Universitat Pompeu Fabra
*Barcelona*

# Lecture 7 · Random Networks I

## Networks, Crowds and Markets

# Today's Lecture

1. Wrapping-up centrality measures: PageRank and HITS.
2. Random graphs, Erdős–Rényi model.
3. Probability recap: binomial and Poisson distribution.
4. Probability recap: Chebyshev and Hoeffding inequality.
5. Degree distribution in Erdős–Rényi graphs.
6. Asymptotics in networks.
7. Threshold phenomena and giant component.

# Recall: PageRank

**Note**

We define random walk on a directed graph in a natural way. The walk can only follow the direction of arrows.

Algebraically more complicated as $A_G$ is not symmetric and the eigenvalues are complex.

- Web graph = directed network of pages and hyperlinks.
- Eigenvector centrality does not work directly in directed graphs with sinks or disconnected components.
- PageRank modifies the random walk with teleportation:

$$P_\alpha = \alpha P + (1-\alpha)\frac{1}{N}\mathbf{1}\mathbf{1}^T,$$

  where $P$ is the transition matrix of the web, $\alpha \in (0,1)$.
- Stationary distribution of $P_\alpha$ = PageRank vector.

# Beyond PageRank: The HITS Algorithm

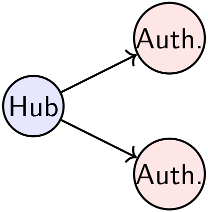
**Goal:** Identify both *authorities* and *hubs* in a directed network.

- A good hub points to many good authorities.
- A good authority is pointed to by many good hubs.

# Beyond PageRank: The HITS Algorithm

**Goal:** Identify both *authorities* and *hubs* in a directed network.

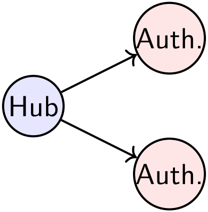
- A good hub points to many good authorities.
- A good authority is pointed to by many good hubs.



# Beyond PageRank: The HITS Algorithm

**Goal:** Identify both *authorities* and *hubs* in a directed network.

- A good hub points to many good authorities.
- A good authority is pointed to by many good hubs.




**Context:**

- Introduced by Jon Kleinberg (1999).
- Used originally to rank web pages within a *topic query*.
- Query-dependent — unlike PageRank, which is global.

# Mathematics of HITS

Let $A$ be the adjacency matrix ($A_{ij} = 1$ if $i \to j$).

Each node $i$ has: authority score $a_i$, hub score $h_i$.

# Mathematics of HITS

Let $A$ be the adjacency matrix ($A_{ij} = 1$ if $i \to j$).

Each node $i$ has: authority score $a_i$, hub score $h_i$.

They satisfy the mutual reinforcement relations:

$$\begin{cases} h \propto Aa, & \text{(hubs get votes from authorities)} \\ a \propto A^\top h, & \text{(authorities get votes from hubs)} \end{cases}$$

# Mathematics of HITS

Let $A$ be the adjacency matrix ($A_{ij} = 1$ if $i \to j$).

Each node $i$ has: authority score $a_i$, hub score $h_i$.

They satisfy the mutual reinforcement relations:

$$\begin{cases} h \propto Aa, & \text{(hubs get votes from authorities)} \\ a \propto A^\top h, & \text{(authorities get votes from hubs)} \end{cases}$$

Combining gives:

$$a \propto A^\top A a, \qquad h \propto A A^\top h.$$

# Mathematics of HITS

Let $A$ be the adjacency matrix ($A_{ij} = 1$ if $i \to j$).

Each node $i$ has: authority score $a_i$, hub score $h_i$.

They satisfy the mutual reinforcement relations:

$$\begin{cases} h \propto Aa, & \text{(hubs get votes from authorities)} \\ a \propto A^\top h, & \text{(authorities get votes from hubs)} \end{cases}$$

Combining gives:

$$a \propto A^\top A a, \qquad h \propto A A^\top h.$$

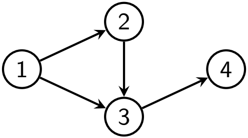
- Take $a$ and $h$ to be **dominant eigenvectors** of $A^\top A$ and $AA^\top$.
- In the iterative HITS algorithm, $a$ and $h$ are renormalized at each step, so the proportionality becomes equality after scaling.
- Equivalent viewpoint: HITS computes the first left and right singular vectors of $A$.

# Example: Hubs and Authorities in a Small Web

**Adjacency matrix:**

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

**Graph representation:**

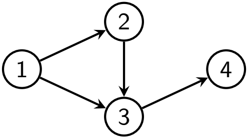


# Example: Hubs and Authorities in a Small Web

**Adjacency matrix:**

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

**Graph representation:**




**Iterative algorithm:**

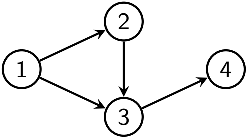
1. Initialize $a_i = h_i = 1$.
2. Repeat $a \leftarrow A^\top h$, normalize; $h \leftarrow Aa$, normalize.

# Example: Hubs and Authorities in a Small Web

**Adjacency matrix:**

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

**Graph representation:**




**Iterative algorithm:**

1. Initialize $a_i = h_i = 1$.
2. Repeat $a \leftarrow A^\top h$, normalize; $h \leftarrow Aa$, normalize.

**Python demo:**

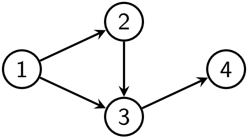
```
import networkx as nx
G = nx.DiGraph()
G.add_edges_from([(1,2),(1,3),(2,3),(3,4)])
hubs, auth = nx.hits(G)
```

# Example: Hubs and Authorities in a Small Web

**Adjacency matrix:**

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

**Graph representation:**




**Iterative algorithm:**

1. Initialize $a_i = h_i = 1$.
2. Repeat $a \leftarrow A^\top h$, normalize; $h \leftarrow Aa$, normalize.

**Python demo:**

```
import networkx as nx
G = nx.DiGraph()
G.add_edges_from([(1,2),(1,3),(2,3),(3,4)])
hubs, auth = nx.hits(G)
```

**Interpretation:**

- Node 1 $\rightarrow$ strong hub (points to many).
- Node 4 $\rightarrow$ strong authority (pointed to by many).

# Random graphs and Erdős–Rényi model

# Why random graphs?

Real networks (social, economic, financial) are noisy and constantly evolving. We need a simple *baseline model* to compare against.

**Definition ( Erdős–Rényi (ER) model )**

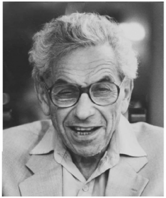
$G(N, p)$: a random graph on $N$ nodes where each of the $\binom{N}{2}$ possible edges appears independently with probability $p$.

# Why random graphs?

Real networks (social, economic, financial) are noisy and constantly evolving. We need a simple *baseline model* to compare against.

**Definition ( Erdős–Rényi (ER) model )**

$G(N, p)$: a random graph on $N$ nodes where each of the $\binom{N}{2}$ possible edges appears independently with probability $p$.

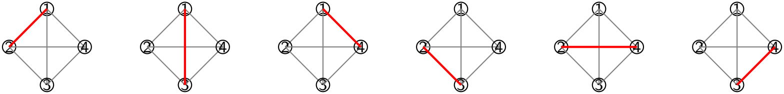


Paul Erdős (1913 - 1996)    Alfréd Rényi (1921-1970)

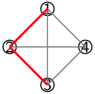
Erdős and Rényi (1959–60) launched the probabilistic study of graphs.

# $G(N, p)$ Model

Take $N = 4$ then the graph can have up to six edges. Each with distribution $\mathrm{Bern}(p)$:



12 13 14 23 24 34

$\mathbb{P}($  $) = p^2(1-p)^4$

If $p = \frac{1}{2}$, each graph appears with the same probability $\frac{1}{2^6} = \frac{1}{64}$.

# Probability recap: Binomial

**Definition**

If $X \sim \mathrm{Bin}(n, p)$ then

$$\mathbb{P}(X = k) = \binom{n}{k} p^k (1-p)^{n-k}, \quad \mathbb{E}[X] = np, \quad \mathrm{Var}(X) = np(1-p).$$

Useful characterization: $X = \sum_{i=1}^{n} Z_i$ with independent $Z_i \sim \mathrm{Bern}(p)$.

In the ER graph $G(N, p)$:

- Number of edges:

$$L \sim \mathrm{Bin}\left(\binom{N}{2}, p\right).$$

- Degree of a fixed vertex $v$:

$$\deg(v) \sim \mathrm{Bin}(N-1, p).$$

# Probability recap: Binomial

**Definition**

If $X \sim \mathrm{Bin}(n, p)$ then

$$\mathbb{P}(X = k) = \binom{n}{k} p^k (1-p)^{n-k}, \quad \mathbb{E}[X] = np, \quad \mathrm{Var}(X) = np(1-p).$$

Useful characterization: $X = \sum_{i=1}^{n} Z_i$ with independent $Z_i \sim \mathrm{Bern}(p)$.

In the ER graph $G(N, p)$:

- Number of edges:

$$L \ \sim \ \mathrm{Bin}\left(\binom{N}{2}, \, p\right).$$

- Degree of a fixed vertex $v$:

$$\deg(v) \ \sim \ \mathrm{Bin}(N-1, p).$$

# Probability recap: Poisson (as Binomial limit)

**Theorem**

If $X_n \sim \mathrm{Bin}(n, p_n)$ with $n \to \infty$ and $np_n \to \lambda > 0$, then

$$X_n \;\longrightarrow\; X \sim \mathrm{Pois}(\lambda), \qquad \mathbb{P}(X = k) = \frac{\lambda^k}{k!} e^{-\lambda}.$$

The approximation $\mathrm{Bin}(n, p) \approx \mathrm{Poiss}(\lambda)$ for $\lambda = pn$ is particularly good if $p$ is small.

**Example ( Quick check )**

For $n = 2000$, $p = 0.003$, $\lambda = np = 6$. Compare $\mathbb{P}(X = 0)$: Binomial $= (1 - p)^{2000} \approx 0.00245$ vs. Poisson $e^{-6} \approx 0.00248$ (very close).

# Degree distribution in $G(N, p)$

If $p = \lambda/(N-1)$, then, for any $v \in V$,

$$\deg(v) \;\sim\; \mathrm{Bin}(N-1, p) \;\approx\; \mathrm{Pois}(\lambda).$$

- Mean degree: $\mathbb{E}[\deg(v)] = (N-1)p$.
- $\mathbb{P}\left(\deg(v) = k\right) \approx \dfrac{\lambda^k}{k!} e^{-\lambda}$.

**Note**

This gives closed forms for expectations; Poisson is a great approximation when $N$ is large and $p$ small.

# Degree distribution: finite $N$ concentration bounds

# Concentration: Chebyshev (simple but general)

**Theorem (Chebyshev inequality)**

For any r.v. $X$ with mean $\mu$ and variance $\sigma^2$,

$$\mathbb{P}(|X-\mu| \geq t) \leq \frac{\sigma^2}{t^2}.$$

**For degree:** $\deg(v) \sim \mathrm{Bin}(N-1, p)$, so

$$\mathbb{P}\big(|\deg(v) - (N-1)p| \geq t\big) \leq \frac{(N-1)p(1-p)}{t^2}.$$

Chebyshev already gives some concentration guarantees (e.g. take $t_0 = \sqrt{\frac{N}{\delta}p(1-p)}$ for small $\delta > 0$) but sharper results are possible.

# Appendix: Proof of the Chebyshev inequality

**Theorem ( Markov's inequality )**

If $Z \geq 0$ then $\mathbb{P}(Z \geq t) \leq \frac{1}{t}\mathbb{E}[Z]$.

Indeed,

$$\mathbb{E}[Z] \;\leq\; \mathbb{E}[Z\mathbb{1}(Z \geq t)] \;\leq\; t\mathbb{E}[\mathbb{1}(Z \geq t)] \;=\; t\mathbb{P}(Z \geq t).$$

Now, Chebyshev's inequality follows easily from Markov's. Take $Z = |X - \mu|$ then

$$\mathbb{P}(|X - \mu| \geq t) \;=\; \mathbb{P}((X - \mu)^2 \geq t^2) \;\leq\; \frac{\mathbb{E}(X - \mu)^2}{t^2} \;=\; \frac{\sigma^2}{t^2}.$$

# Sharper concentration: Hoeffding for Binomial

**Theorem (Hoeffding inequality)**

If $X = \sum_{i=1}^{n} Z_i$ with independent $Z_i \in [0,1]$ and $\mathbb{E}X = \mu$, then for $t > 0$,

$$\mathbb{P}(|X - \mu| \geq t) \;\leq\; 2\exp\left(-\frac{2t^2}{n}\right).$$

**Applied to degree:** $\deg(v)$ has $N-1$ independent Bernoulli summands,

$$\mathbb{P}(|\deg(v) - (N-1)p| \geq t) \;\leq\; 2\exp\left(-\frac{2t^2}{N-1}\right).$$

Fix $v \in V$. Taking $t_0 = \sqrt{\frac{N-1}{2}\log(\frac{2}{\delta})}$ for small $\delta > 0$ gives

$$\mathbb{P}\big(|\deg(v) - (N-1)p| \geq t_0\big) \;\leq\; \delta.$$

note much better behavior of $t_0$ on $\delta$

# Sharper concentration: Hoeffding for Binomial

**Theorem (Hoeffding inequality)**

If $X = \sum_{i=1}^{n} Z_i$ with independent $Z_i \in [0,1]$ and $\mathbb{E}X = \mu$, then for $t > 0$,

$$\mathbb{P}(|X - \mu| \geq t) \;\leq\; 2\exp\Big(-\tfrac{2t^2}{n}\Big).$$

**Applied to degree:** $\deg(v)$ has $N-1$ independent Bernoulli summands,

$$\mathbb{P}(|\deg(v) - (N-1)p| \geq t) \;\leq\; 2\exp\left(-\frac{2t^2}{N-1}\right).$$

Fix $v \in V$. Taking $t_0 = \sqrt{\frac{N-1}{2}\log(\frac{2}{\delta})}$ for small $\delta > 0$ gives

$$\mathbb{P}\big(|\deg(v) - (N-1)p| \geq t_0\big) \;\leq\; \delta.$$

note much better behavior of $t_0$ on $\delta$

e.g. $N = 1001$, $\delta = 0.05$, $p = 0.1$. Then with prob. $\geq 0.95$

$\deg(v) \;\in\; (100 - 42.95, 100 + 42.95) \;=\; (57.05, 142.95)$.

# Uniform degree bounds

Recall: $\mathbb{P}(|\deg(v) - (N-1)p| \geq t) \leq 2\exp\left(-\frac{2t^2}{N-1}\right)$ for all $t > 0$.

Suppose we now want to provide a bound for the degrees all $v \in V$.

# Uniform degree bounds

Recall: $\mathbb{P}(|\deg(v) - (N-1)p| \geq t) \leq 2\exp\left(-\frac{2t^2}{N-1}\right)$ for all $t > 0$.

Suppose we now want to provide a bound for the degrees all $v \in V$.

Take $t_0 = \sqrt{\frac{N-1}{2}\log(\frac{2N}{\delta})}$ we get that, for any fixed $v \in V$,

$$\mathbb{P}(|\deg(v) - (N-1)p| \geq t_0) \;\leq\; \frac{\delta}{N}.$$

**Union bound**: For any two events $\mathbb{P}(A \cup B) \leq \mathbb{P}(A) + \mathbb{P}(B)$.

$$\mathbb{P}(\exists v \;\; |\deg(v) - (N-1)p| \geq t_0) \;\leq\; \sum_{v \in V} \mathbb{P}(|\deg(v) - (N-1)p| \geq t_0) \;\leq\; \delta.$$

# Uniform degree bounds

Recall: $\mathbb{P}(|\deg(v) - (N-1)p| \geq t) \leq 2\exp\left(-\frac{2t^2}{N-1}\right)$ for all $t > 0$.

Suppose we now want to provide a bound for the degrees all $v \in V$.

Take $t_0 = \sqrt{\frac{N-1}{2}\log(\frac{2N}{\delta})}$ we get that, for any fixed $v \in V$,

$$\mathbb{P}(|\deg(v) - (N-1)p| \geq t_0) \;\leq\; \frac{\delta}{N}.$$

**Union bound**: For any two events $\mathbb{P}(A \cup B) \leq \mathbb{P}(A) + \mathbb{P}(B)$.

$$\mathbb{P}(\exists v \;\; |\deg(v) - (N-1)p| \geq t_0) \;\leq\; \sum_{v \in V} \mathbb{P}(|\deg(v) - (N-1)p| \geq t_0) \;\leq\; \delta.$$

e.g. $N = 1001$, $\delta = 0.05$, $p = 0.1$. Then with prob. $\geq 0.95$ all degrees lie in $(100 - 72.8, 100 + 72.8) \;=\; (27.2, 172.8)$.

# Asymptotics in networks

# Asymptotic Thinking in Random Graphs

**Why asymptotics?**

- We study $G(N, p)$ as $N \to \infty$ to reveal general patterns.
- Precise constants matter less than the scaling behavior of $p$ with $N$.

# Asymptotic Thinking in Random Graphs

**Why asymptotics?**

- We study $G(N, p)$ as $N \to \infty$ to reveal general patterns.
- Precise constants matter less than the scaling behavior of $p$ with $N$.

- $f(N) = o(g(N))$ means $f(N)/g(N) \to 0$.
- $f(N) = O(g(N))$ means $|f(N)| \leq C|g(N)|$; for some $C > 0$ and $N$ large enough.
- $f(N) \sim g(N)$ means $f(N)/g(N) \to 1$.

# Asymptotic Thinking in Random Graphs

**Why asymptotics?**

- We study $G(N,p)$ as $N \to \infty$ to reveal general patterns.
- Precise constants matter less than the scaling behavior of $p$ with $N$.

- $f(N) = o(g(N))$ means $f(N)/g(N) \to 0$.
- $f(N) = O(g(N))$ means $|f(N)| \leq C|g(N)|$; for some $C > 0$ and $N$ large enough.
- $f(N) \sim g(N)$ means $f(N)/g(N) \to 1$.

**Probabilistic language:**

- "With high probability" (w.h.p.) means $\mathbb{P}(\text{event}) \to 1$ as $N \to \infty$.
- Example: in $G(N,p)$ with $p = \frac{\log N}{N}$, the graph is connected *w.h.p.*

# Average degree: dense vs sparse graphs

When $N$ grows, the connection probability $p = p_N$ can scale differently.

**Dense regime:** $(p_N)$ tends to a constant $c > 0$.

- $\mathbb{E}[\deg(v)] \approx cN$ grows linearly with $N$.
- The number of edges $L \approx c\binom{N}{2}$.
- Not a realistic large network, but a useful contrast.

**Sparse regime:** $p_N = \lambda/(N-1)$ (or smaller).

- $\mathbb{E}[\deg(v)] \approx \lambda$ stays constant as $N \to \infty$.
- The total number of edges $L \approx \lambda N/2$ grows linearly with $N$.

# Average degree: dense vs sparse graphs

When $N$ grows, the connection probability $p = p_N$ can scale differently.

**Dense regime:** $(p_N)$ tends to a constant $c > 0$.

- $\mathbb{E}[\deg(v)] \approx cN$ grows linearly with $N$.
- The number of edges $L \approx c\binom{N}{2}$.
- Not a realistic large network, but a useful contrast.

**Sparse regime:** $p_N = \lambda/(N-1)$ (or smaller).

- $\mathbb{E}[\deg(v)] \approx \lambda$ stays constant as $N \to \infty$.
- The total number of edges $L \approx \lambda N/2$ grows linearly with $N$.

**Language note:**

- Saying "real networks are sparse" means that as they grow, the *average degree* stays bounded, not that $p$ is small for a fixed $N$.
- The scaling of $p_N$ determines which asymptotic regime we are in.

# Maximum degree in $G(N, p)$

Let $\Delta = \max_v \deg(v)$ be the **maximum degree**.

**Dense regime:** $(p_N)$ tends to a constant $c > 0$.

- With high probability (remember we ignore constants here):

$$\Delta \;=\; Np \,+\, O\big(\sqrt{N \log N}\big).$$

**Sparse regime:** $p_N = \lambda/(N-1)$ (or smaller).

- Each $\deg(v) \approx \mathrm{Pois}(\lambda)$ — mean $\lambda$.
- By extreme–value theory for Poisson tails:

$$\Delta \;\approx\; \frac{\log N}{\log \log N}.$$

This is very thin tailed: $N = 10^3, 10^6, 10^{12}$ gives $\frac{\log N}{\log \log N} = 4.3, 6.3, 9.2$.
In real networks we observe "hubs".

# Notation: average degree vs expected degree

For a graph $G$ with $N$ vertices and $L$ edges:

- The **empirical average degree** is (a random variable)

$$\overline{\deg}(G) \; = \; \frac{1}{N} \sum_{v \in V} \deg(v) \; = \; \frac{2L}{N}.$$

- The **expected degree** under a random graph model is

$$\mathbb{E}[\deg(v)] \; = \; \mathbb{E}[\overline{\deg}(G)] \qquad \text{for all } v \in V.$$

**Example (Erdős–Rényi $G(N, p)$):**

$$\overline{\deg}(G) \approx (N-1)p, \qquad \mathbb{E}[\deg(v)] = (N-1)p.$$

We saw that for large $N$, $\overline{\deg}(G)$ is concentrated around $\mathbb{E}[\deg]$.

# Threshold phenomena and giant component

# Threshold phenomena in ER (concept)

## Definition

A **threshold** for a graph property $\mathcal{P}$ is a function $p^*(N)$ such that:

$$p \ll p^*(N) \Rightarrow G(N,p) \text{ has } \neg\mathcal{P} \text{ w.h.p.},$$

$$p \gg p^*(N) \Rightarrow G(N,p) \text{ has } \mathcal{P} \text{ w.h.p.}$$

ER graphs display many sharp thresholds:

- Emergence of a giant component.
- Connectivity (no isolated vertices).
- Appearance of fixed subgraphs (e.g., triangles).

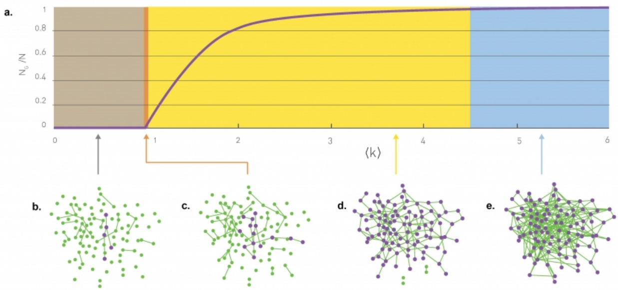
# Regimes of $G(N, p)$ (sparse case $p = c/N$)

It is useful to describe random graphs in terms of the **expected degree**

$$\mathbb{E}[\deg(v)] \approx c.$$

- **Subcritical regime ($c < 1$):** only small tree-like components; largest size $\sim \log N$.
- **Critical point ($c = 1$):** largest component has size $\sim N^{2/3}$; no giant yet.
- **Supercritical regime ($c > 1$):** a unique **giant component** emerges, containing a positive fraction of nodes.
- **Connected regime ($c \gtrsim \log N$):** almost surely the whole graph becomes connected.

# Illustration of regimes




**Interpretation:** As $c$ increases, the largest connected component grows from negligible size, through a sudden phase transition ($c = 1$), and eventually absorbs almost all nodes.

# Why the giant component matters (econ/social)

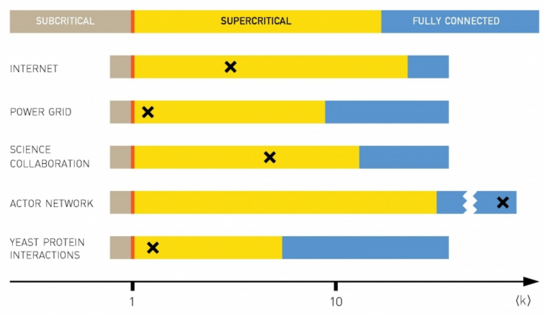
Consider the world's friendship network:

- Clearly disconnected (think small remote communities)
- But "our" component is large, spans most of the world.
- There should be no two big components.

Giant components are important:

- **Contagion & diffusion:** A giant component enables large cascades (diseases, information, bank runs).
- **Market connectivity:** Sufficient density is needed for trade/payment networks to connect most participants.
- **Infrastructure design:** Tuning $p$ (or expected degree $c$) above 1 ensures large-scale reachability.

# Where are real networks?




Most real-world networks live **well above the critical point**.

They are highly connected (often even “superconnected”), yet they also exhibit additional structure (clustering, hubs, communities).

The ER model a *baseline*: it shows that above $c = 1$, large-scale connectivity is the default, but real networks have richer features.

# Connectivity threshold in $G(N, p)$

**Theorem**

The threshold for connectivity in $G(N, p)$ is

$$p^*(N) = \frac{\log N}{N}.$$

More precisely:

$$\begin{cases} p = \frac{\log N + \omega(N)}{N}, & G(N, p) \text{ is connected w.h.p.,} \\ p = \frac{\log N - \omega(N)}{N}, & G(N, p) \text{ is disconnected w.h.p..} \end{cases}$$

Here, $\omega(N)$ means any function that grows to infinity (however slowly).
Examples: $\log \log N$, $\sqrt{\log N}$, or even $\log \log \log N$.

# Idea of proof (intuition)

- A vertex is isolated with probability

$$\Pr(v \text{ isolated}) = (1-p)^{N-1} \approx e^{-pN}.$$

- Expected number of isolated vertices:

$$\mathbb{E}[N_0] = Ne^{-pN}.$$

- If $p = c\frac{\log N}{N}$, then

$$\mathbb{E}[N_0] \approx N^{1-c}.$$

- For $c < 1$, $\mathbb{E}[N_0] \to \infty$; many isolated vertices $\to$ disconnected.

  For $c > 1$, $\mathbb{E}[N_0] \to 0$; isolated vertices disappear.

**Careful:** No isolated vertices do not automatically imply connectivity. However, one can show that once all isolated vertices disappear, all other components merge into one giant component w.h.p.

# Simulation in NetworkX (Colab) — generate and inspect

## Python (run in Google Colab)

```
import networkx as nx
import matplotlib.pyplot as plt

n, p = 200, 0.015  # try also p = 0.005, 0.02, 0.05
G = nx.erdos_renyi_graph(n, p)

print("Nodes:", G.number_of_nodes())
print("Edges:", G.number_of_edges())

# Empirical vs expected average degree
deg = [d for _, d in G.degree()]
print("Empirical mean degree:", sum(deg)/n)
print("Theoretical mean degree:", (N-1)*p)

# Largest component size
components = list(nx.connected_components(G))
largest = max(components, key=len)
print("Largest component size:", len(largest))

# Draw (small n looks better)
plt.figure(figsize=(5,5))
pos = nx.spring_layout(G, seed=7)
nx.draw(G, pos, node_size=30, edge_color="#cccccc")
plt.show()
```

# Simulation in NetworkX — degree histogram

## Python (run in Google Colab)

```
import numpy as np
import matplotlib.pyplot as plt

deg = np.array([d for _, d in G.degree()])
print("Empirical mean degree:", deg.mean())
print("Theoretical mean degree:", (N-1)*p)

plt.figure(figsize=(5,4))
bins = np.arange(deg.max()+2) - 0.5
plt.hist(deg, bins=bins)
plt.xlabel("Degree k"); plt.ylabel("Count")
plt.title("Degree distribution in G(N,p)")
plt.show()
```

**Observation.** For $p = c/N$ the histogram should resemble a Poisson($c$), with empirical mean degree $\overline{\deg}(G)$ close to theoretical $\mathbb{E}[\deg]$.

# Summary

- ER $G(N, p)$ is the baseline random network: tractable degrees and component structure.
- Degrees: Binomial $\to$ Poisson in sparse regime; strong concentration via Hoeffding.
- Phase transitions: giant component at $p \sim 1/N$; connectivity at $p \sim (\log N)/N$.
- Why we care: gives parameter ranges where large-scale behavior becomes plausible.