% is a chance to get H or T
    - H1: Our coin is biased in favour of tails with $p = 0.7$
- A probability to get a certain set of H and T is:
  $P = \binom{N}{N_T} p_T^{N_T} (1 - p_T)^{N_H}$

Example: fair coin

- We need to calculate a probability to get 7 or more tails under the assumption that our coin is fair:
  $P = \binom{10}{7} 0.5^{N_T=7}(1-0.5)^{N_H=3} + \binom{10}{8} 0.5^{N_T=8}(1-0.5)^{N_H=2} + \binom{10}{9} 0.5^{N_T=9}(1-0.5)^{N_H=1} + \binom{10}{10} 0.5^{N_T=10}(1-0.5)^{N_H=0} = 0.171875$
- Our p-value is approximately 0.17. Is it small enough to reject Ho?
- Small p-values in practice: 0.1, 0.05, 0.01
- The only thing we can say here is that our hypothesis that the coin is fair can not be rejected

What if we do not have a null model?

Promotoion and gender



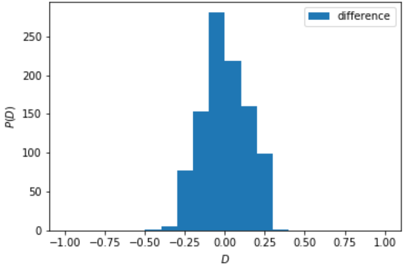
- H0 (Null hypothesis) The variables sex and decision are independent.
- H1 (Alternative hypothesis) The variables sex and decision are not independent

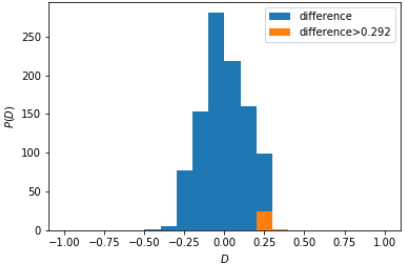
Source: http://shorturl.at/KNP01

Example: promotion

- Difference $\frac{21}{24} - \frac{14}{24} = 0.292$ - estimate of true difference
- Shuffling: we keep 35 labeled *promoted* and 13 labeled as *not promoted* and we randomly assing these into *male* and *female* group
- Shuffling difference: $\frac{18}{24} - \frac{17}{24} = 0.042$
- We will repeat the shuffling procedure S times, calculate the difference, and estimate the distribution of distances

Example: promotion





p-value to have difference 0.292 is 0.021

We reject null hypothesis. Data provide evidence of sex discrimination against female candidates.

Complex networks

- They have rich topological structure
- They have broad degree distribution (power-law); they are small world; they are clustered; they are correlated; they have non-trivial centrality distributions; they have communities; ...
- But which of these properties are relevant and inherent to real world networks?

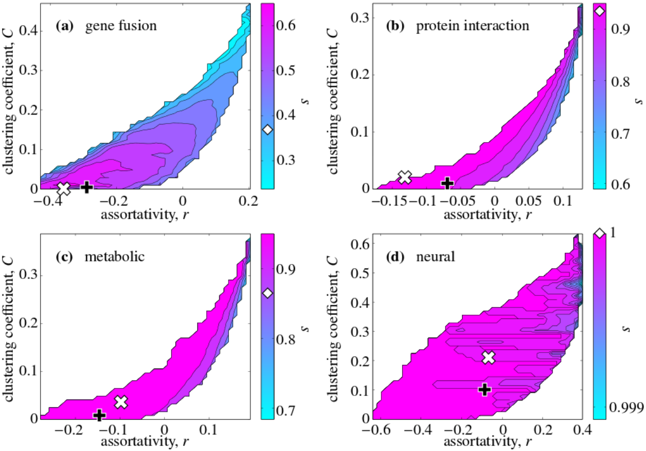
Complex networks

- We will use a null model of complex networks. But which?
- Erdos-Renyi graphs: random graphs with Poisson degree distribution

- Real networks do not have Poisson degree distribution

- If we use as ER model as null model there is a high probability that measured property is *significant*

Network measures are not independent



P. Holme and J. Zhao, PRE 75(4), 046111, (2007)

Our goal

- We measure some property I in the real network

- We want to check if this property is atypical for random networks which have some similar property as real network: number of nodes, number of links, degree sequence, degree-degree correlations, etc.

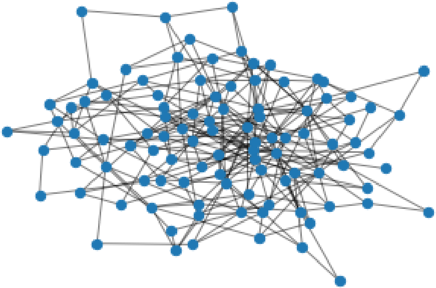
- Recipe: we will use a suitable null model and create S random networks using this model; measure property I in each random network and compare the average with its value in real networks

Network null models

- Models that perserve average degree:
  - on average (soft conditions) - Erdos-Renyi (ER) model with given p
  - strict (hard conditions) - ER model with given L
- Models that perserve degree distribution:
  - on average (soft conditions) - configuration model
  - strict (hard conditions) - configuration model
  - strict (hard conditions) - rewiring model
- Models that pesrserve degree-degree correlations:
  - strict (hard conditions) - rewiring model

Erdos-Renyi networks

Erdos-Renyi networks - N nodes; p - probability to have a link between any pair of nodes;

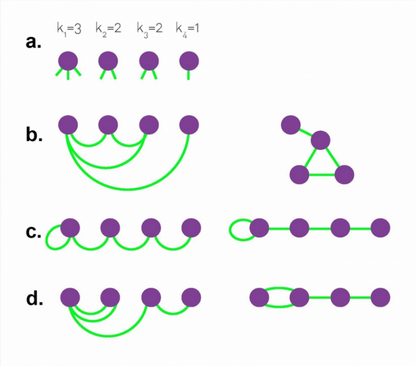


$p = \frac{\langle k \rangle}{N-1} = \frac{2L}{N(N-1)}$; We keep number of nodes and average degree on average

Configuration model: degree sequence

- Given degree sequence $\{k_1, k_2, \ldots, k_N\}$
- We keep:
  - number of nodes N;
  - average degree $\langle k \rangle$
  - degree of a node, on average

Configuration model: degree sequence



- Assign degree to each node
- Randomly select a stub pair and connect them; randomly choose another pair from the remaining $2L-2$ stubs and connect them; repeat until you connect all pairs

Source: Network science book, Secton 4.8

Configuration model: degree sequence

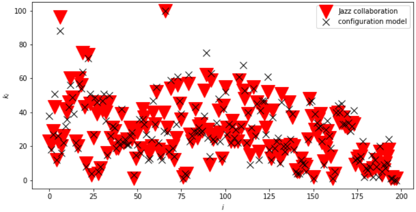
- Described procedure creates networks with fixed degree sequence
- The procedure is ergodic and networks are sampled with equal probability
- Issue: created networks may have self-loops and multiple edges between nodes, while real networks usually do not have this

Configuration model: degree sequence

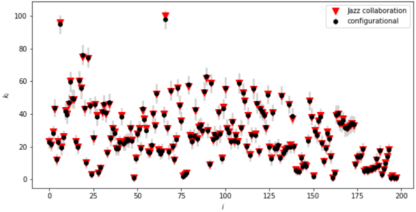
- We want sample of networks that do not have multiple edges and self-loops; strict conditions; hard to generate networks with described procedure
- That is why we will loosen conditions - instead of fixed $\{k_1, \ldots, k_N\}$ we will demand $\{\langle k_1 \rangle, \ldots, \langle k_1 \rangle\}$
- Similar to ER model:
  - We set number of nodes N
  - Connecting probability $p_{ij} = \frac{k_i k_j}{2L-1}$

Configuration model: degree sequence

single network

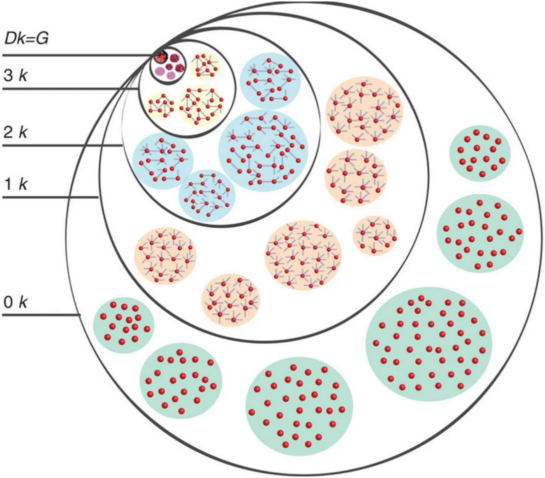


ensemble



Space of all networks

All networks with fixed N and fixed L

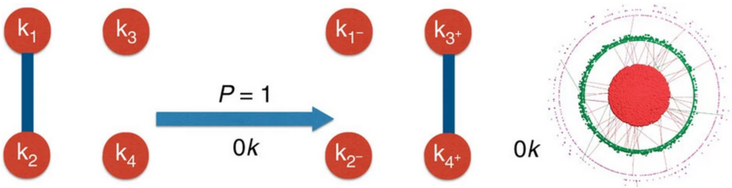


Network rewiring

- We start from the original network and rewire it following some algorithm
- Rewiring means that in each step we destroy n edges between n pair of nodes and create n edges between different n pairs of nodes
- Algorithm depends on the properties we want to perserve

0K random networks

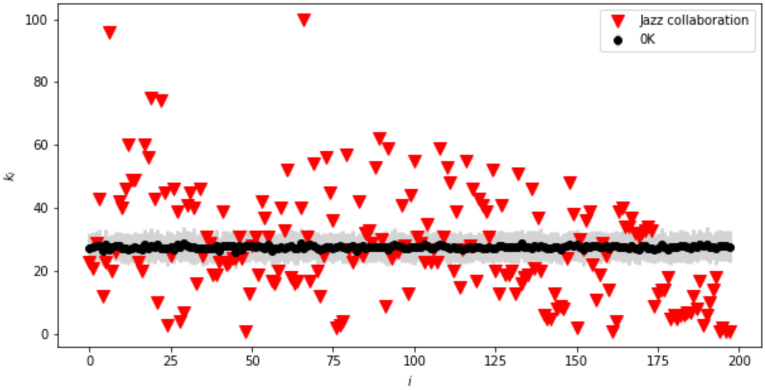
We fix N and L; we keep average degree



Obtained networks are Erdos-Renyi networks with fixed N, L

0K random networks

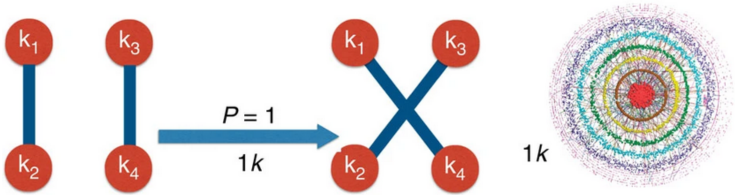
Original: $N = 198$; $L = 2742$; $\langle k \rangle = 27.70$; $\langle c \rangle = 0.62$; $r = 0.020$



0K: $N = 198$; $L = 2742$; $\langle k \rangle = 27.70$; $\langle c \rangle = 0.14$; $r = -0.005$

1K random networks

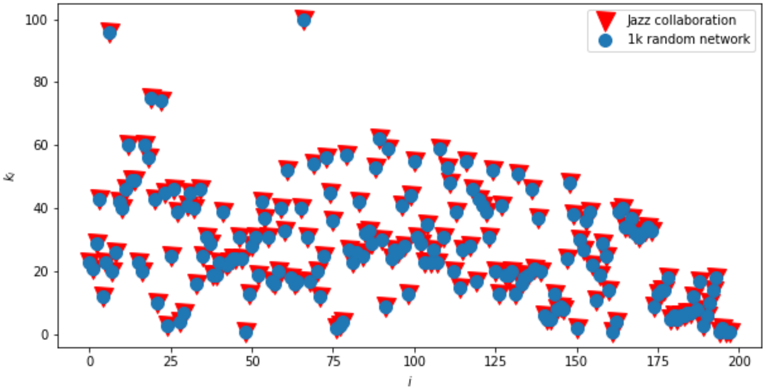
We fix N, L, and degree sequence $\{k_1, k_2, \ldots, k_N\}$



Random networks have the same number of nodes, edges, average degree, degree sequence

1K random networks

Original: $N = 198$; $L = 2742$; $\langle k \rangle = 27.70$; $\langle c \rangle = 0.62$; $r = 0.020$



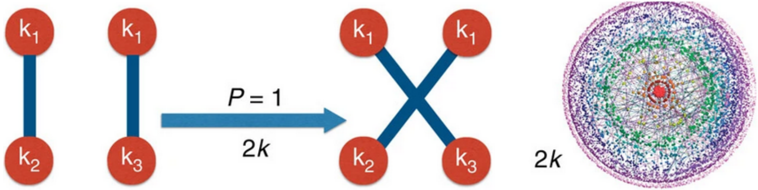
1K: $N = 198$; $L = 2742$; $\langle k \rangle = 27.70$; $\langle c \rangle = 0.27$; $r = -0.076$

Joint degree matrix

- Joint degree matrix D fully capctures the degree-degree correlations in a network
- It is of size $k_{max} \times k_{max}$ and $D_{k_1 k_2}$ is number of edges between nodes with degree k_1 and k_2
- For undirected network is symetric, $D_{k_1 k_2} = D_{k_2 k_1}$
- $\sum_{k=1}^{k_{max}} \sum_{q=k}^{k_max} D_{kq} = L$; $N_k = \frac{1}{k}(D_{kk} + \sum_q D_{kq})$

2K random networks

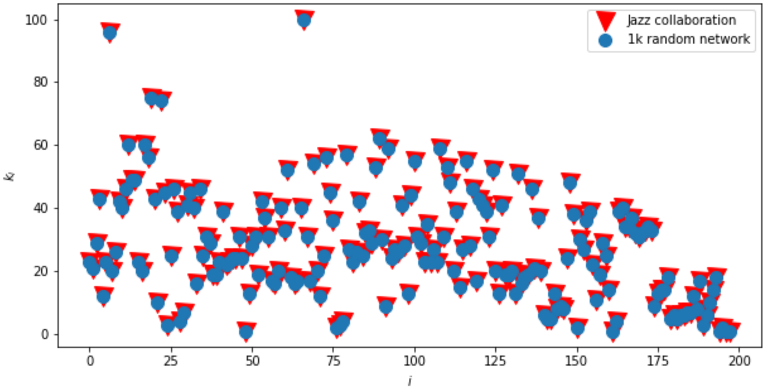
We fix N, L, joint degree mytrix D



Random networks have the same number of nodes, edges, average degree, degree sequence, and joint degree matrix

2K random networks

Original: $N = 198$; $L = 2742$; $\langle k \rangle = 27.70$; $\langle c \rangle = 0.62$; $r = 0.020$



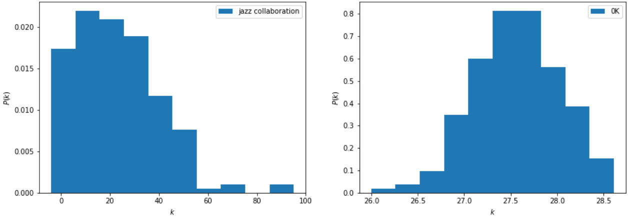
1K: $N = 198$; $L = 2742$; $\langle k \rangle = 27.70$; $\langle c \rangle = 0.26$; $r = -0.020$

Comparison

- For measures such as: average clustering, assortativity index, average degree, average shortest path, we use z-score: $z = \frac{x-\mu}{\sigma}$

- For distributions we compare them qualitatively

Example



Clustering z-score (original vs. 1k): 50.48620578514376

Assortativity z-score (original vs. 1k): 24.720631076626727

Ergodicity and uniformity

- Algorithm is ergodic if you can reach each network in the space of network
- Algorithm is uniform if sample probability for all graphs is equal
- ER, configuration model, 0K, 1K and 2K rewiring are ergodic and uniform