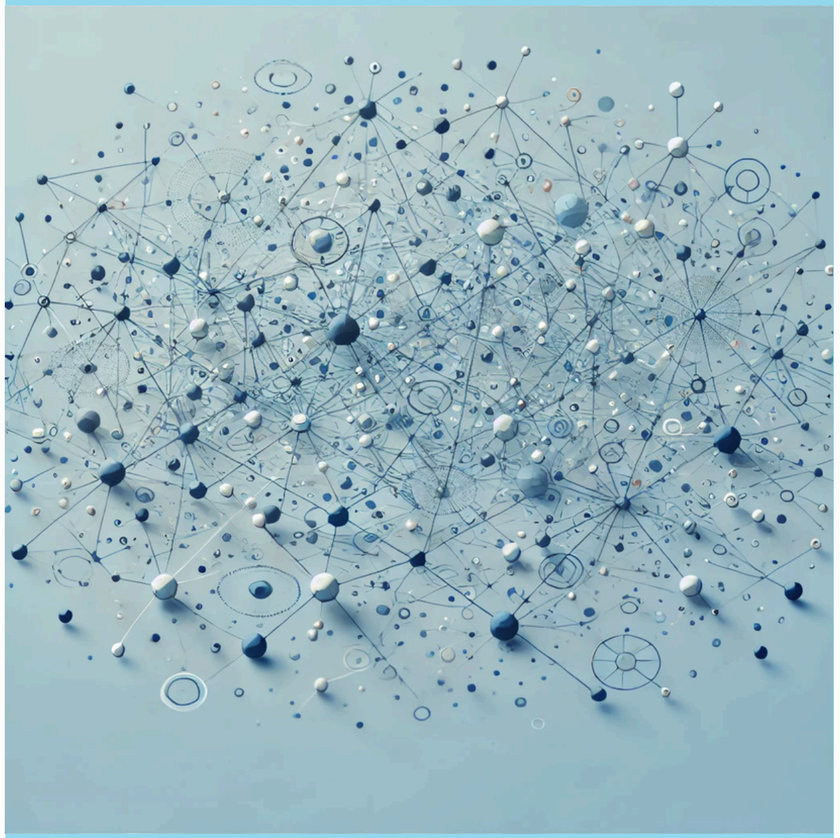


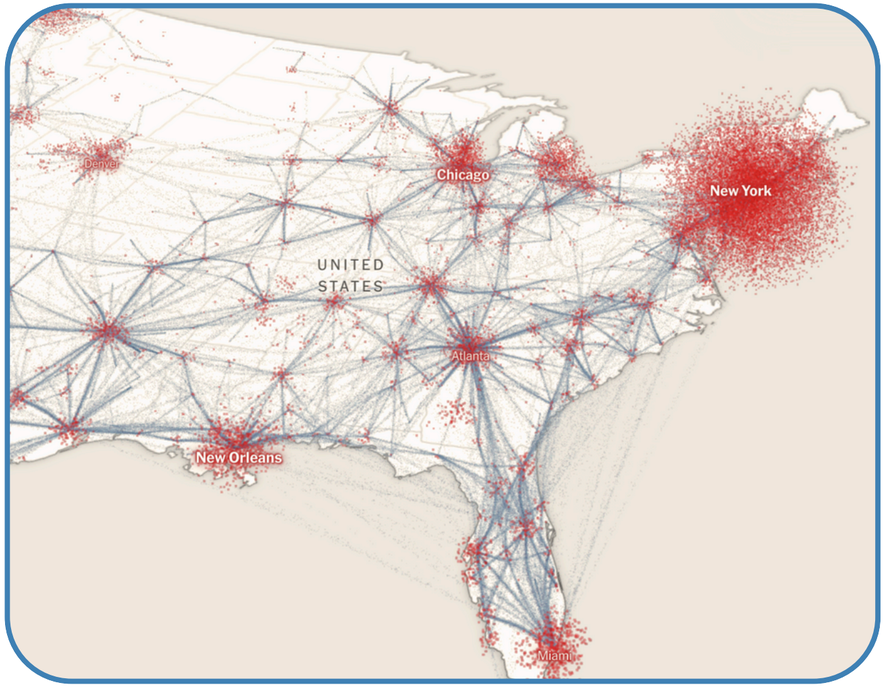
Universität Konstanz

UNIVERSITÄT KONSTANZ

# Network Ensembles

Network Science of Socio-Economic Systems

Giordano De Marzo




# Recap

## Processes on Complex Networks

Many processes take place on a network, in particular spreading processes

## Epidemic Spreading

We introduced he most common epidemic models the SI, the SIS and the SIR model

## Epidemic Spreading on Networks

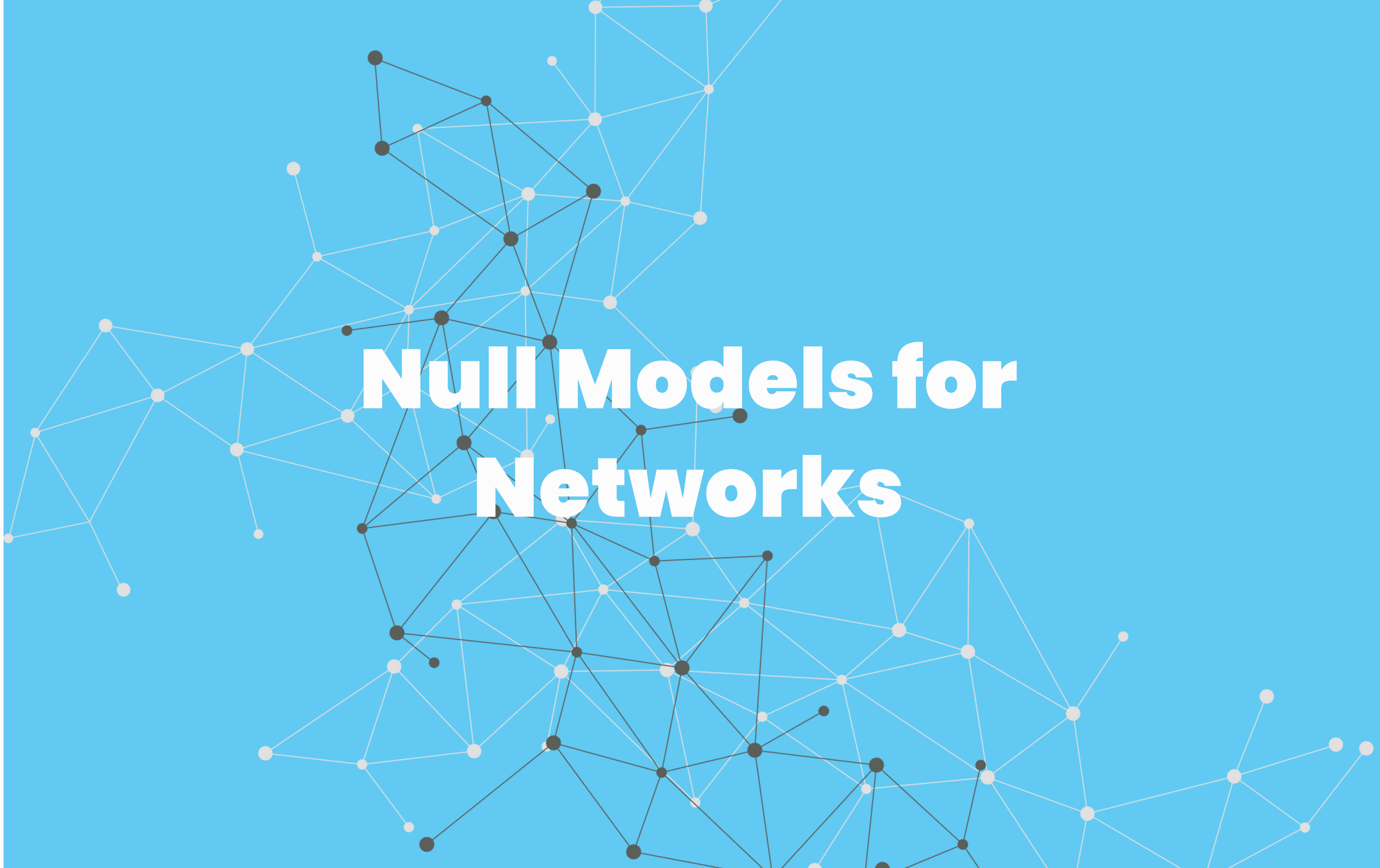
The network topology plays an important role in determining the size of an epidemics

## Complex Contagion

Behaviors spreads following complex contagion processes, where a single exposition is not enough for getting infected

# Outline

1. Null Models for Networks
2. Network Ensembles
3. Applications of Null Models
4. Bipartite Networks Projection



# Null Models for Networks

# What is a Null Model?

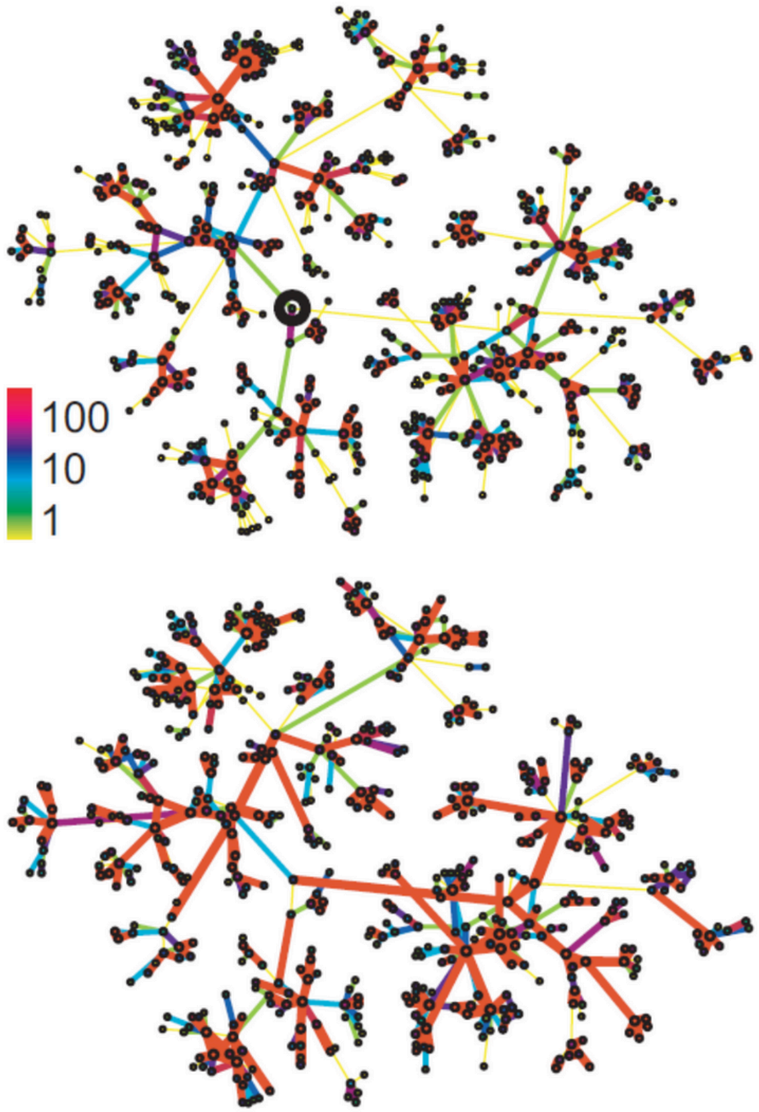
We analyzing real networks we compute their

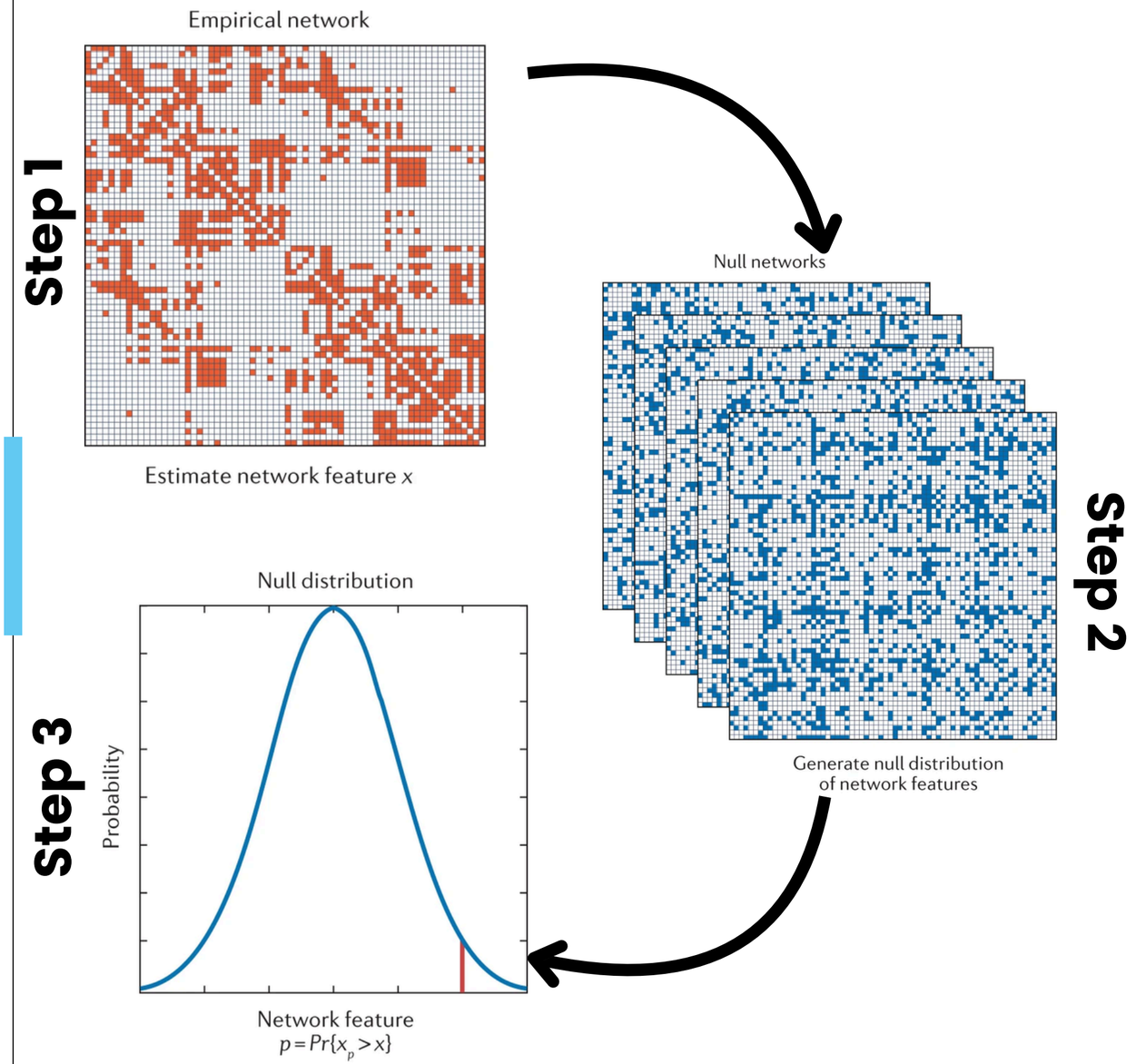
- clustering coefficient
- diameter
- presence of communities

We need to understand if these properties

- can be explained by a random process
- are instead deriving from are more sophisticated mechanism

**A null model is a term of comparison for networks. It generates networks that preserve some properties of the original one, while performing a randomization of its structure**






# Using Null Models

Using a null model involves 3 steps

1. Analyze the real network
   - estimate the feature of interest
2. Generate the null networks
   - decide the type of randomization
   - generate several null networks
3. Compute the distribution of the feature in the null networks
   - determine if what measured in the real case is significant

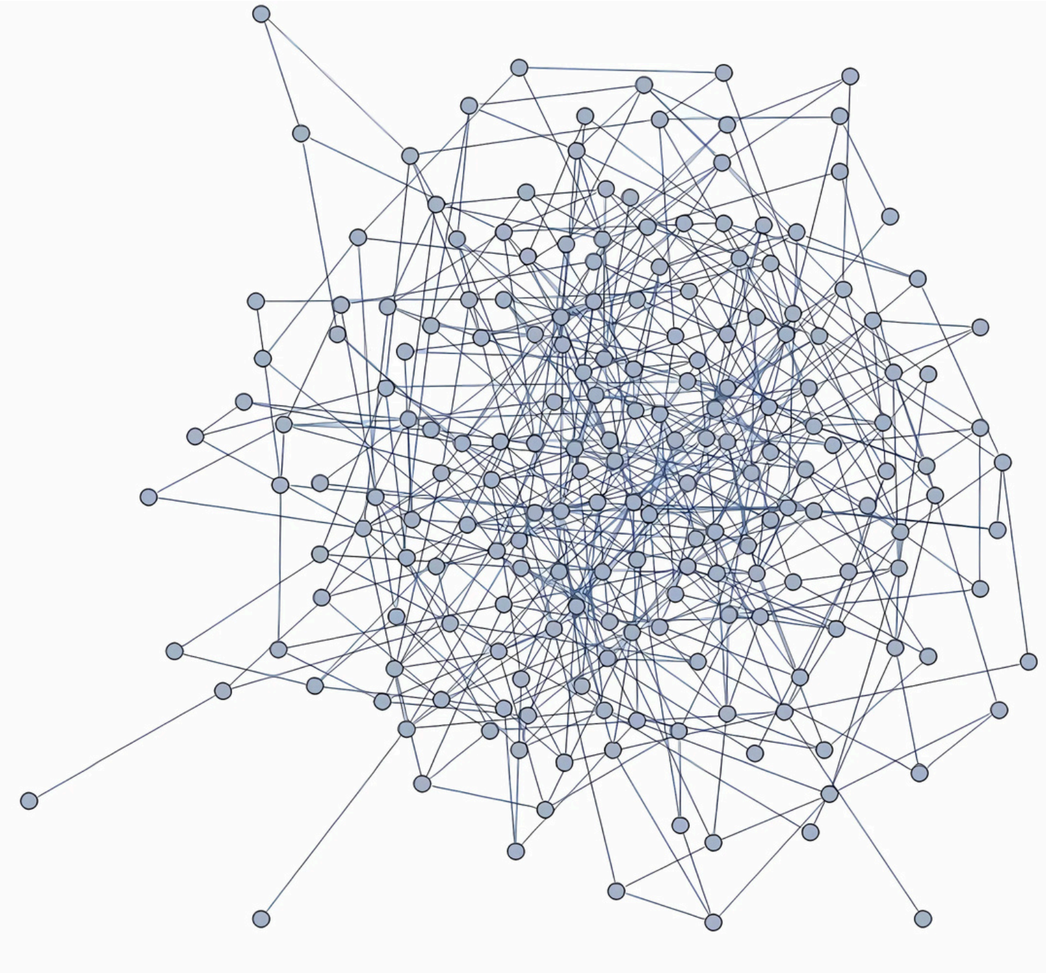
# Erdős–Rényi Model

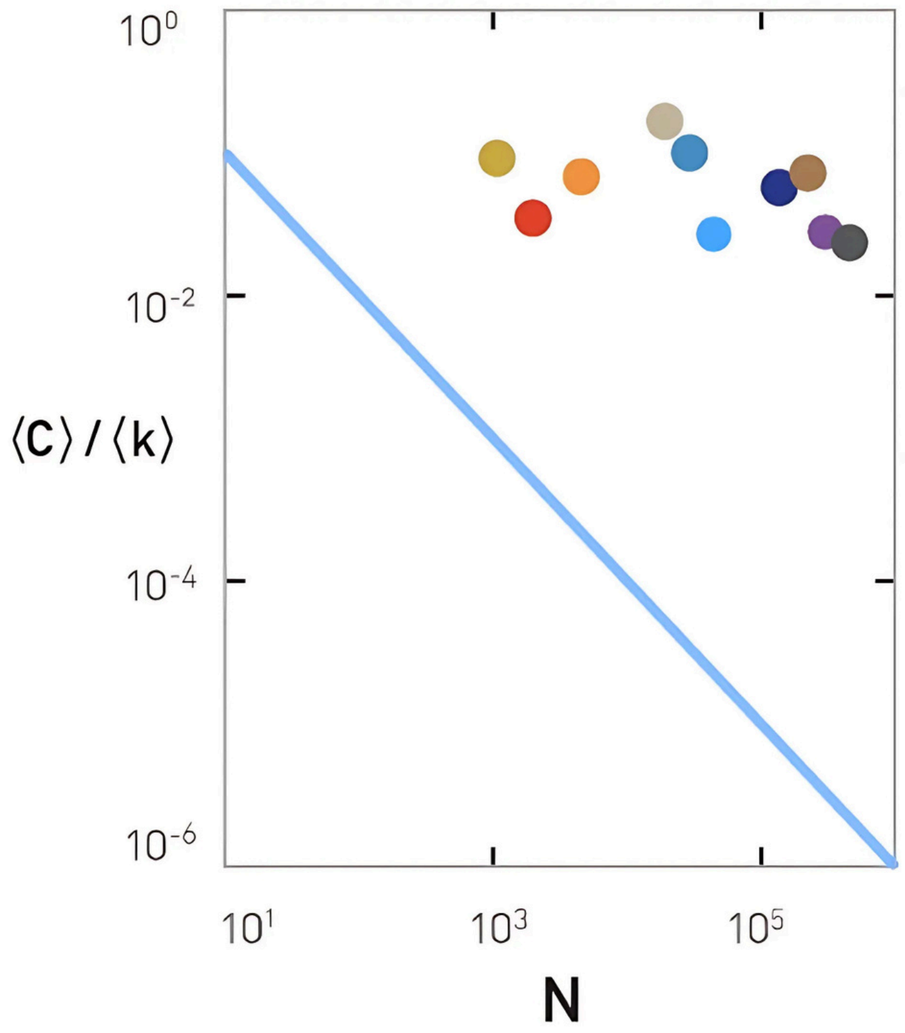
We already encountered an example of random graph, the Erdős–Rényi Model

- the ER model generates networks with a specified average degree
- links are formed completely at random

The ER model can be used as a null model

- we compute the average degree of a real network
- we shuffle all the links in the network generating a random graph with the same average degree






# Recap on Random Graphs

We already used the Erdős–Rényi model as a null model to study real networks

- ER graphs are characterized by the Small-World property like real networks
- differently from real networks have an exponential degree distribution
- the clustering in ER graphs is much smaller than in real networks

Out of the 3 most important properties, the ER null model can only explain one

# Configuration Model

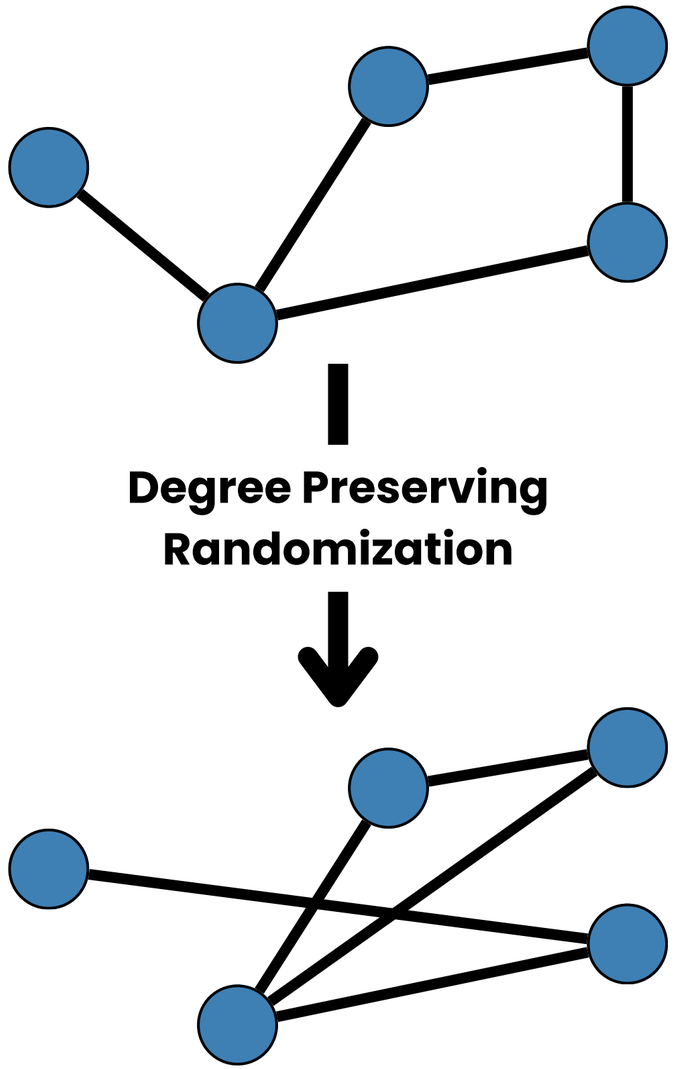
Performing a full randomization of the links like in the ER model is too much

- it destroys all the network structure
- we need something more constrained

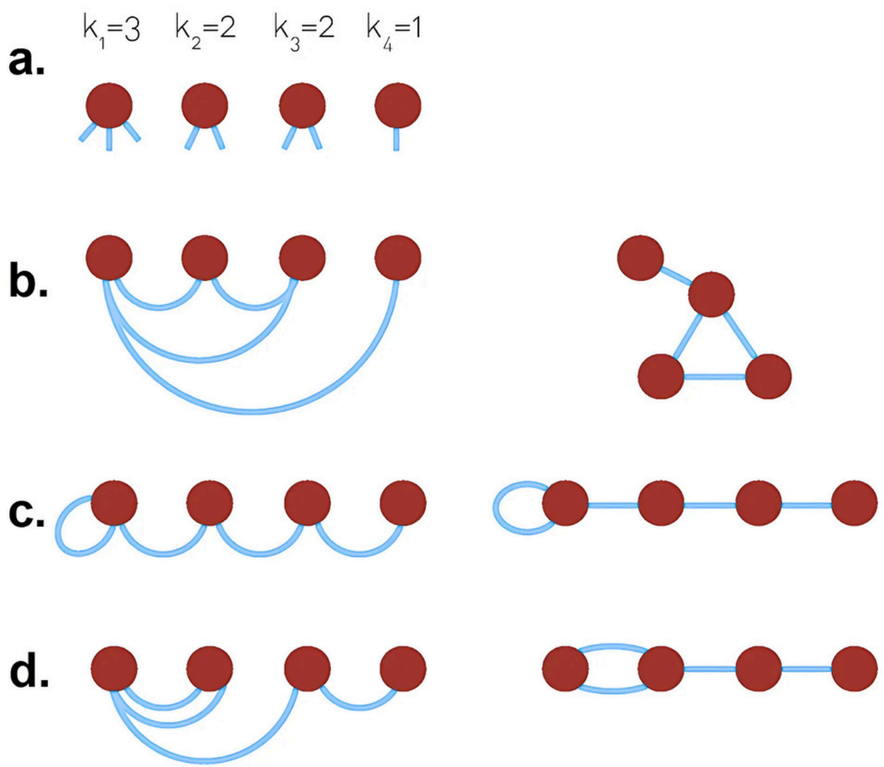
The idea is to perform a randomization, but leaving the degree fixed

- we compute the degree-sequence in the real network
- we generate networks with the same sequence

**This is null model is called Configuration Model and preserves the degree distribution**



# Rewiring Algorithms




The most intuitive way to understand the configuration model is in term of a rewiring process

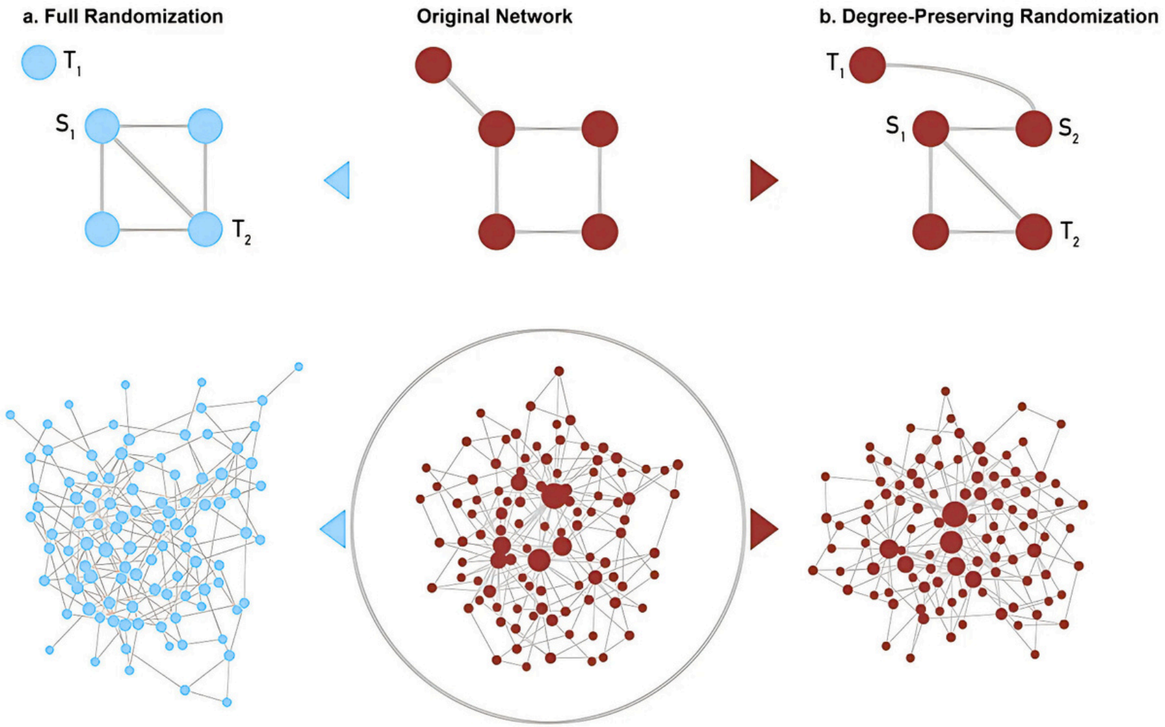
- we start with N disconnected nodes
- each of them has $k_i$ unmatched links, where $k_i$ is their degree in the real network
- we create null networks matching all links from all nodes

This generates all possible configurations while preserving the degrees

# Full vs Degree Preserving Randomization

We can see a comparison between the two null models we introduced

- ER model
  - full randomization of the links
- Configuration model
  - degree preserving randomization



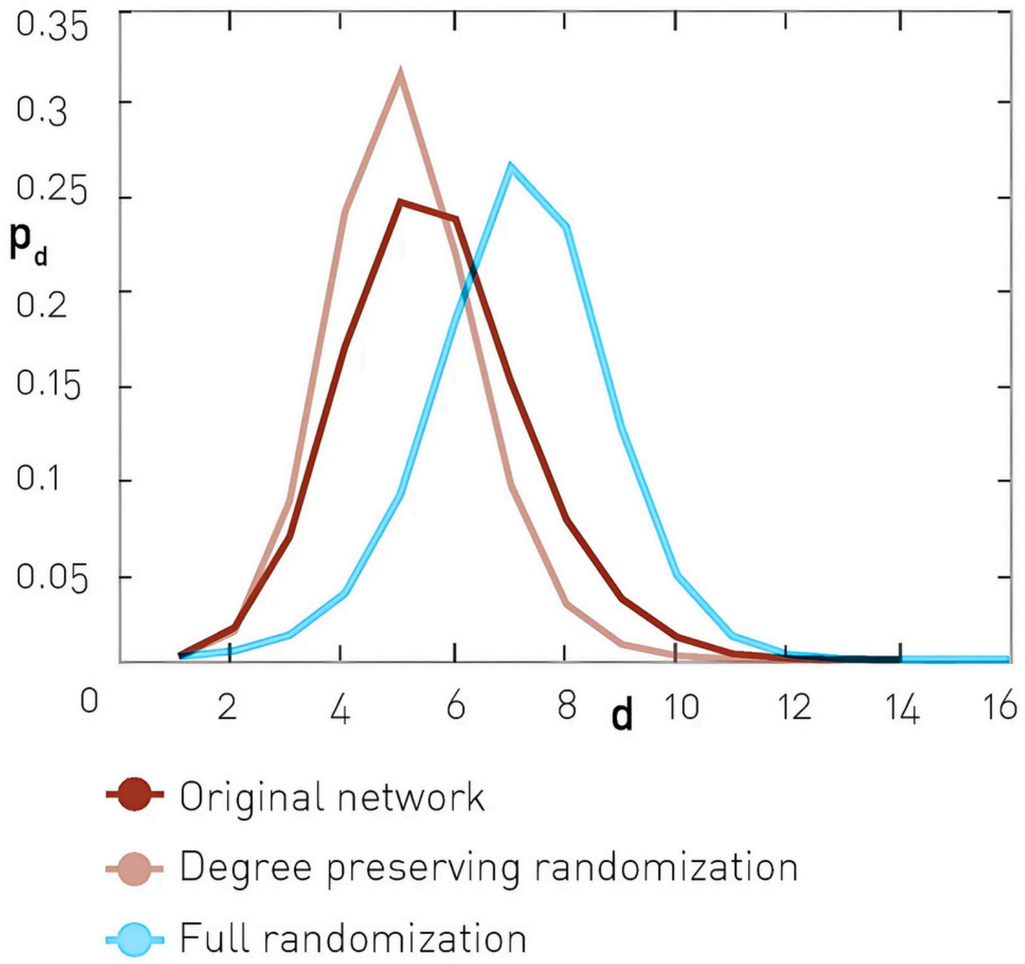
# Full vs Degree Preserving Randomization

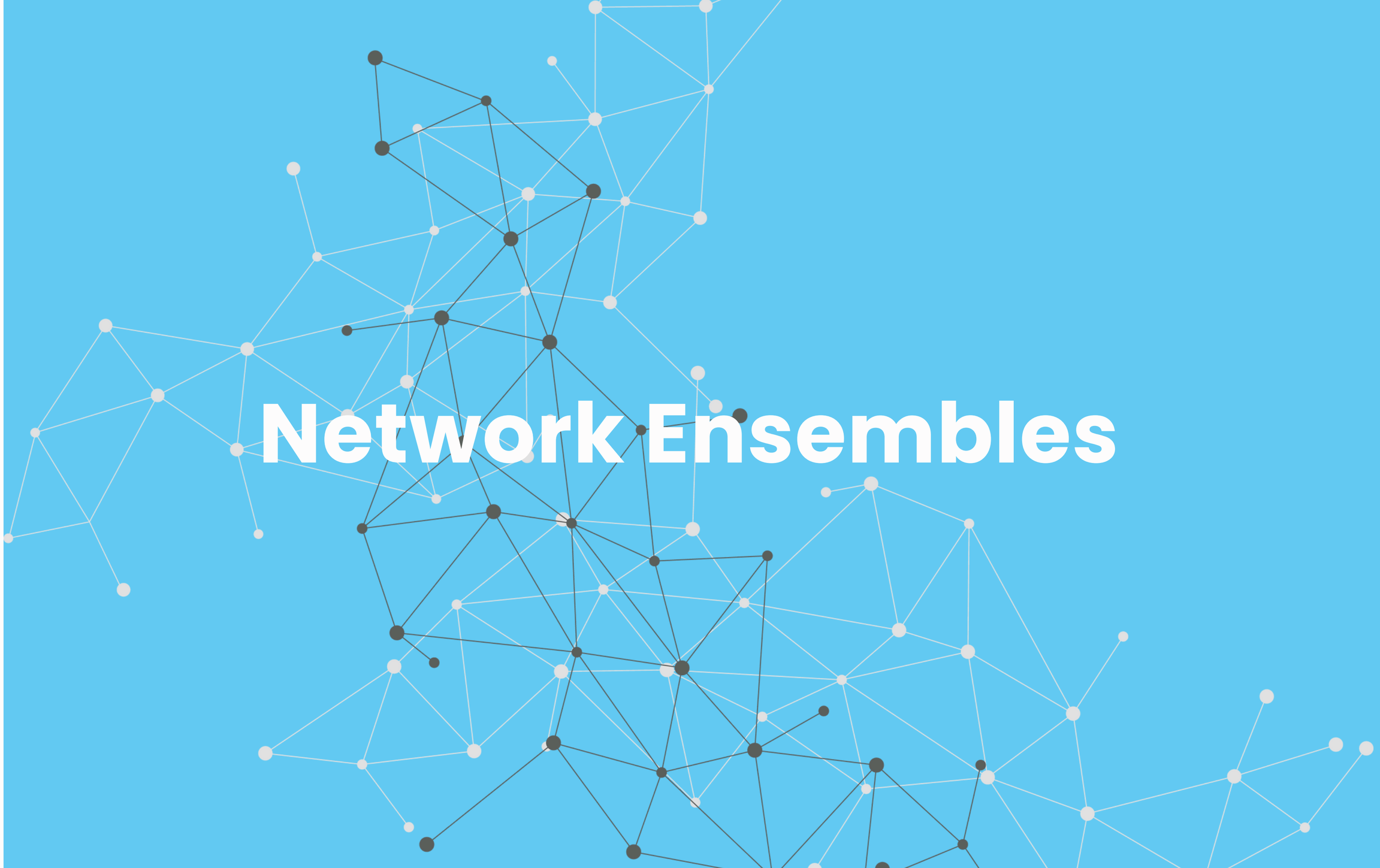
We can see a comparison between the two null models we introduced

- ER model
  - full randomization of the links
- Configuration model
  - degree preserving randomization

In the protein interaction network

- the configuration model better reproduces the distribution of distance in the networks




Network Ensembles

# Ensembles in Physics

Let us consider the air contained in our room

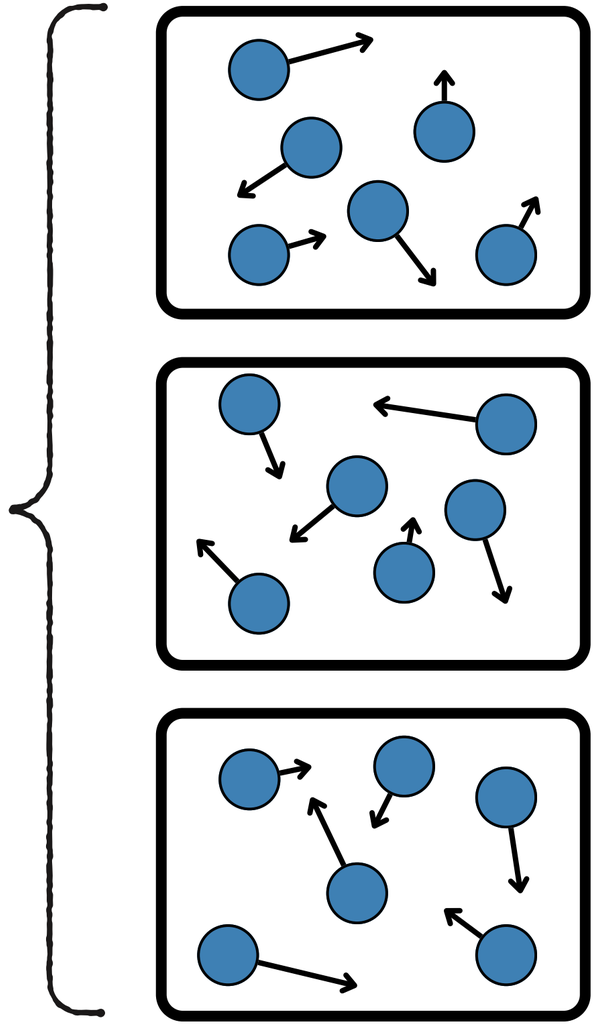
- its macro state is described by few parameters
- pressure, volume and temperature

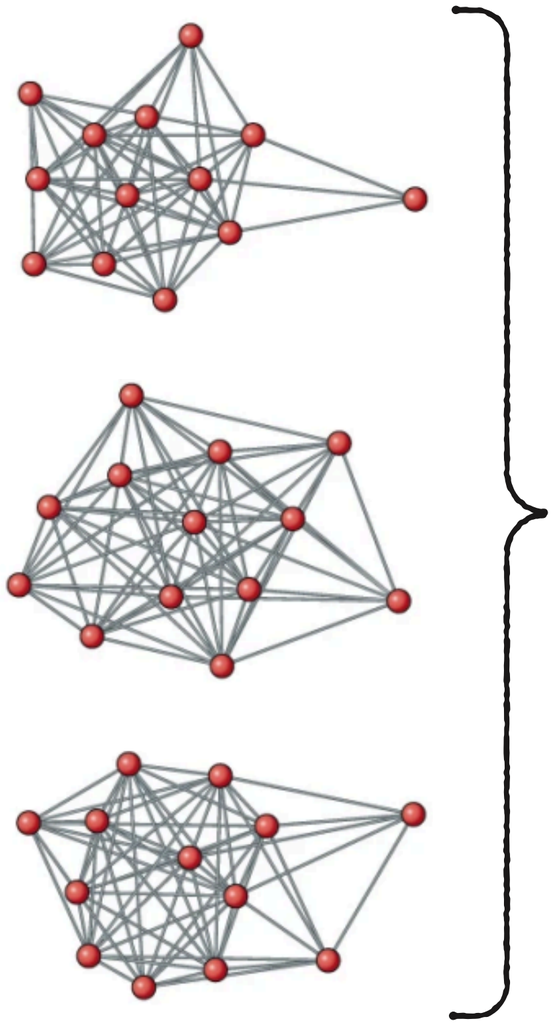
At the microscopic level however

- different particles configurations have the same P, V, T

**An ensemble is the set of all microscopic configurations with the same macroscopic properties**

**Same pressure, volume and temperature**





**Same average degree**

# Network Ensembles

The concept of ensemble can be applied also to networks

**An ensemble of networks is the set of all possible networks sharing the same macro property**

For instance

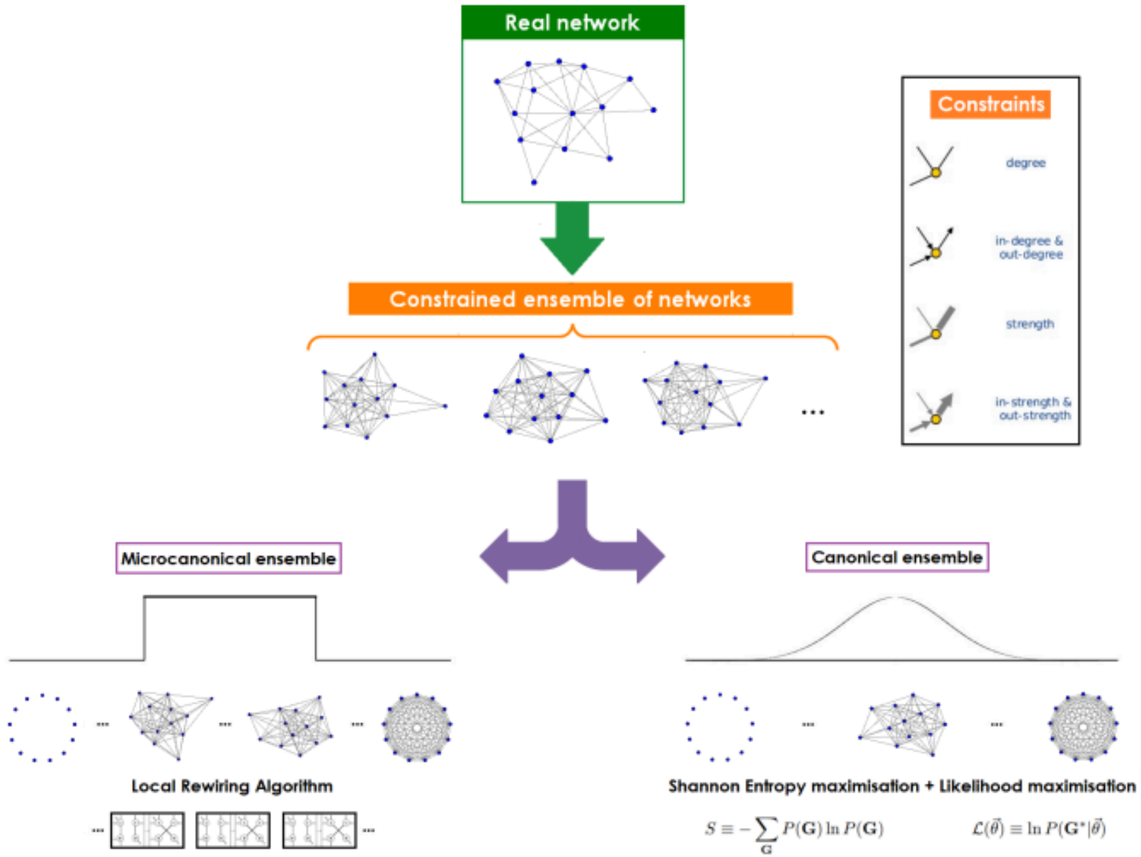
- the ER model generates an ensemble of networks all characterized by the same average degree
- the configuration model an ensemble of networks all characterized by the same degree sequence

# Soft vs Hard Constraints

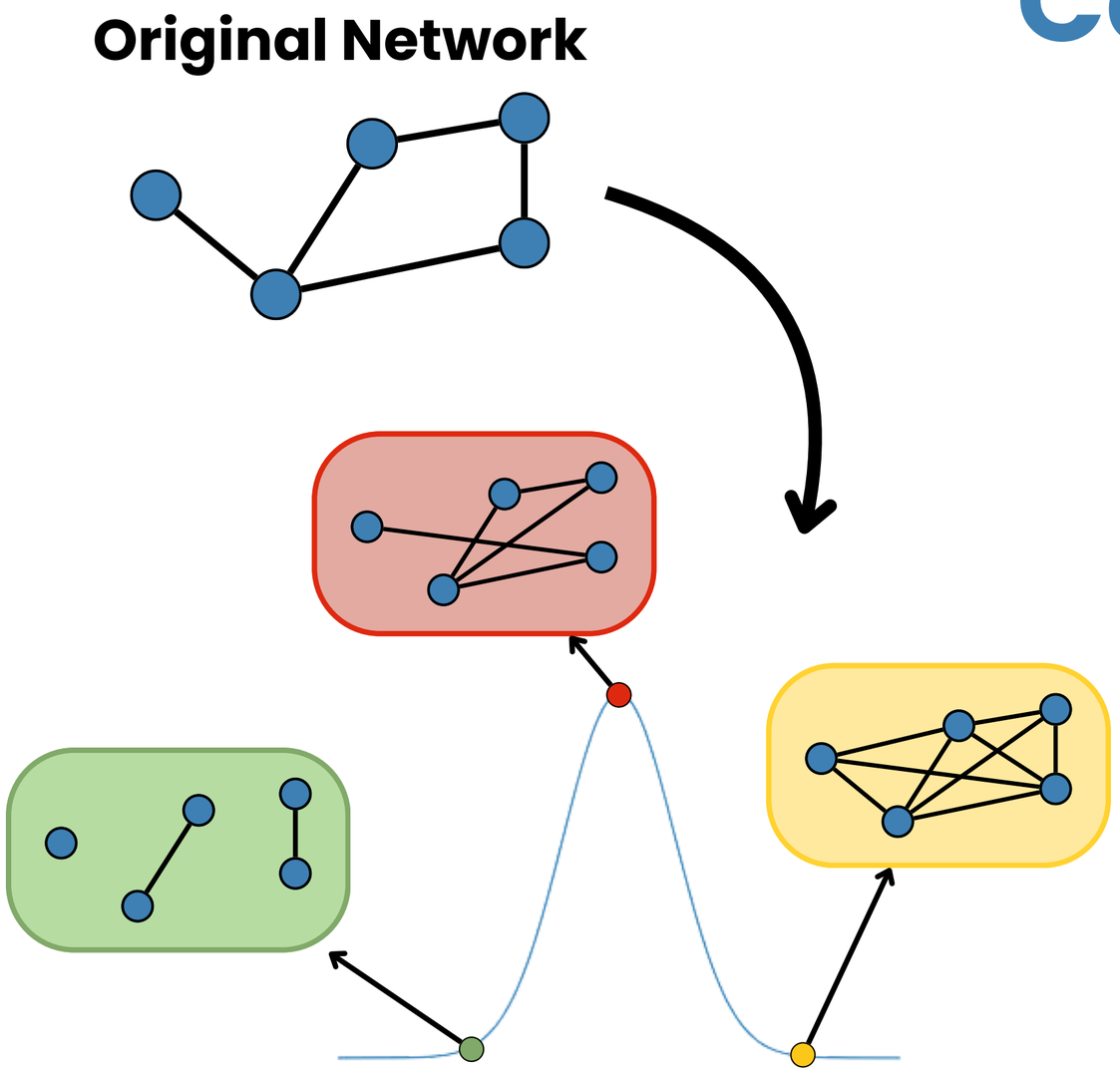
When performing a randomization we always set some constraints. There are two possible approaches

- **Hard Constraints (Microcanonical)**
  Each network in the ensemble satisfies the constraints exactly
- **Soft Constraints (Canonical)**
  The constraints are satisfied on average. Each network in the ensemble slightly deviates from the real one

Soft constraints are generally easier to implement



# Canonical Ensemble




In a canonical ensemble of networks

- networks satisfying the constraints exactly are assigned the largest probability
- networks that do not satisfy the constraints have a small but non zero probability

The constraint we measures from real networks could be

- inexact due to errors
- the result of a stochastic process

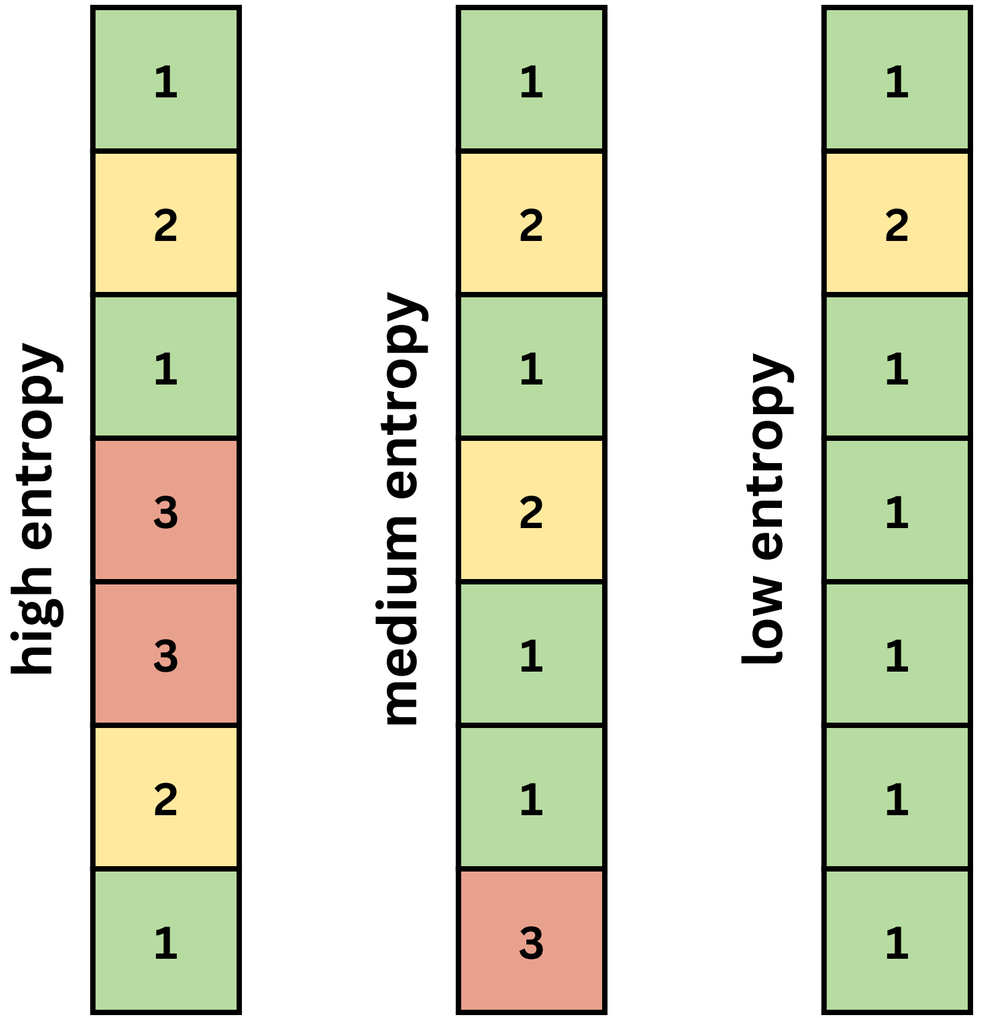
**It makes sense to allow fluctuations!**

# What is Entropy?

Entropy is
- a measure of information content
- a measure of how surprising a sequence is

Larger entropy corresponds to
- a more random sequence
- a sequence that is harder to compress
- less predictable sequences

In this sense entropy is also a measure of order. A high entropy sequence will appear random, while a low entropy one deterministic



# Shannon Entropy

Given a long sequence we can compute the probabilities p(i) by which the M symbols are generated. The Shannon entropy S of the sequence is defined in terms of these probabilities as

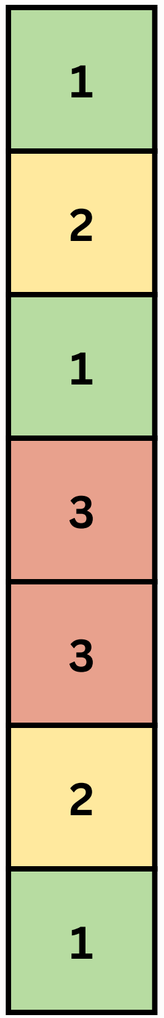
$$S = -\sum_i p(i) \log p(i)$$

The maximal entropy is obtained when all M symbols are equally likely

- $S_{max}$=-log(1/M)
- In our case S=0.46 and $S_{max}$=0.48

Entropy can be generalized to continuous distributions

$$S = -\int p(x) \log p(x) dx$$




**p(1)=3/7**

**p(2)=2/7**

**p(3)=2/7**

# Entropy Maximization

Entropy is maximal when the probability distribution is uniform

- maximal entropy corresponds to the least informative probability
- in a null model we want to constraint something (e.g. the degree sequence) while leaving the rest of the features maximally random

The key idea in deriving a canonical ensemble of networks consists in maximizing the entropy of the probability distribution of networks P(G)

- the ensemble is defined by P(G), the probability of generating a given graph G
- if we maximize the Shannon entropy of P(G) we get a uniform distribution
  - in this case all graphs would be equally likely
- to implement the constraints we perform a constrained entropy maximization
  - we maximize entropy
  - at the same time we require the probability P(G) to satisfy some constraints
  - for instance we may require the average degree to be a given value

# Canonical Network Ensemble

Let's write the math behind this idea. The Shannon entropy is

$$S = -\sum_{G \in \mathcal{G}} P(G) \ln P(G)$$

We consider M constraints $x_i(G)$ that we want the graphs to satisfy on average

- for instance we could have $x_1(G)$=average degree and $x_2(G)$=clustering

These are mathematically defined as

$$\sum_G P(G) x_i(G) = \langle x_i \rangle$$

To maximize the entropy subject to these constraint we use the Lagrange multiplier formula. Introducing the Lagrange multipliers $\theta_i$ we get

$$\frac{\partial}{\partial P(G)} \left[ S + \alpha \left( 1 - \sum_G P(G) \right) + \sum_i \theta_i \left( \langle x_i \rangle - \sum_G P(G) x_i(G) \right) \right] = 0$$

# Canonical Network Ensemble

Solving this equation leads to the following expression for the graph probability

$$P(G) = \frac{e^{-H(G)}}{Z}$$

Here H(G) is called graph Hamiltonian and Z is a normalization constant

$$H(G) = \sum_i \theta_i x_i(G) \qquad Z = \sum_G e^{-H(G)}$$

The M Lagrange multipliers are obtained solving the M constraints equations

$$\sum_G P(G) x_i(G) = \langle x_i \rangle$$

Here is where the real networks enters. We use it to compute the M real values of the graph properties $x_i(G)$ and we use these values to solve the equations for the Lagrange multipliers

# Erdős–Rényi Model

First we consider as constraint the total number of links m in the network

$$\sum_G P(G)m(G) = m$$

The Hamiltonian and the probability are

$$H(G) = \theta m(G) \qquad P(G) = \frac{\mathrm{e}^{-H(G)}}{Z} = \frac{\mathrm{e}^{-\theta m(G)}}{\sum_G \mathrm{e}^{-\theta m(G)}} = \frac{\mathrm{e}^{-\theta m(G)}}{(1+\mathrm{e}^{-\theta})^{\frac{1}{2}N(N-1)}}$$

To compute θ we use the constraint equation

$$\sum_G P(G)m(G) = m \to \frac{1}{(1+\mathrm{e}^{\frac{1}{2}N(N-1)})} \sum_G \mathrm{e}^{-\theta m(G)} m(G) = m \to \frac{1}{1+\mathrm{e}^{\theta}} = p \quad \text{with} \quad p = \frac{m}{\frac{1}{2}N(N-1)}$$

Using this we can see that the probability P(G) is nothing but the ER network

$$P(G) = p^{m(G)}(1-p)^{\frac{1}{2}N(N-1)-m(G)}$$

Prob. for m(G) random links to exist

Prob. for the rest of the links not to exist

# Configuration Model

To get the canonical configuration model we have to fix the degree sequence, so we will have N constraints, one for each node

$$\sum_G P(G)k_i(G) = k_i$$

Performing the same procedure as before we can obtain the linking probability, that gives the probability for a link between two nodes i and j to exist

$$p_{ij} = \frac{e^{-(\theta_i+\theta_j)}}{1+e^{-(\theta_i+\theta_j)}}$$

Also in this case we can compute the N Lagrange multipliers using the constraint

$$\sum_j p_{ij} = k_i$$

Once we have the multipliers, we can use $p_{ij}$ to sample each link independently

# Comparison of Ensembles

| | Microcanonical | Canonical |
|---|---|---|
| **Constraints** | Satisfied exactly | Satisfied on average |
| **Algorithm** | Link swapping | Entropy maximization |
| **Drawbacks** | Slow<br>Potentially biased | Soft constraints |
| **Advantages** | Hard constraints | Robust to noise |

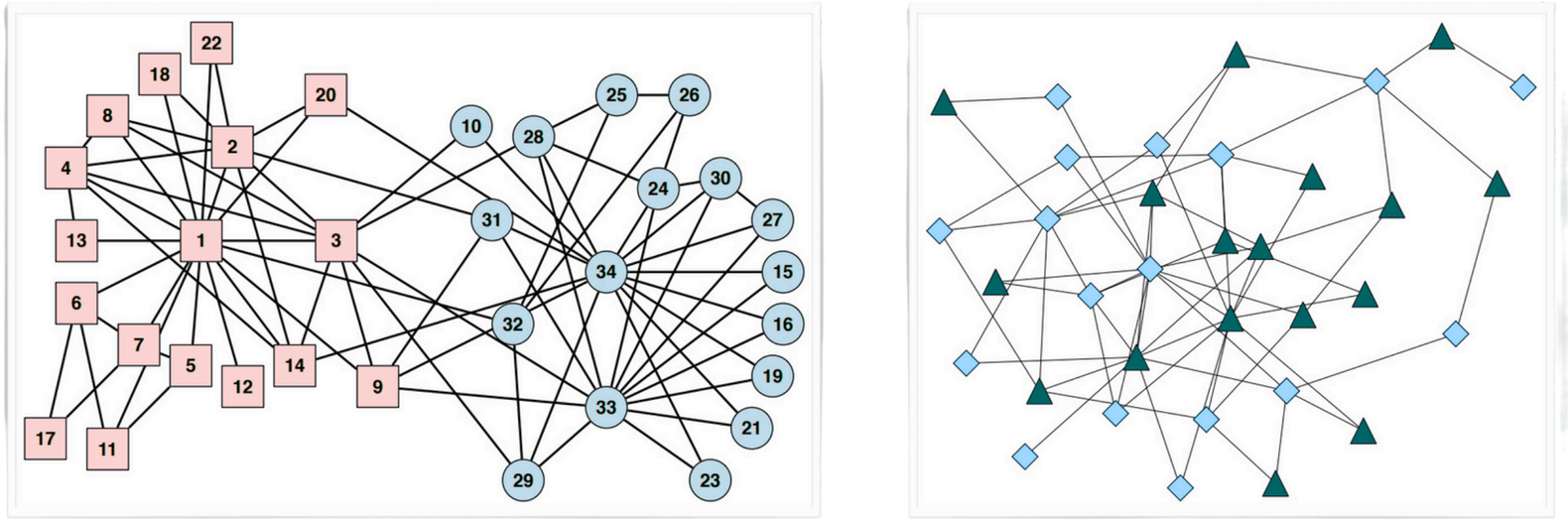
# Applications of Null Models

# Randomizing Real Networks

One of the main applications of null models is to validate network properties

- on the left we report the Zachary Karate Club network
- on the left we show one of its possible randomized versions
- this has been obtained using the configuration model

**The community structure is a statistically significant feature!**

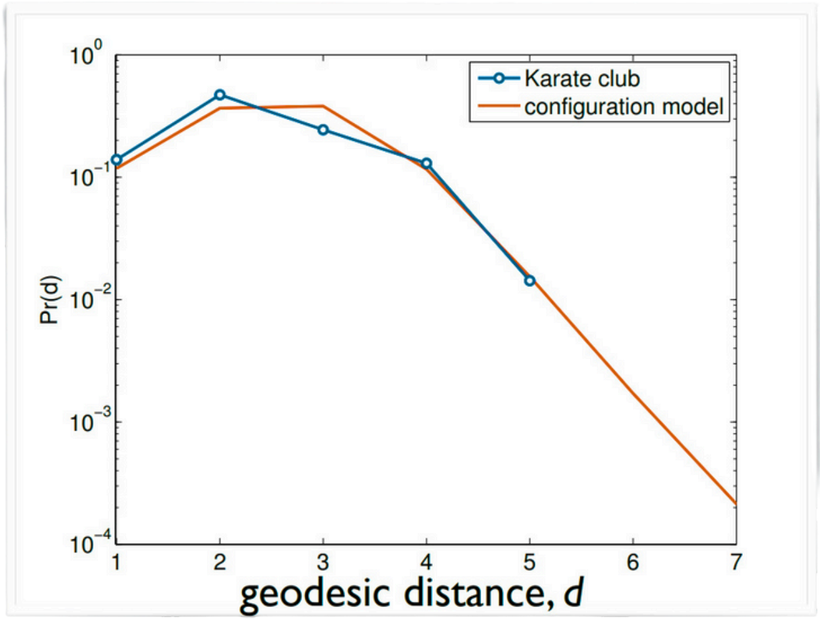


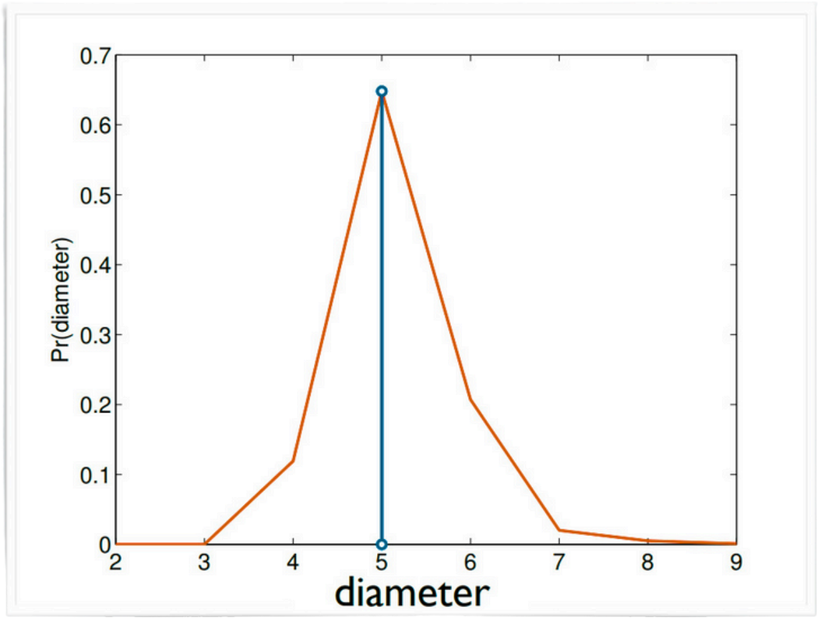

https://sites.santafe.edu/~aaronc/slides/Clauset_2019_CSSS_Networks_3.pdf

# Diameter of Zachary Karate Club

Differently the distribution of the distance and the diameter are very well reproduced by the null model

- the distribution of the distance in the null model vs real network is almost the same
- the distribution of the diameter in the null networks is peaked on the real value



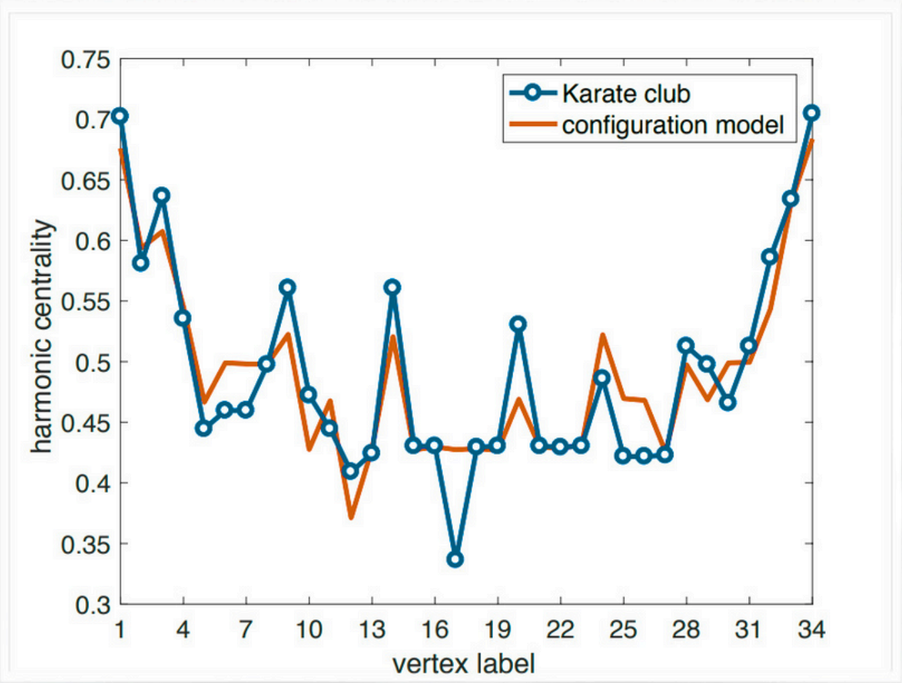



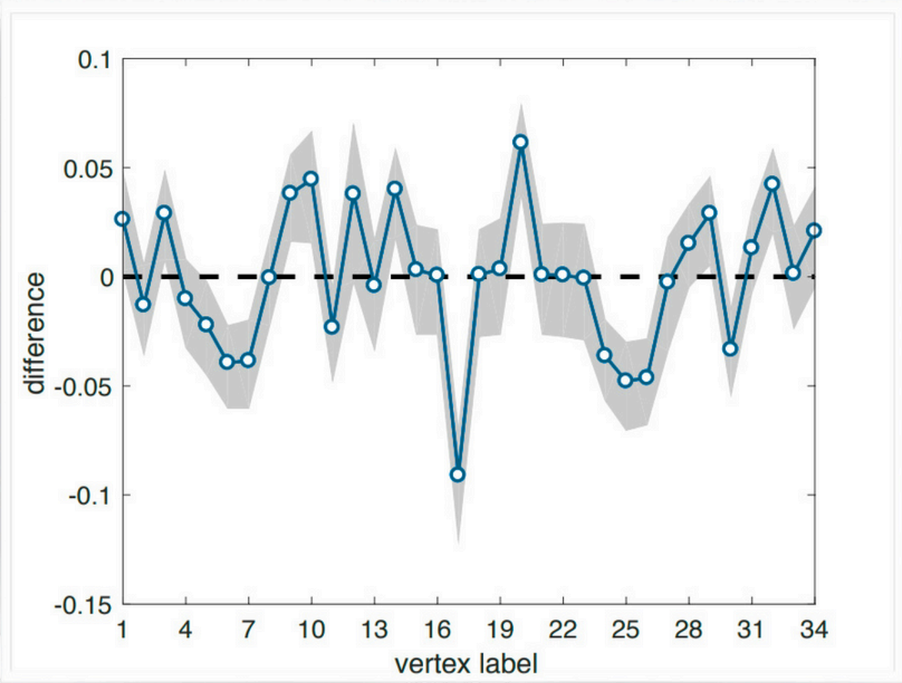

https://sites.santafe.edu/~aaronc/slides/Clauset_2019_CSSS_Networks_3.pdf

# Centrality in Zachary Karate Club

Same story for the harmonic centrality (similar to closeness centrality)

- there are very small deviations with respect to the null model
- it is almost completely explained by the degree sequence



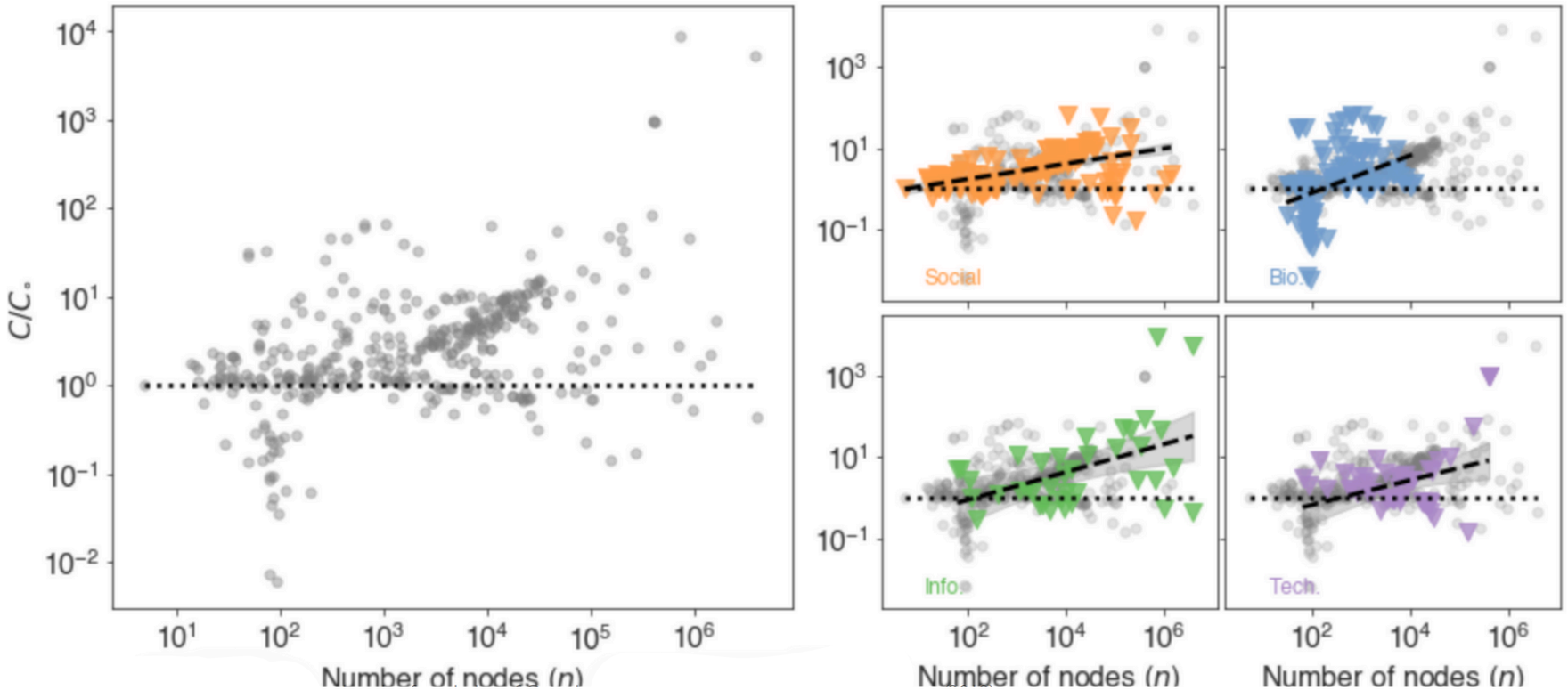




https://sites.santafe.edu/~aaronc/slides/Clauset_2019_CSSS_Networks_3.pdf

# Clustering in Real Networks

Using the configuration model we can study the clustering of real networks

- real networks have higher clustering, but much is explained by the null model




https://sites.santafe.edu/~aaronc/slides/Clauset_2019_CSSS_Networks_3.pdf

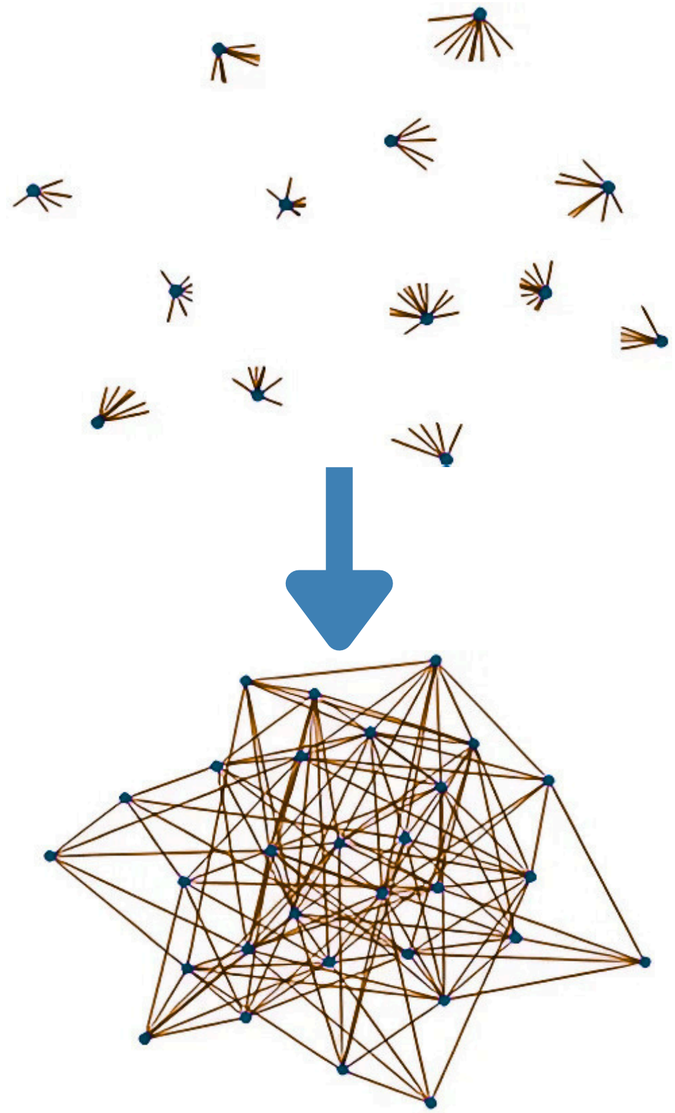
# Network Reconstruction

A very important application of null models, in particular of maximum entropy based ones, is network reconstruction

- often only partial information about a network is available
- for instance we may know all degrees, but not how the links are distributed

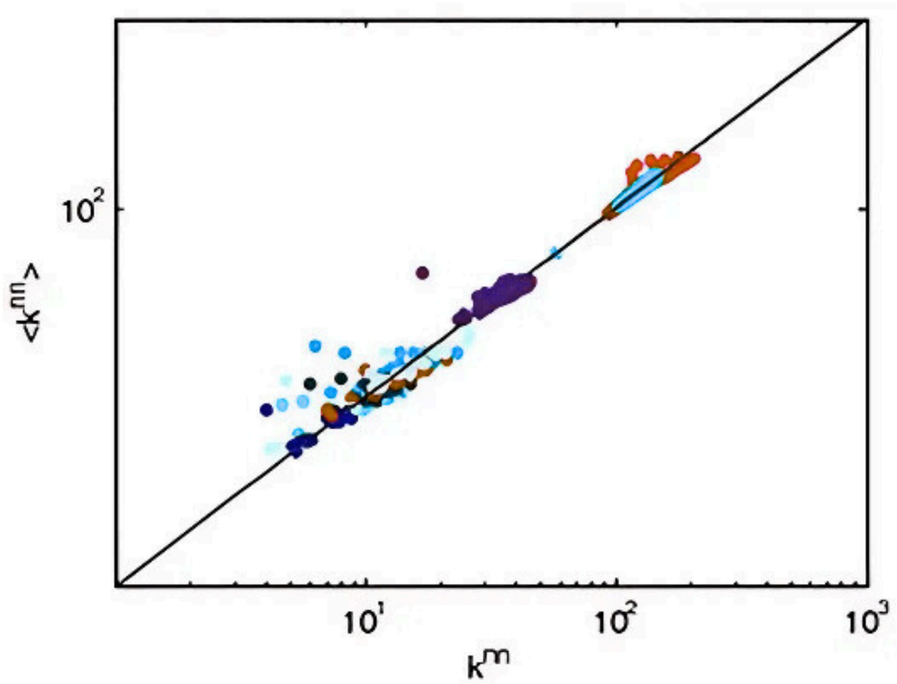
The problem is like finding the best curve fitting a set of points

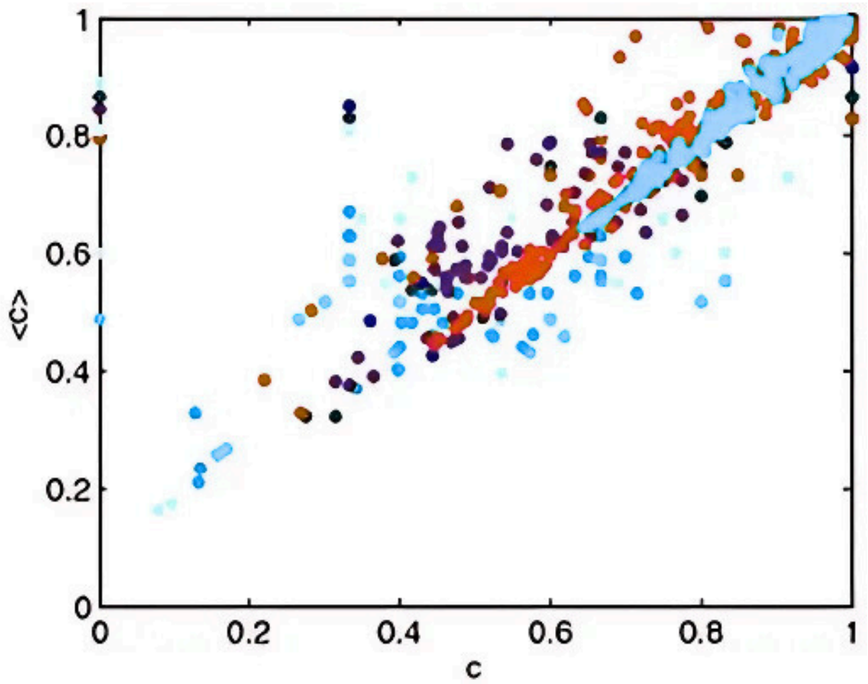
- we want to find the best fit network
- however instead of a network we get an ensemble of networks



# Undirected Binary Networks

In the case of binary undirected networks we can reconstruct the network structure only knowing the degree sequence. This allows to get null networks that very well approximate the real ones



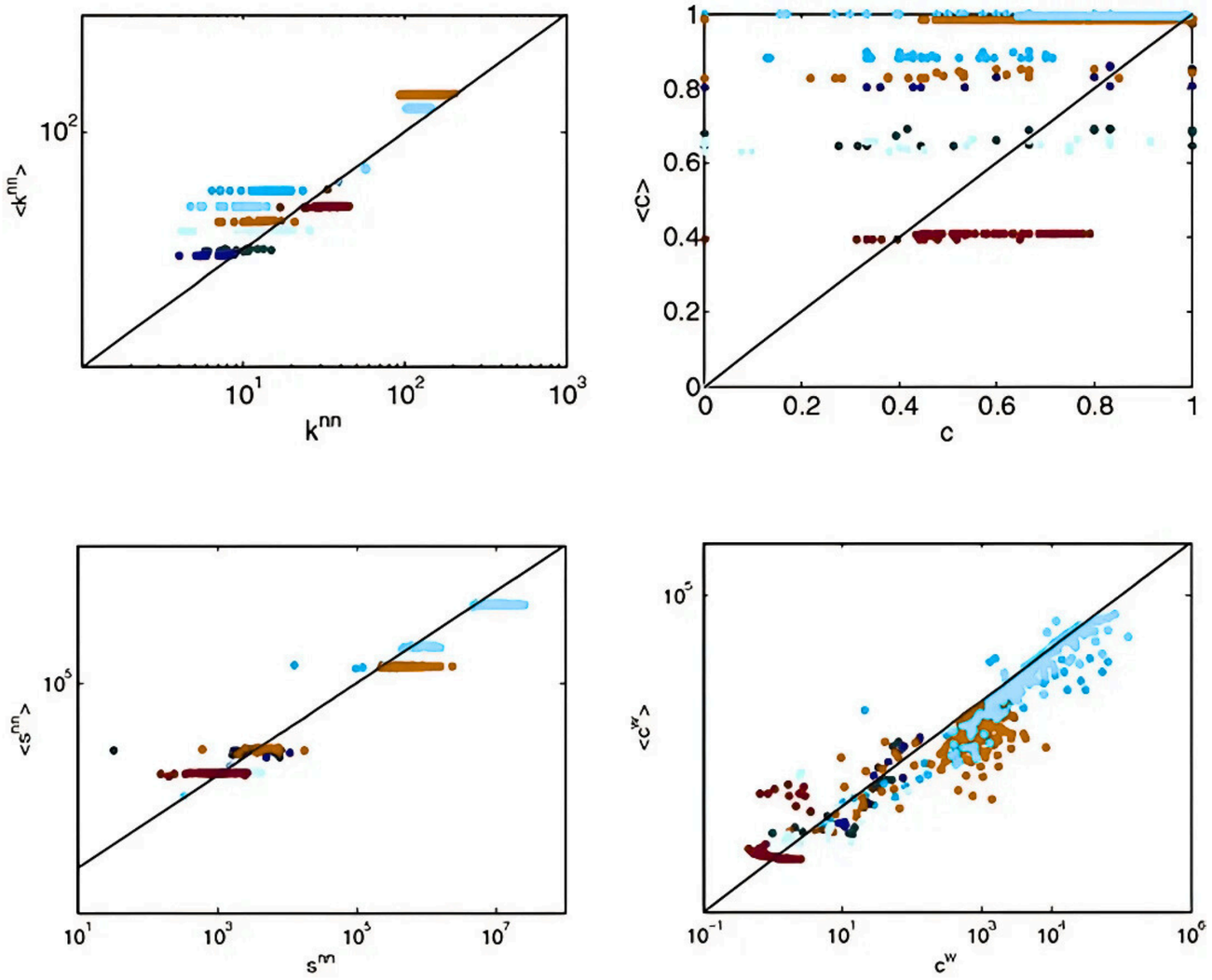

Squartini, T., Garlaschelli, D. (2017). *Network Reconstruction*. In: Maximum-Entropy Networks. SpringerBriefs in Complexity. Springer, Cham. https://doi.org/10.1007/978-3-319-69438-2_4

# Weighted Networks

A similar approach can be tried also for weighted networks

- in this case we constrain the sequence of the strengths instead of the degrees
- we obtain a variant of the configuration model

However in this case the null networks fail to capture the main properties of the real networks

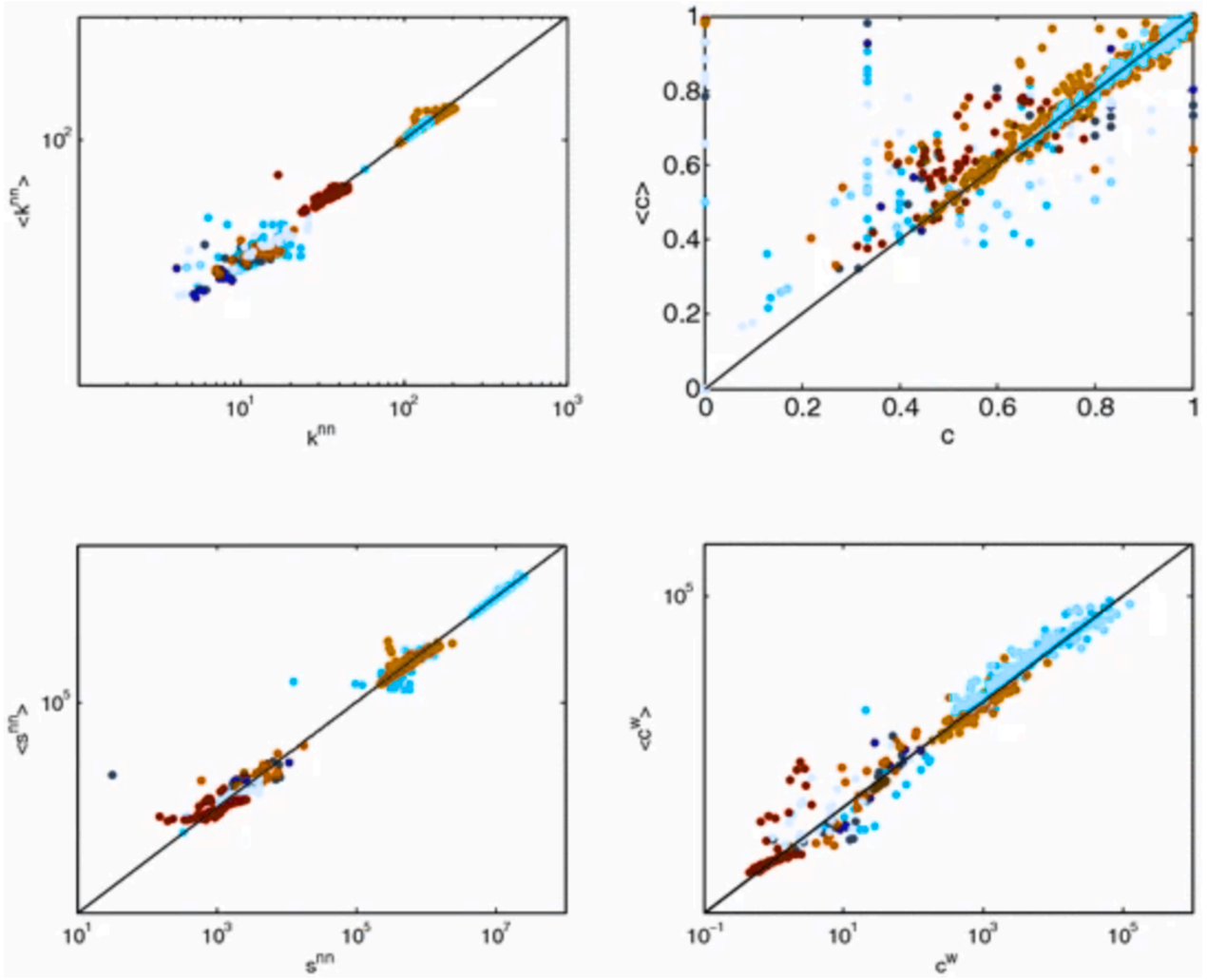


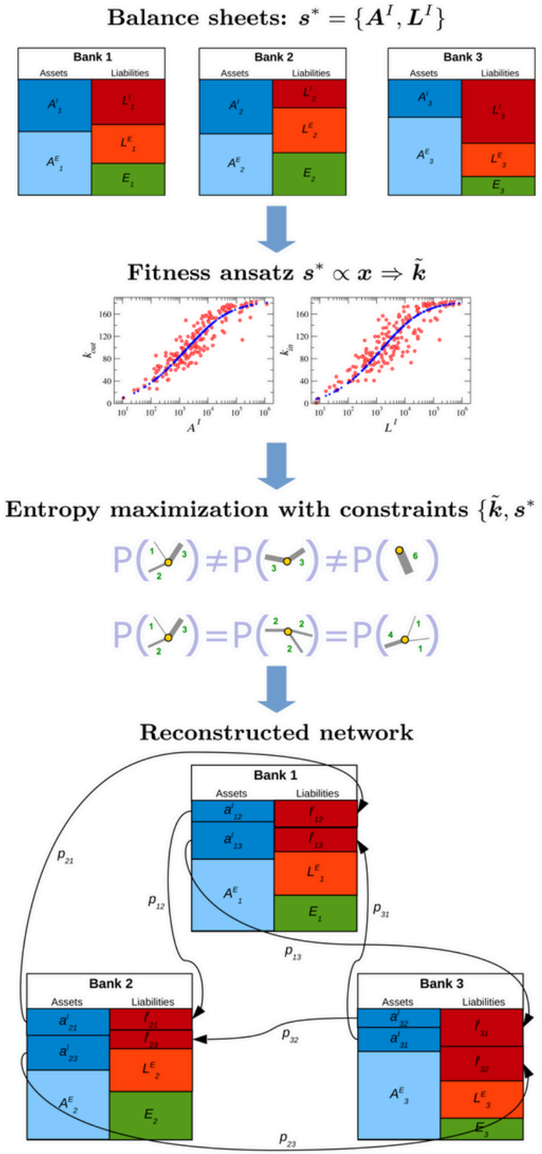
# Enhanced Configuration Model

Things improve considerably if we instead use the Enhanced Configuration Model

- we fix both the degree and the strength sequence
- we keep also structural information from the degree

In this way the null networks are very good fits to the real ones






# Bank Networks

Network reconstruction is crucial is the analysis of bank networks

- banks only provide limited information
- they do not state their exposition to other banks
  - for instance bank A may have stock or bonds of bank B
  - if bank B goes bankrupt, this could cause a sudden failure of A also
- more sophisticated versions of the maximum entropy recipe are used

**These techniques allow to understand the risk of cascade failures in the bank network**

# Bipartite Networks Projection

# Bipartite Networks

Many networks involve nodes of different categories

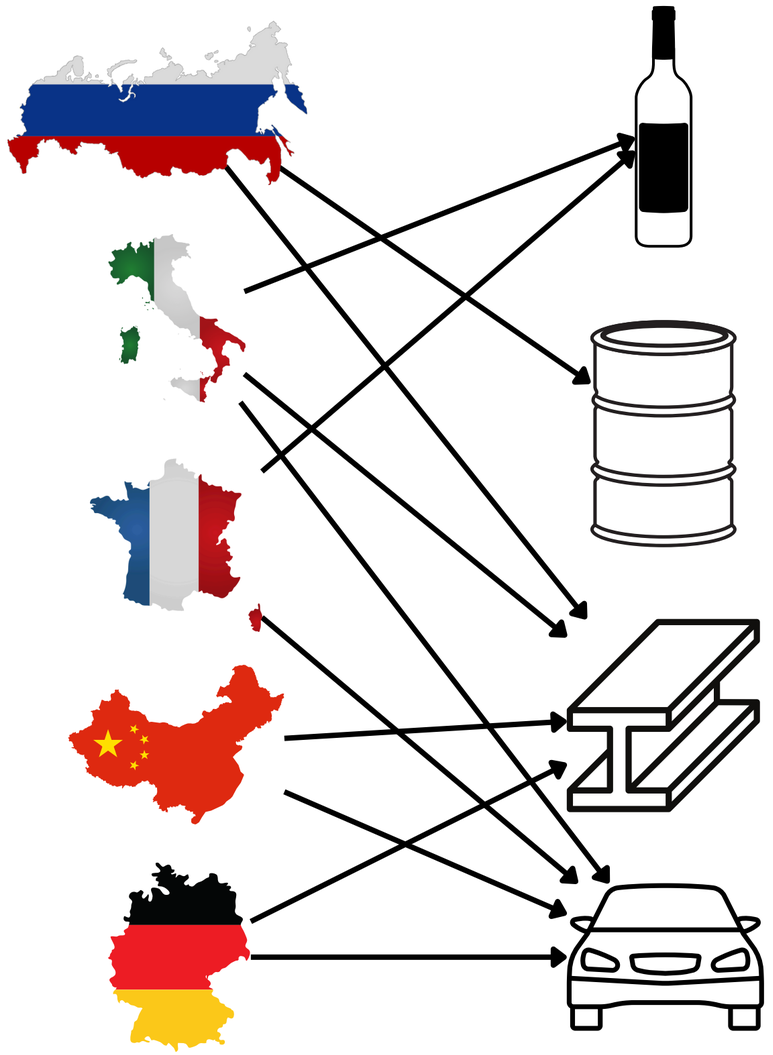
- for instance we may have a network of countries and products
- links connect countries to the products they export

Such a network is called bipartite

- it is composed of two “layers” of nodes
- there are no connections among nodes in the same layer

Bipartite networks require specific tools

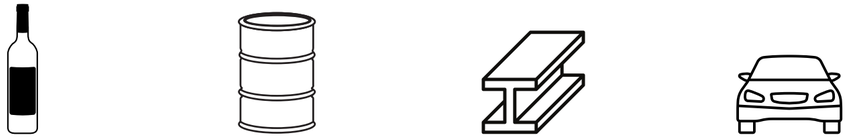
- how do we detect communities?
- what about centrality measures?



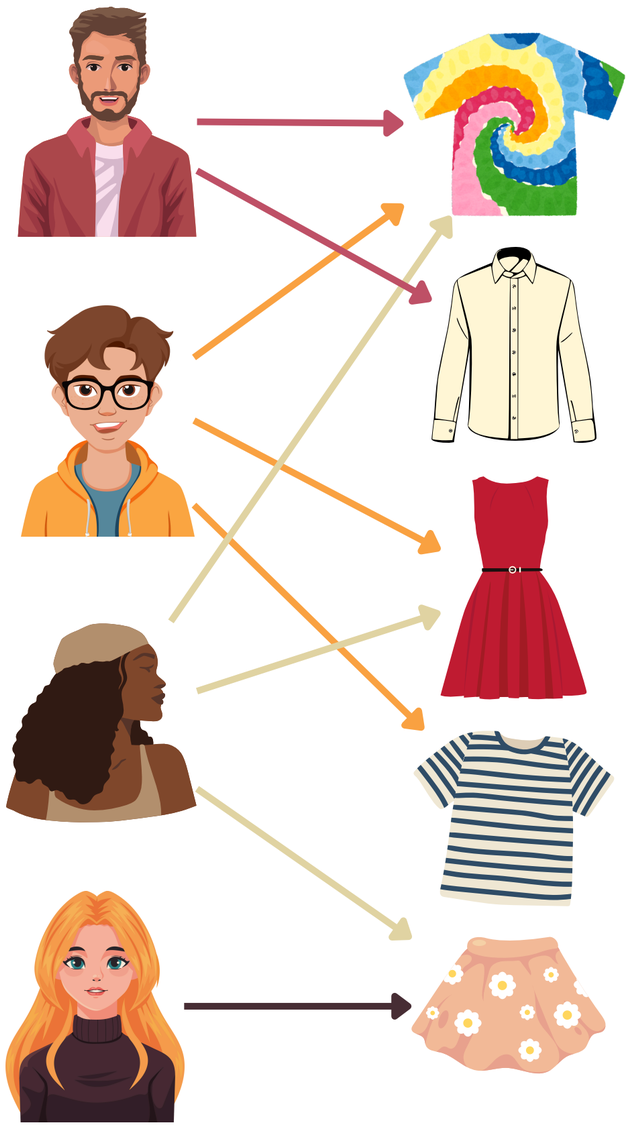
# Biadjacency Matrix

- A bipartite network is described by its biadjacency matrix M
- it is an CxP matrix
  - C is the number of nodes in the first layer (countries)
  - P si the numer of nodes in the second layer (products)
- it works similarly to the adjacency matrix
  - $M_{cp}$=1 if node c is connected to node p
  - $M_{cp}$=0 otherwise
- we can easily generalize it to a weighted biadjacency matrix by allowing values different from 0/1

**M**



| | | | |
|---|---|---|---|
| 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 1 |
| 1 | 0 | 0 | 1 |
| 0 | 0 | 1 | 1 |
| 0 | 0 | 1 | 1 |



# Examples of Bipartite Networks

Bipartite networks are ubiquitous.
Notable examples include

- mutualistic networks
  - plants-pollinators
- purchases networks
  - customers-items
- online platforms
  - users-content

Often bipartite networks contain

- actors or active nodes (e.g. users)
- passive nodes (e.g. content)

# Bipartite Networks Projection

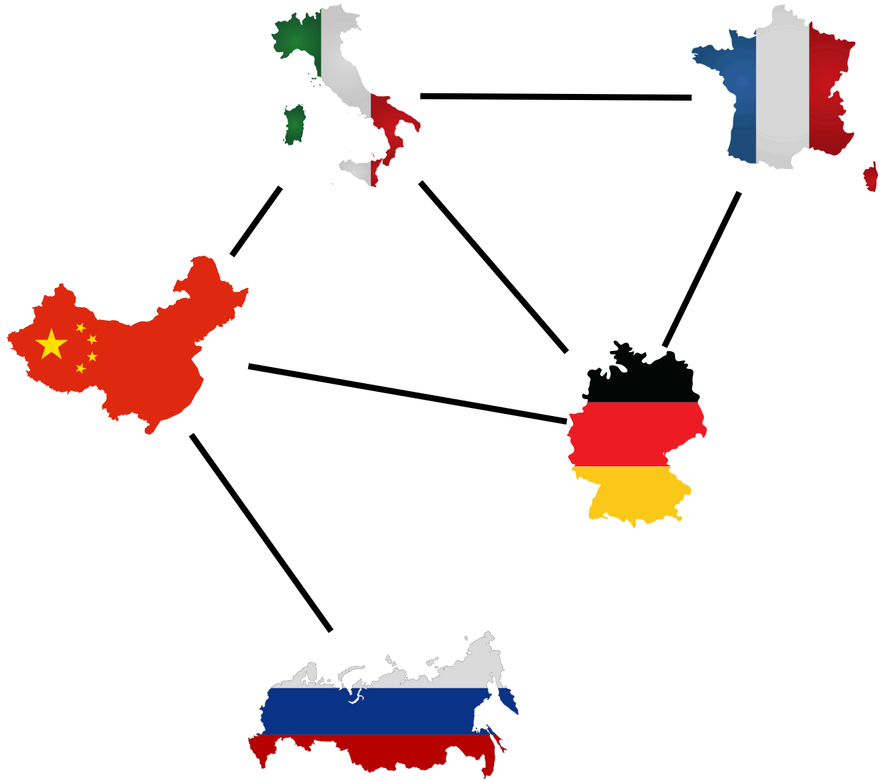
Bipartite networks can be projected to obtain a monopartite network

- this network contains only nodes of one type
- connections between nodes imply a similarity of some sort

The most simple approach to network projections is based on cooccurrences

- nodes in layer 1 that are linked to the same nodes in layer 2 will be similar
- we can then define the adjacency matrix A of the projected networks as

$$W_{ij} = \sum_p M_{ip} M_{jp} \rightarrow A_{ij} = \begin{cases} 1 & \text{if} \quad W_{ij} > 0 \\ 0 & \text{otherwise} \end{cases}$$



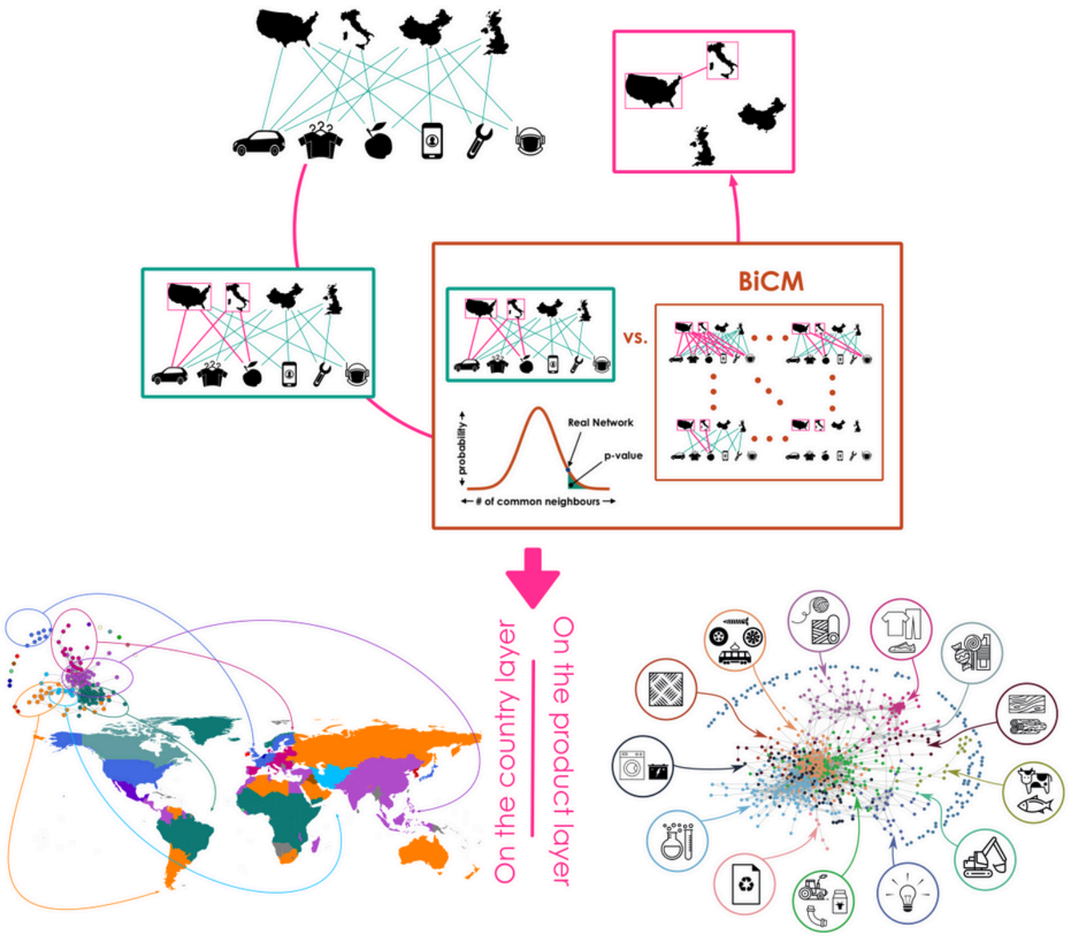
# Projection Validation

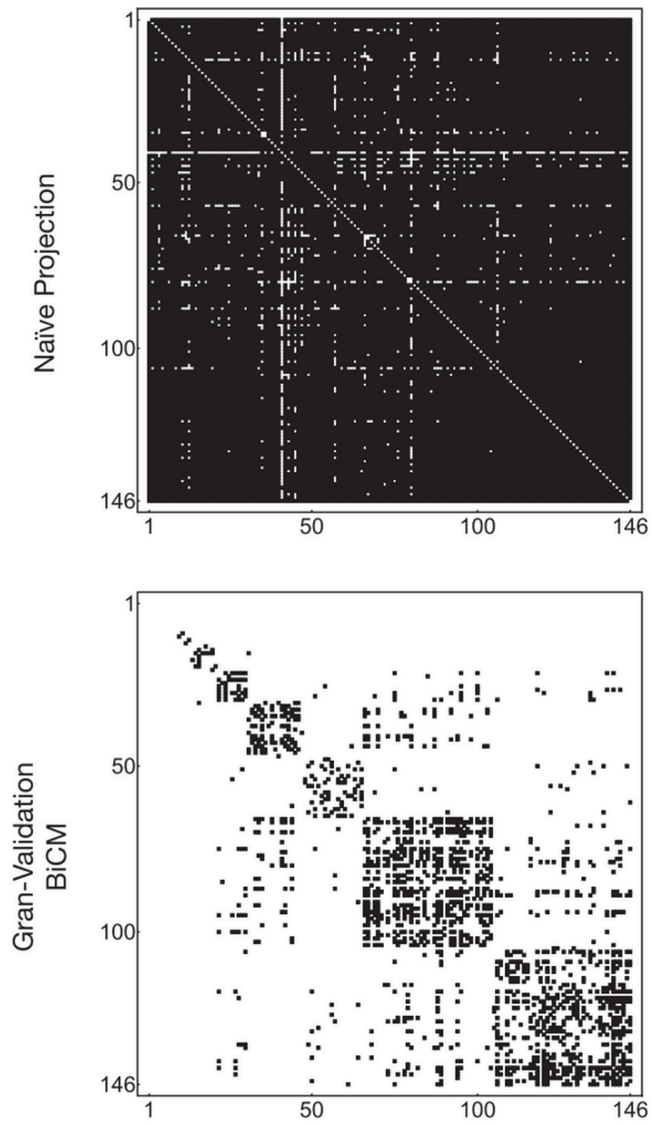
When performing bipartite networks projection null models are crucial

- the best approach consists in using the Bipartite Configuration Model (BiCM)
- it fixes the degree of nodes in both layers

The idea is to compute the weighted projection matrix W in the real and in an ensemble of degree preserved randomized networks

- only links with a value $W_{ij}$ larger than what obtained in the null models are set to 1





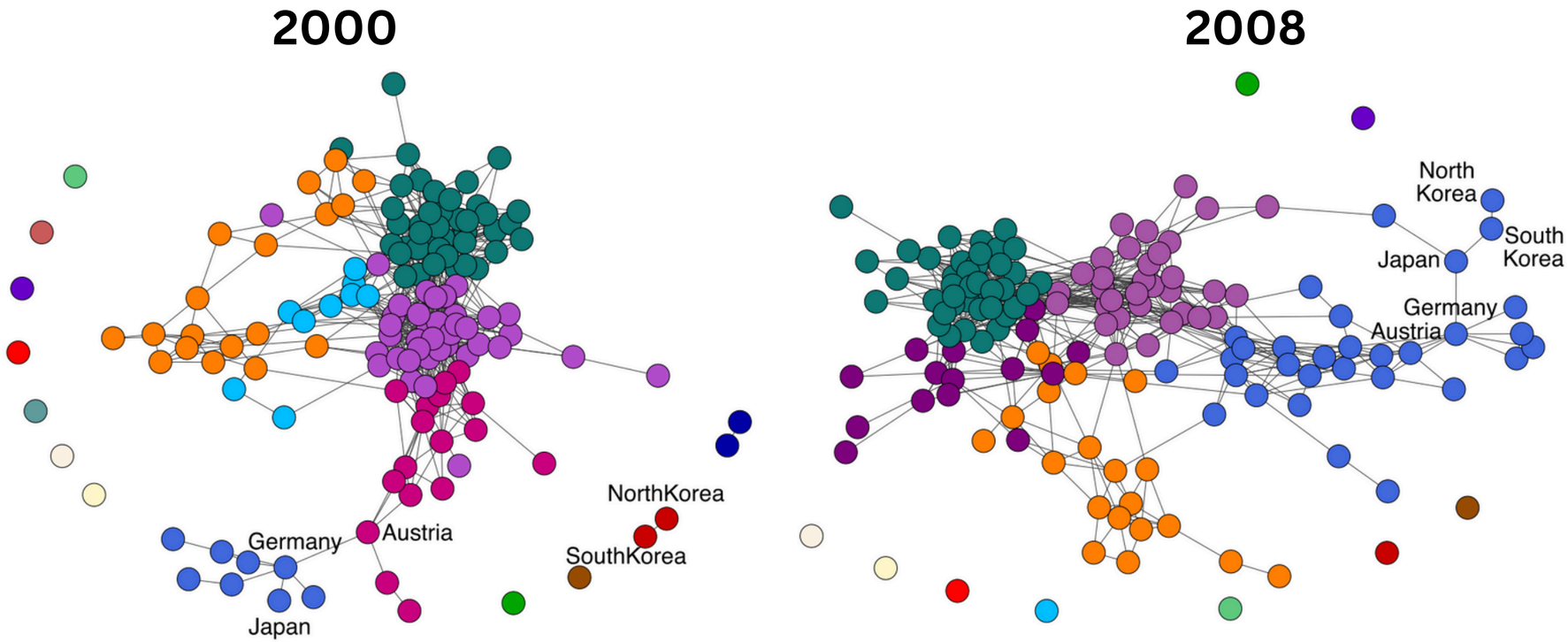

# Naive vs Validated Projection

We consider the case of countries and products

- we project the bipartite network to get the network of countries
- in the naive projection we previously introduced almost all countries are connected
- this is due to spurious cooccurrences
  - two countries that export many products will have many cooccurrences just by chance
- instead the BiCM validated projection returns a much more meaningful structure

# The Country Network

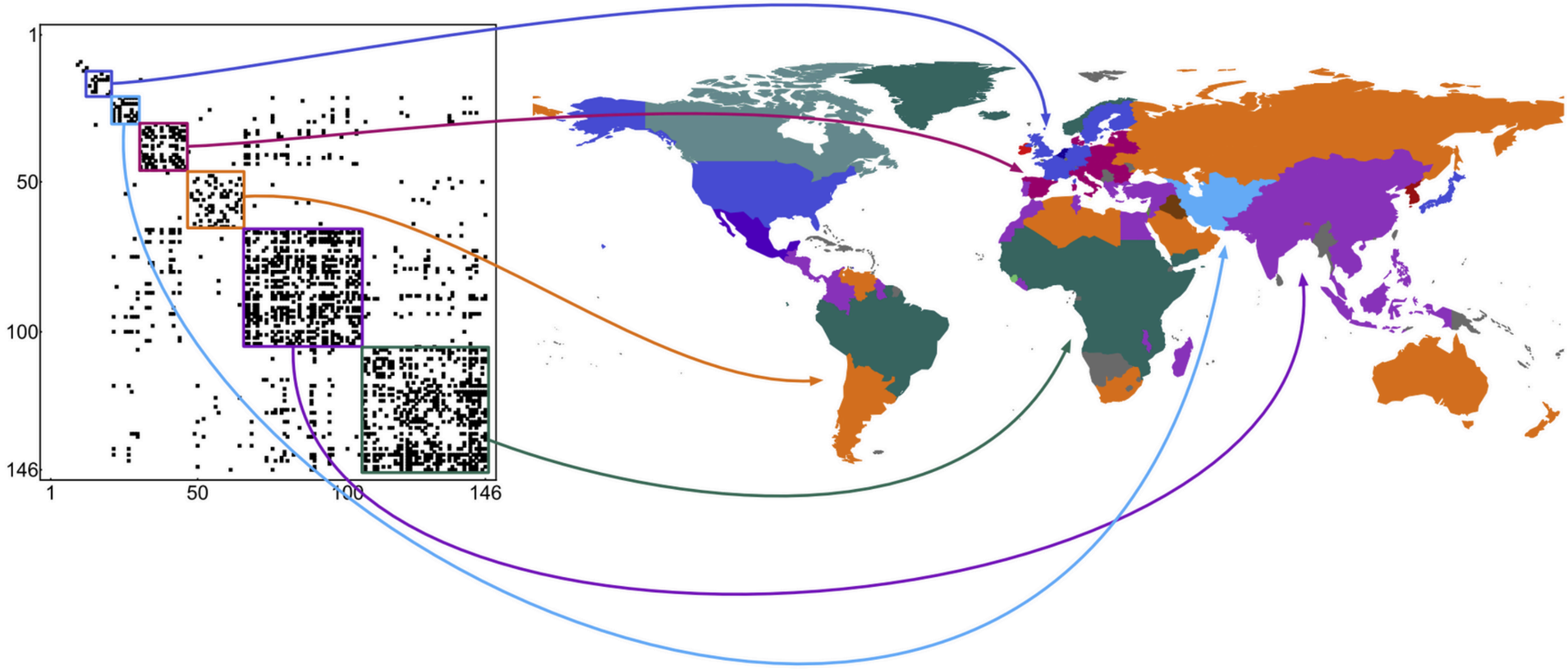
The BiCM validated projection returns a country network with a well defined structure and a low density of links. This is because only statistically significant connections are retained




Saracco, Fabio, et al. "*Inferring monopartite projections of bipartite networks: an entropy-based approach*." New Journal of Physics 19.5 (2017): 053022.

# Detecting Communities

The network presents a clear network structure with communities corresponding for instance to tropical countries or advanced economics

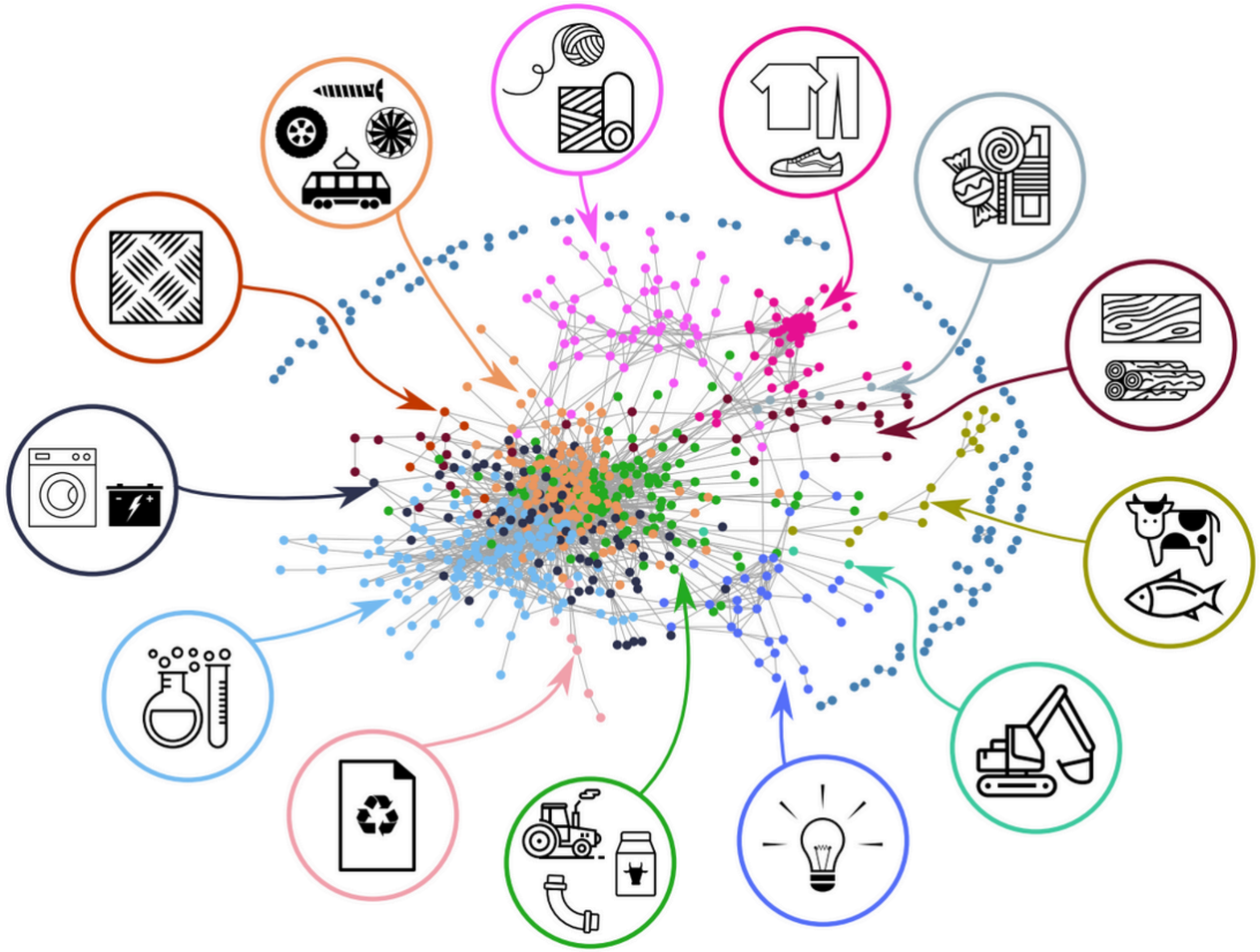


# The Product Network

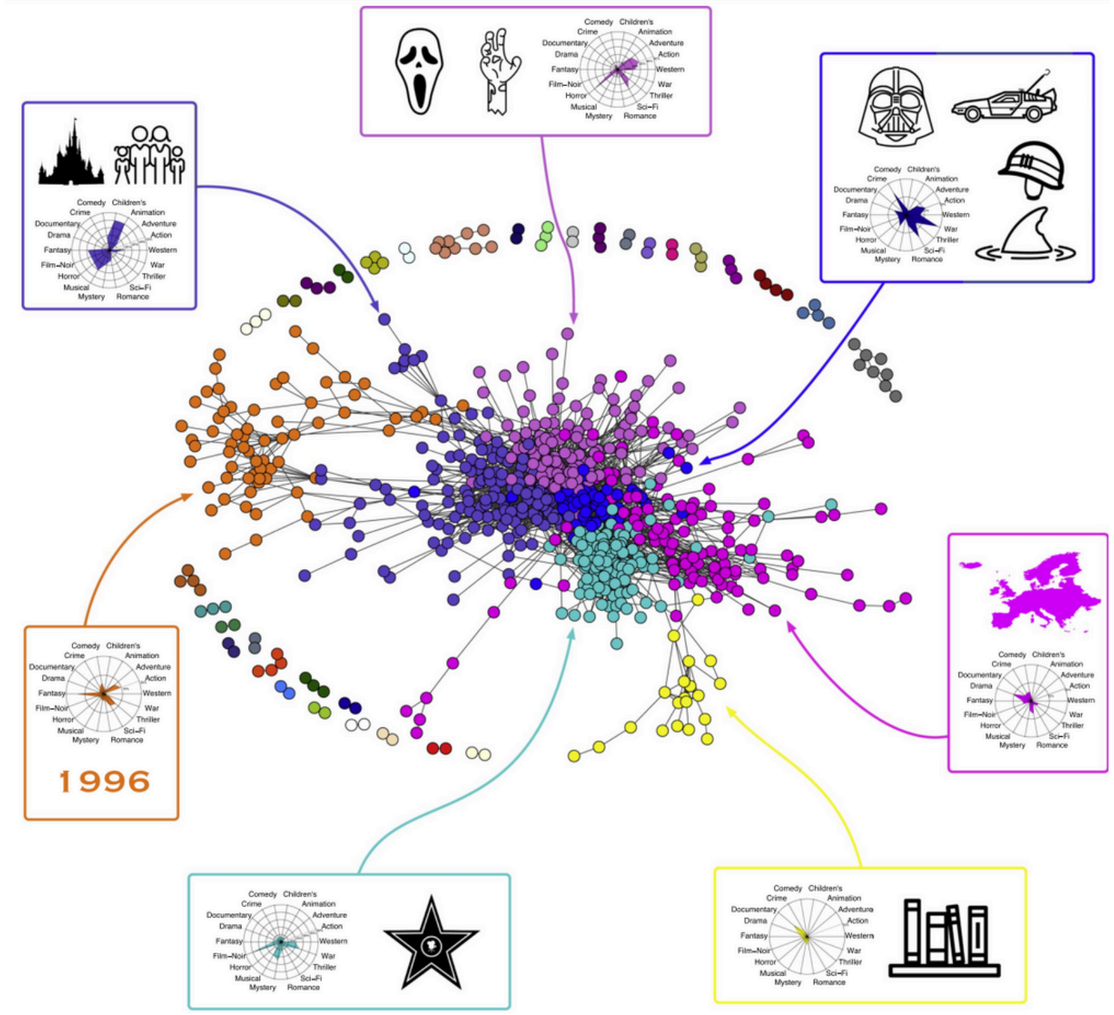
The same procedure can be applied also to products

- this time we look the other way around
- given two products, we see how many common countries export them
- we then validate the projection with the BiCM

Also in this case we obtain a network of product with a well defined community structure



# MovieLens Dataset




MovieLens is a dataset of movies rating

- it can be seen as a bipartite network
  - we have users on one layer
  - movies on the other layer
- we can get a biadjacency matrix by binarizing the ratings
  - $M_{ij}$=1 if user i rated movie j 3 out of 5 or more

Starting from the bipartite network we can then perform network projection

- for instance we can get the network of movies
- we observe also in this case a clear community structure

# Conclusions

## Null Models for Networks

Null models are crucial to validate the properties of networks. Examples include the ER graph and the configuration model (fixed degree sequence)

## Network Ensembles

An ensemble is the set of all possible graphs with a given macroscopic property. We showed how to get a (canonical) ensemble of networks given some constraints that we want to satisfy on average

## Applications of Null Models

Null model can be used for validating communities or other properties, but also to reconstruct networks from partial information. This is very relevant in finance.

## Bipartite Networks Projection

Null models, such as the BiCM, allow to obtain statistically validated projection of bipartite networks, performing much better than naive approaches