Decoding Connections: Workshop in Network Data Analysis

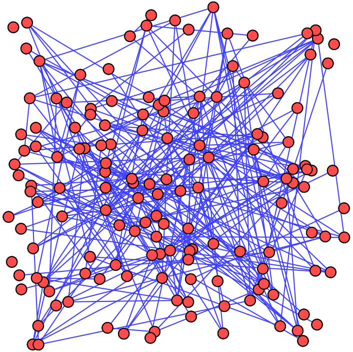
Beatrice Franzolini[1]
[1]Bocconi Institute for Data Science and Analytics
Bocconi University

June 8, 2025 - SICSS-Lake Como

Contents

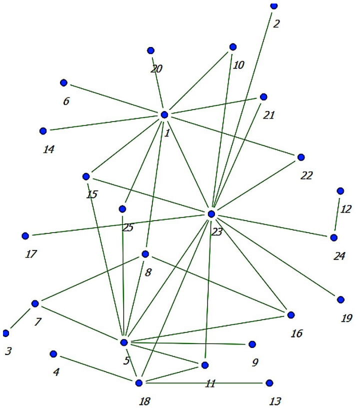
Introduction to Network Data

Network Data



- Network data describe the connections (**edges**/**links**) among a set of entities (**nodes** / **vertices**), showing who or what is connected to whom or what.
- Because those connections create interdependencies and define a specific structure (different than most datasets), we need **specialized statistical techniques** to make sense of the patterns and analyze networks.

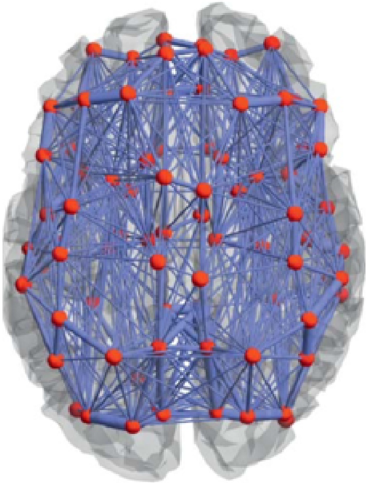
Example 1:
Wiretapping Network of Drug Dealing in Colombia[1]



(a) Acero Drug Network

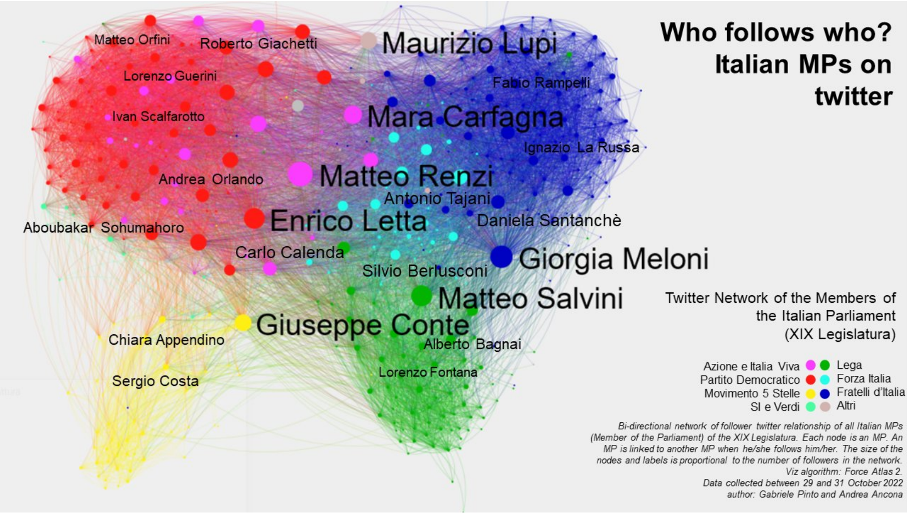
[1] Kaustav Basu, Arunabha Sen, Identifying individuals associated with organized criminal networks: A social network analysis, Social Networks, Volume 64, 2021, Pages 42-54,

Example 2: DTI Networks [2]



[2] J. Cabral , M. L. Kringelbach , G. Deco, Functional Graph Alterations in Schizophrenia: A Result from a Global Anatomic Decoupling? Pharmacopsychiatry 2012; 45(S 01): S57-S64

Example 3: X Network of Italian Members of Parliament



Examples of Network Data

- **Social networks:** individuals connected by “friendship" or interactions (e.g., likes, DMs).
- **Information networks:** webpages linked by hyperlinks; citation networks
- **Physical networks:** transportation and infrastructure networks
- **Biological/medical networks:** protein–protein interaction; neural connectomes
- **Organizational networks:** co-authorship; trade partnerships
- **A good resource with many network data**: `https://networkrepository.com/index.php`

Examples of Research Questions Related to Network Data

- **Influence & centrality:** Who are the most central/influential actors?
- **Community detection:** How to uncover cohesive subgroups? How to detect groups of subjects that behave similarly within the network?
- **Structure–outcome relations:** How are the network connections influenced by a set of available covariates?
- **Evolution of ties:** Do links form by preferential attachment or homophily?
- **Robustness & intervention:** What happens if key nodes are removed?

A Formal Representation of a Network

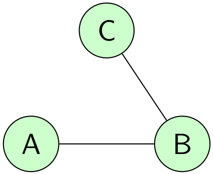
In order to analyze network data, we need first a way to represent them formally!

Network data are represented by graphs.

A *graph* G is an ordered pair $G = (V, E)$ where:

- V is a set of n vertices (nodes).
- $E \subset \{\{u, v\} : u, v \in V, u \neq v\}$ is a set of unordered, distinct pairs.

Notation: $|V| = n, \quad |E| = m, \quad d_i = \deg(i) = \#\{j : \{i, j\} \in E\}$.



$V = \{A, B, C\}$ $\quad |V| = n = 3$

$E = \{\{A, B\}, \{B, C\}\}$ $\quad |E| = m = 2$

$d_A = 1\,,\ d_B = 2\,,\ d_C = 1$

Undirected vs Directed networks

Undirected Edges are unordered pairs $\{u, v\}$; mutual relation (e.g., friendship).

Directed Edges are ordered pairs (u, v); asymmetric relation (e.g., Twitter follow).

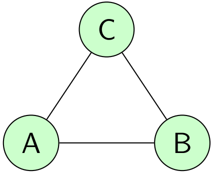


Figure: Undirected network

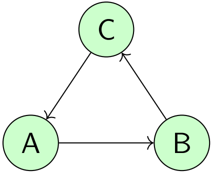


Figure: Directed network

In directed graphs, $\deg^{\text{in}}(i) \neq \deg^{\text{out}}(i)$.

Simple vs Multi graphs

- **Simple graph:** at most one edge per node-pair, no loops.
- **Multigraph:** allows parallel edges and self-loops.

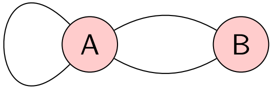


Figure: Multigraph example

Networks with Attributes

- **Node attributes:** categories, covariates (e.g., gender, age).
- **Edge weights:** tie strength (e.g., number of emails).
- **Signed networks:** positive/negative ties (e.g., like vs dislike on YouTube).

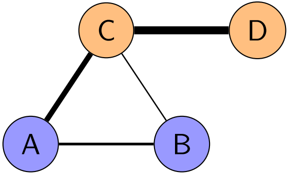


Figure: Attributed network example, colors represent node attributes (node-colored network), thickness of the edges represents edge weights

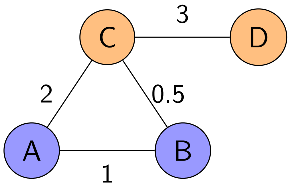
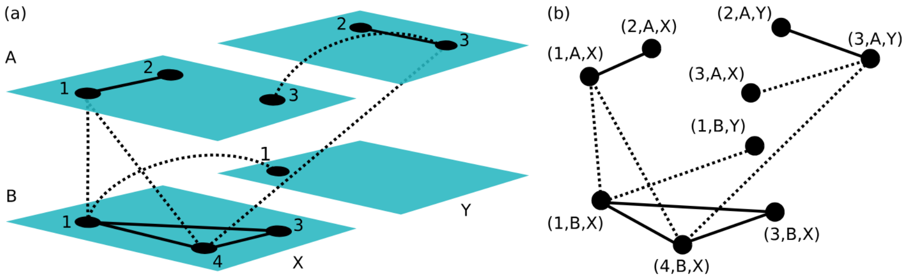


Figure: Attributed network example, colors represent node attributes (node-colored network), numbers on the edges represents edge weights

And actually... many others[3]



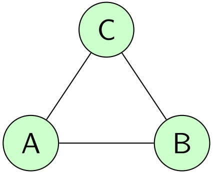
[3] Kivelä, M., Arenas, A., Barthelemy, M., Gleeson, J. P., Moreno, Y., & Porter, M. A. (2014). Multilayer networks. Journal of complex networks, 2(3), 203-271.

How to Encode Network Data

Adjacency List

- For each node, list neighbors (feasible for sparse or small graphs).
- Example (triangle A–B–C):

$$A : [B, C],\ B : [A, C],\ C : [A, B]$$



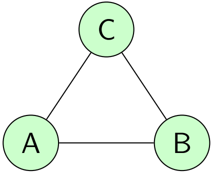
R / igraph:

```
library(igraph)
edges <- data.frame(from=c("A","A","B"), to=c("B", "C", "C"))
g <- graph_from_data_frame(edges,dir=FALSE)
adj_list <- adjacent_vertices(g, V(g))
print(adj)list)
```

Edge List

- Two-column table of edges.
- Example (triangle A–B–C):

$$\begin{matrix} A & B \\ A & C \\ B & C \end{matrix}$$



R / igraph:

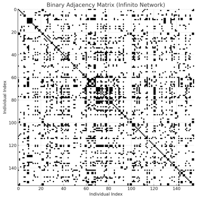
```
as_edgelist(g)
```

Adjacency Matrix

- $n \times n$ matrix A with $A_{ij} = 1$ if edge exists, else 0.
- Symmetric for undirected; memory $O(n^2)$.

$$A = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix}$$

A real data example: Infinito network [a]

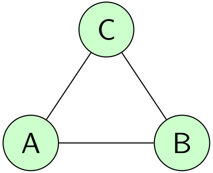


[a] Calderoni, F., & Piccardi, C. (2014). Uncovering the structure of criminal organizations by community analysis: The infinito network. In 2014 tenth international conference on signal-image technology and Internet-based systems (pp. 301-308). IEEE.

R / igraph:

```
as_adjacency_matrix(g)
```

Summing Up: Encoding



Adjacency Matrix

$$A = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix}$$

Edge List

```
A    B
A    C
B    C
```

Adjacency List

```
A: [B, C]
B: [A, C]
C: [A, B]
```

Descriptive Analysis of Networks

Density of a Network

Density

$$\text{Density} = \frac{m}{\binom{n}{2}} = \frac{2\,m}{n(n-1)} \quad (0 \leq \text{Density} \leq 1)$$

where m = num. of edges and n = num. of nodes

- **What it measures:** Fraction of realized edges out of all possible $\binom{n}{2}$.
  Density $\approx 0 \implies$ very sparse; Density $\approx 1 \implies$ almost complete.
- **Special cases:**
  - Complete graph K_n: $m = \frac{n(n-1)}{2} \implies$ Density $= 1$.
  - Tree on n nodes: $m = n - 1 \implies$ Density $= \frac{2}{n}$, which vanishes as n grows.
- **Why use it:**
  - Compare overall connectivity across networks of different sizes.
  - Quick sanity check (e.g. is my statistical model generating graphs that are too sparse?).

In R / igraph: `edge_density(g)`

Vertex Degree

Undirected network. The degree of node i, denoted

$$d_i = \sum_j A_{ij},$$

is the number of edges incident on i, where the adjacency matrix entry

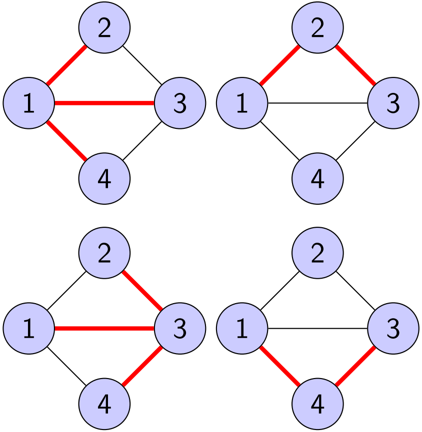
$$A_{ij} = \begin{cases} 1, & \text{if there is an (undirected) edge between } i \text{ and } j, \\ 0, & \text{otherwise.} \end{cases}$$

Directed network.

$$d_i^{\text{in}} = \sum_j A_{ji}, \qquad d_i^{\text{out}} = \sum_j A_{ij}.$$

Incoming degree counts arrows into i; outgoing degree counts arrows out.

Vertex Degree Distribution



Degree distribution.

Let n be the number of nodes. Then

$$P(k) = \frac{\#\{\, i : d_i = k \}}{n}$$

is the fraction of nodes of degree k.

In our example $(d_1, d_2, d_3, d_4) = (3, 2, 3, 2)$, so

$$P(2) = \frac{2}{4} = 0.5, \quad P(3) = \frac{2}{4} = 0.5.$$

R / igraph: `degree(g)`

Vertex Centrality

Many network-analysis questions boil down to:

- Which nodes are *most important* in the network?

Research questions examples:

- "What airports are key bottlenecks in transportation?"
- "Who should we vaccinate first to stop an epidemic most efficiently?"
- "Which employee's departure would fragment the organization most?"
- "Which web pages serve as gateways to the broader Internet?"
- "Which user in a social media network has the greatest influence potential?"

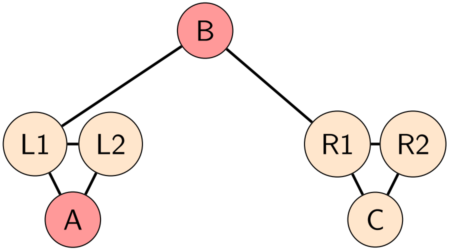
Centrality measures answer these questions = quantify different notions of "importance".

Degree centrality (i.e., node degree)

$$C_D(i) = d_i,$$

Limitations of Degree Centrality

Two nodes with the same degree but very different roles



- **Degree centrality:** $d_A = 2$ and $d_B = 2$.
- **Node A:** lies entirely within one cluster – its removal *does not* disconnect the network.
- **Node B:** is the *only* bridge between two clusters – its removal *splits* the network into two disconnected parts.
- **Takeaway:** Node degree is just local popularity. Nodes with equal degree can play very different global roles. *Degree centrality alone* can miss critical structural importance.

We need different measures of centrality!

Shortest Path

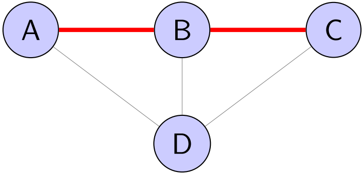
Path and Shortest Path

In an *unweighted* graph $G = (V, E)$, a *path* from node u to node v is a sequence of distinct vertices

$$u = x_0, x_1, \ldots, x_k = v \quad \text{with} \quad (x_{i-1}, x_i) \in E \text{ for } i = 1, \ldots, k.$$

The *length* of such a path is simply the number of edges, k.
A *shortest path* between u and v is one having the minimum possible k.

Notation: $d(u, v) = \min\{\, k : \exists \text{ a path of length } k \text{ from } u \text{ to } v\}$.



Here, $d(A, C) = 2$, since the minimum number of hops from A to C is two (via B).

Vertex Centrality: Definitions

Betweenness centrality

$$C_B(i) = \sum_{s \neq t \neq i} \frac{\sigma_{st}(i)}{\sigma_{st}},$$

where σ_{st} is the number of shortest paths $s \to t$, and $\sigma_{st}(i)$ counts those that pass through i. Measures how much i “bridges” pairs of nodes.

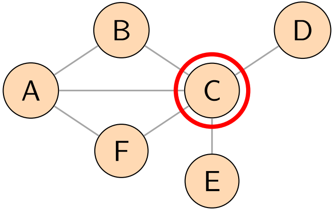
Closeness centrality

$$C_C(i) = \frac{1}{\sum_j d(i,j)},$$

with $d(i,j)$ the shortest-path distance. Quantifies how quickly i can reach all others.

In R: `betweenness(g)`, `closeness(g)`

Vertex Centrality: Toy Network Example



- **Degree centrality**:

$$C_D(A) = 3,\ C_D(B) = 2,\ C_D(C) = 5,\ C_D(D) = 1,\ C_D(E) = 1,\ C_D(F) = 2.$$

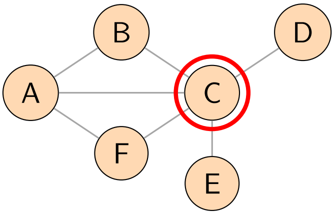
- **Betweenness centrality**:

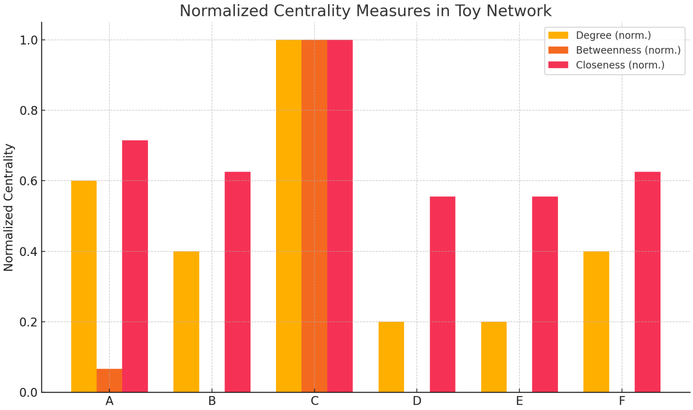
$$C_B(A) = 1,\ C_B(B) = 0,\ C_B(C) = 8,\ C_B(D) = 0,\ C_B(E) = 0,\ C_B(F) = 0.$$

- **Closeness centrality**:

$$C_C(A) = \tfrac{1}{7} \approx 0.143, \quad C_C(B) = C_C(F) = \tfrac{1}{8} = 0.125,$$
$$C_C(C) = \tfrac{1}{5} = 0.200, \quad C_C(D) = C_C(E) = \tfrac{1}{9} \approx 0.111\,.$$

Vertex Centrality: Toy Network Example



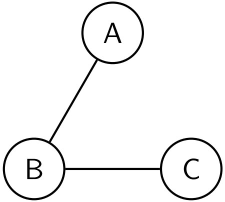


Transitivity in Networks

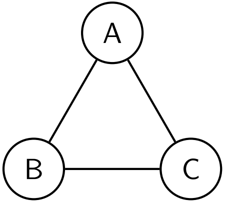
What is transitivity?

- Intuitively: "a friend of a friend is likely also my friend."
- More formally: if edges $A{-}B$ and $B{-}C$ exist, how often do we also see $A{-}C$?
- Captures the tendency toward *closure* and local cohesion in real-world networks.
- High transitivity $\implies$ strong community structure.

Basic patterns to measure transitivity:



Open triad
(two-path without closure)



Closed triad = *Triangle*
(fully connected triple)

There are three distinct triads in a triangle!

Local & Average Clustering Coefficients

Local and Average clustering coefficients

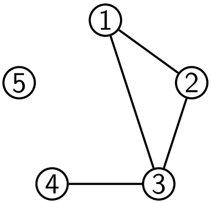
- **Local clustering coefficient:**

$$C(v) = \frac{\text{num. of couples of ``friends" of } v \text{ that are ``friends"}}{\text{num. of couples of ``friends" of } v}$$

$$= \begin{cases} \frac{\#\{\text{triangles containing } v\}}{\binom{\deg(v)}{2}}, & \deg(v) \geq 2, \\ 0, & \text{otherwise.} \end{cases}$$

- **Average clustering coefficient:** $\bar{C} = \frac{1}{n} \sum_{v \in V} C(v)$.

Example:



$C(1) = ?, \quad C(2) = ?, \quad C(3) = ?,$

$C(4) = ?, \quad C(5) = ?,$

$\bar{C} = ?.$

Local & Average Clustering Coefficients

Local and Average clustering coefficients

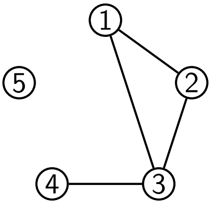
- **Local clustering coefficient:**

$$C(v) = \frac{\text{num. of couples of "friends" of } v \text{ that are "friends"}}{\text{num. of couples of "friends" of } v}$$

$$= \begin{cases} \dfrac{\#\{\text{triangles containing } v\}}{\binom{\deg(v)}{2}}, & \deg(v) \geq 2, \\ 0, & \text{otherwise.} \end{cases}$$

- **Average clustering coefficient:** $\bar{C} = \frac{1}{n} \sum_{v \in V} C(v)$.

Example:



$C(1) = 1, \quad C(2) = 1, \quad C(3) = \frac{1}{3},$

$C(4) = 0, \quad C(5) = 0,$

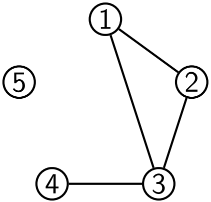
$$\bar{C} = \frac{1 + 1 + \frac{1}{3} + 0 + 0}{5} \approx 0.467.$$

Global Clustering Coefficient

Global clustering coefficient

$$C_{\text{global}} = \frac{3 \times \text{num. of triangles}}{\text{num. of triads}}$$

Example:



number of triangles $= ?$

number of triads $= ?$

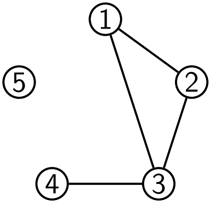
$C_{\text{global}} = ?$

Global Clustering Coefficient

Global clustering coefficient

$$C_{\text{global}} = \frac{3 \times \text{num. of triangles}}{\text{num. of triads}}$$

Example:

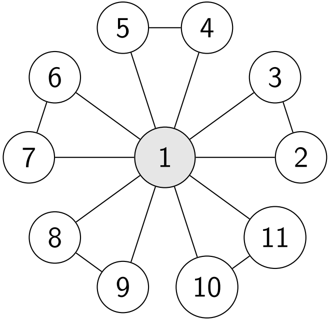


number of triangles $= 1$

number of triads $= 5$

$C_{\text{global}} = 3/5 = 0.6$

Local vs Global Transitivity



$$C_{\text{local}}(1) = \frac{\#\{\text{triangles at } 1\}}{\binom{10}{2}} = \frac{5}{45} = \frac{1}{9} \approx 0.111$$

$$C_{\text{local}}(i) = \frac{1}{\binom{2}{2}} = 1, \quad i = 2, \ldots, 11$$

$$\bar{C} = \frac{1}{11} \sum_{v=1}^{11} C_{\text{local}}(v) = \frac{\frac{1}{9} + 10 \cdot 1}{11} = \frac{91}{99} \approx 0.919$$

$$C_{\text{global}} = \frac{3 \times (\#\text{triangles} = 5)}{\sum_v \binom{d_v}{2}} = \frac{15}{55} = \frac{3}{11} \approx 0.273$$

Summing Up: Descriptive Statistics

- **General measures structure and size**

Density:	how much connected is the network. (in [0,1])	`edge_density(g)`
Degree dist:	how many nodes with x connections.	`table(degree(g))`

- **Measures of centrality**

Degree:	how many connections with node v	`degree(g)`
Betweenness:	how often v connects others	`betweenness(g)`
Closeness:	how close if v to others	`closeness(g)`

- **Measures of transitivity**

Average clustering:	local transitivity	`transitivity(g, type="local")`
Global clustering:	global transitivity	`transitivity(g, type="global")`

Simple Probabilistic Models

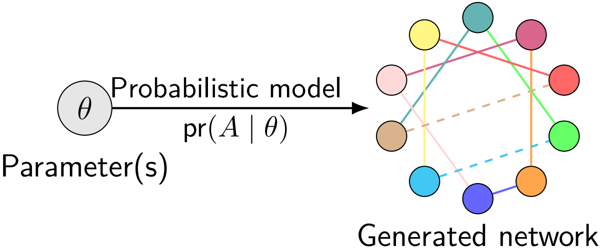
Probabilistic Generative Network Models

Given the observed set of nodes V, we can probabilistically model the network by assuming some ***distribution*** generating the links between them, i.e., define the distribution of the adjacency matrix A:

$$\mathrm{pr}(A \mid \theta)$$

Then:

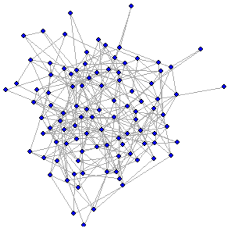
- Estimate θ from the observed network.
- Predict links for *new* nodes.

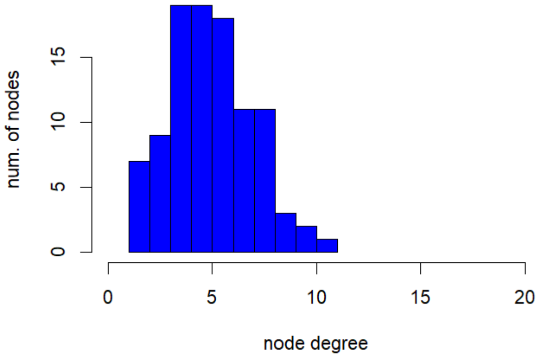


Erdős–Rényi Random Graph[4]

Each pair of the n vertices is connected with **probability** p independently.

$$A_{u,v} \mid p \overset{\text{iid}}{\sim} \text{Bernoulli}(p) \quad \forall u < v$$





- $\mathbb{E}[m] = \binom{n}{2} p$
- $d_i \sim \text{Bin}(n-1, p)$.

[4] Erdős & Rényi (1959). "On Random Graphs. I" Publicationes Mathematicae. 6 (3–4): 290–297.

Preferential Attachment Models[5]

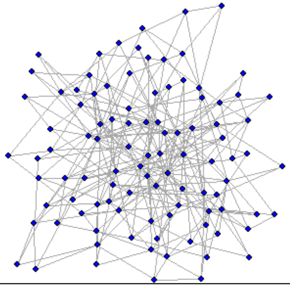
i. There is a fixed **initial network** G_{m_0} with $2 \leq m_0 << m$ nodes

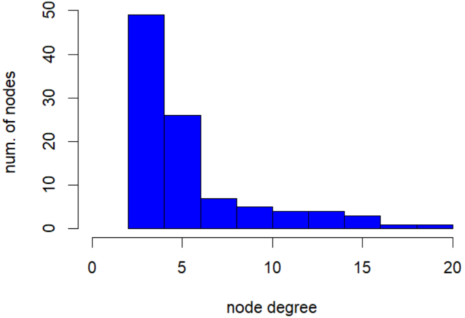
ii. A new node v enters the network and creates m_0 **links** with the existing nodes sampling them according to

$$\text{pr}(A_{v,u} = 1) \underset{u}{\propto} f(\theta, d_u)$$

with f increasing function in d_u depending on **the parameter** θ.

iii. Step ii. is repeated until all n nodes are in the network.





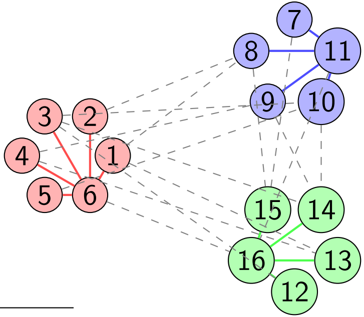
[5] Barabási & Albert (1999). "Emergence of scaling in random networks". Science. 286 (5439): 509–512.

Stochastic Block Models

Stochastic Block Model (SBM)[6] Overview

- A generative model for networks: n nodes are partitioned into K latent blocks.
- Each node i is assigned to a community $z_i \in \{1, \dots, K\}$ (unknown labels).
- Edge probabilities depend only on the communities:

$$\text{pr}(A_{ij} = 1 \mid z_i = k,\, z_j = \ell) = \theta_{k\ell}, \quad \forall i > j.$$



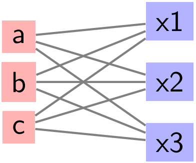
[6] Holland, P. W., Laskey, K. B., & Leinhardt, S. (1983). Stochastic blockmodels: First steps. *Social Networks*, 5(2), 109–137.

Blocks vs. Communities

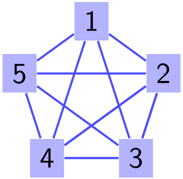
Blocks in SBM
Nodes share the same *connectivity patterns*, i.e., they behave similarly, but they are not necessarily connected among themselves.

i, j in same block means

$$\text{pr}(A_{i\cdot} \mid z_i = k) \approx \text{pr}(A_{j\cdot} \mid z_j = k).$$



Community
Usually refers to a subset of nodes that form a *densely connected* subgraph:

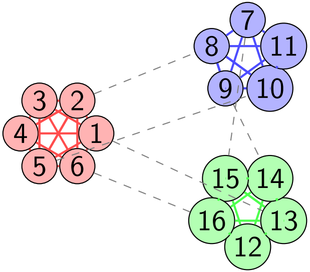


- **Block:** A set of nodes with *equivalent linking profiles* to all blocks.
- **Community:** A cluster with *high internal density* of edges.

Assortativity in Stochastic Block Models

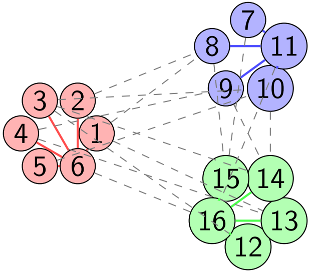
Assortative SBM

Dense intra-block connectivity, few inter-block links.



Non-assortative SBM

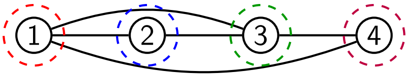
Sparse intra-block connectivity, relatively more inter-block links.



- **Assortative:** High probability of edges within blocks ($\theta_{kk} \gg \theta_{k\ell}$), reflecting strong community structure.
- **Non-assortative:** Low within-block edge probability ($\theta_{kk} \leq \theta_{k\ell}$), showing disassortative or core–periphery patterns.

Inference in SBMs

- Two sets of unknowns:
  - *Block assignments* $\mathbf{z} = (z_1, \ldots, z_n)$, discrete labels $z_i \in \{1, \ldots, K\}$.
  - *Connection probabilities* $\Theta = (\theta_{k\ell})_{k \leq \ell \leq K}$, continuous parameters.
- **Frequentist MLE:** $(\hat{z}, \hat{\Theta}) = \arg\max_{z,\Theta} \sum_{i<j} \left[A_{ij} \log \theta_{z_i,z_j} + (1 - A_{ij}) \log(1 - \theta_{z_i,z_j})\right]$.
- **Degenerate MLE if K free:**
  Allowing arbitrary K gives the trivial solution: $K = n, \quad z_i = i, \quad \hat{\theta}_{ij} = A_{ij}$,
  i.e. each node in its own block, perfectly fitting every edge.



Take-away: To avoid this, one must

- Fit SBMs for different fixed K and compare fitting via information criteria,
- Use a *Bayesian nonparametric* approach / impose complexity penalties.

Frequentist SBM: Estimation

ML objective: $$(\hat{z}, \hat{\Theta}) = \arg\max_{z,\Theta} \sum_{i<j} \Big[A_{ij} \log \theta_{z_i,z_j} + (1 - A_{ij}) \log(1 - \theta_{z_i,z_j}) \Big]$$

E–step: $$\gamma_{ik} \propto \pi_k \prod_{j \neq i} \theta_{k,z_j}^{A_{ij}} (1 - \theta_{k,z_j})^{1-A_{ij}}$$

M–step: $$\pi_k \leftarrow \tfrac{1}{n} \sum_i \gamma_{ik}, \quad \theta_{k\ell} \leftarrow \frac{\sum_{i<j} \gamma_{ik}\gamma_{j\ell} A_{ij}}{\sum_{i<j} \gamma_{ik}\gamma_{j\ell}}$$

ML and EM require fixing K.

Integration classification likelihood criterion:

$$\mathrm{ICL}(K) = -2\,\ell(\hat{z}, \hat{\Theta}) + \left[\tfrac{1}{2} K(K+1)\right] \log\left(\binom{n}{2}\right)$$

$\mathrm{ICL}(K)$ often has a clear minimum but must be computed for each K.

The Bayesian Paradigm (Informal)

- **What is Bayesian inference?**
  A way to learn about unknown quantities by *updating* knowledge with observed data.
- **Bayes' Rule:** $\underbrace{\text{pr}(\theta \mid \text{data})}_{\text{posterior}} \propto \underbrace{\text{pr}(\text{data} \mid \theta)}_{\text{likelihood}} \times \underbrace{\text{pr}(\theta)}_{\text{prior}}.$
  - *Prior* $p(\theta)$: what you believe about θ *before* seeing the data.
  - *Likelihood* $p(\text{data} \mid \theta)$: how probable the observed data are, given θ.
  - *Posterior* $p(\theta \mid \text{data})$: your updated belief after seeing the data.
- **Key ideas / Why use it?:**
  - *Full probabilistic*: posterior is a full distribution, not just a point estimate.
  - *Regularization*: the prior can shrink or penalize extreme estimates (avoids overfitting).
  - *Modularity*: easy to build hierarchical and complex models by stacking priors.
  - *Integration of sources*: provides a coherent framework for combining data with existing knowledge or info from different data sources.

A Bayesian Nonparametric Approach to SBM

- **Bayesian paradigm:** Place priors on both block assignments and connection probabilities, then infer the posterior

$$p(\mathbf{z}, \Theta \mid A) \;\propto\; p(A \mid \mathbf{z}, \Theta)\, p(\Theta)\, p(\mathbf{z}).$$

  This naturally penalizes over-complex partitions (avoiding $K = n$ degeneracy).
- **Nonparametric:** Number of blocks K need not be fixed in advance – it can grow with the data.
- **Priors:**
  - *Partition prior* $p(\mathbf{z})$: Chinese Restaurant Process (CRP) with concentration α.
  - *Edge-probability prior* $p(\Theta)$: i.i.d. $\mathrm{Beta}(\beta, \beta)$ for each $\theta_{k\ell}$.
- **Key benefit:** Let the data "decide" how many blocks are needed, trading off fit vs. complexity.

Infinite SBM Generative Model

$$
\begin{aligned}
&z_i \sim \mathrm{CRP}(\alpha), \quad i = 1, \ldots, N, \\
&\theta_{k\ell} \sim \mathrm{Beta}(\beta, \beta), \quad \forall k \leq \ell, \\
&A_{ij} \mid z_i = k, z_j = \ell, \Theta \ \sim\ \mathrm{Bernoulli}(\theta_{k\ell}), \quad A_{ji} = A_{ij}.
\end{aligned}
$$

- α controls the tendency to create new blocks: small α favors fewer, larger clusters; large α allows many small clusters.
- β encodes prior belief on connection sparsity; $\beta = 1$ gives a uniform prior on $[0, 1]$.

The Chinese Restaurant Process

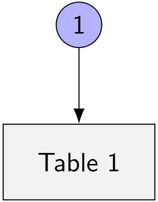
- **Seating metaphor:** Customers (nodes) enter one by one into a restaurant with infinitely many tables.
- **Assignment rule for customer i:**

$$\mathrm{pr}(z_i = k \mid z_1, z_2, \ldots, z_{i-1}) = \begin{cases} \dfrac{n_k}{i-1+\alpha}, & \text{existing table } k, \\ \dfrac{\alpha}{i-1+\alpha}, & \text{new table}, \end{cases}$$

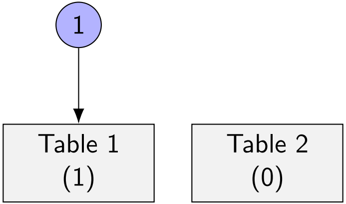
  where n_k is the current size of table k.
- **Properties:**
  - Expected number of tables $\approx \alpha \log N$.
  - Equivalent to a Dirichlet Process.

CRP Step 1: Customer 1

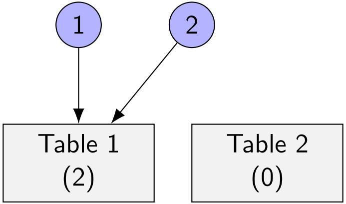
$$P(z_1 = 1) \; = \; 1$$



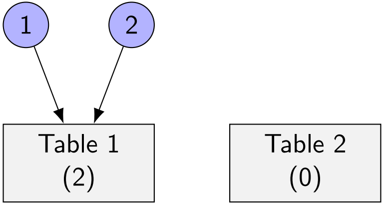
CRP Step 2: Customer 2



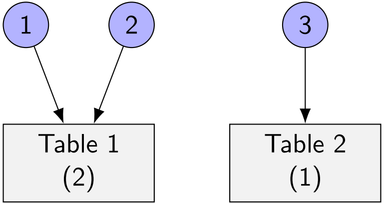
$$P(z_2 = 1 \mid z_1) = \frac{1}{1+\alpha}, \quad P(z_2 = \text{new} \mid z_1) = \frac{\alpha}{1+\alpha}$$



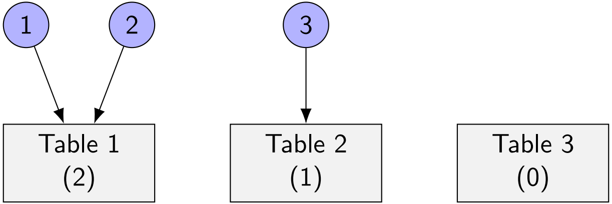
CRP Step 3: Customer 3



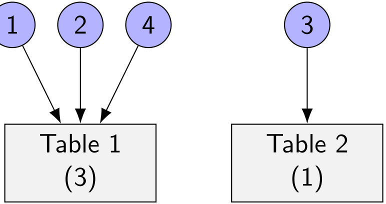
$$P(z_3 = 1 \mid z_{1:2}) = \frac{2}{2+\alpha}, \quad P(z_3 = 2 \mid z_{1:2}) = \frac{\alpha}{2+\alpha}$$



CRP Step 4: Customer 4



$$P(z_4 = 1 \mid z_{1:3}) = \frac{2}{3+\alpha}, \quad P(z_4 = 2 \mid z_{1:3}) = \frac{1}{3+\alpha}, \quad P(z_4 = \text{new}) = \frac{\alpha}{3+\alpha}$$



Bayesian Nonparametric SBM: Estimation

Gibbs sampler

1. **Initialize** assignments $z^{(0)}$ (e.g. randomly).
2. **Iterate for** $t = 1, \ldots, T$**:**
   - For each node i:
     2.1 *Remove* i from its current block, updating counts n_k^{-i}.
     2.2 *Compute* for each existing block k:
     $$p_k \;\propto\; n_k^{-i} \times \Pr(A_{i,\cdot} \mid z_i = k, z_{-i}),$$
     and for a *new* block:
     $$p_{\text{new}} \;\propto\; \alpha \times \Pr(A_{i,\cdot} \mid \text{new block}).$$
     2.3 *Sample* $z_i^{(t)}$ from $\{p_k, p_{\text{new}}\}$.
   - (Optional) *Sample* $\theta_{k\ell} \sim \mathrm{Beta}(\beta + m_{k\ell},\, \beta + t_{k\ell} - m_{k\ell})$.
3. **Output:** Posterior samples $\{z^{(t)}, \Theta^{(t)}\}$.

Summing Up: SBMs

- **Generative view:**

$$A_{ij} \mid z_i = k,\ z_j = \ell, \Theta \ \sim\ \text{Bernoulli}(\theta_{k\ell}).$$

- **Parameters to infer:**
  - Block assignments $\{z_i\}_{i=1}^{n}$
  - Connection matrix $\Theta = (\theta_{k\ell})_{k,\ell=1}^{K}$
  - Number of blocks K
- **Inference strategies:**

  Frequentist EM / MLE with fixed $K \xrightarrow{\text{ICL}}$ select K

  Bayesian Gibbs, place CRP prior on z, Beta prior on $\theta_{k\ell}$
- **Trade-offs:**
  - Fixed-K SBM: faster, needs external model selection
  - CRP: automatic K discovery, quantifies uncertainty, higher computational cost

The End

Thank you for listening!

Questions?