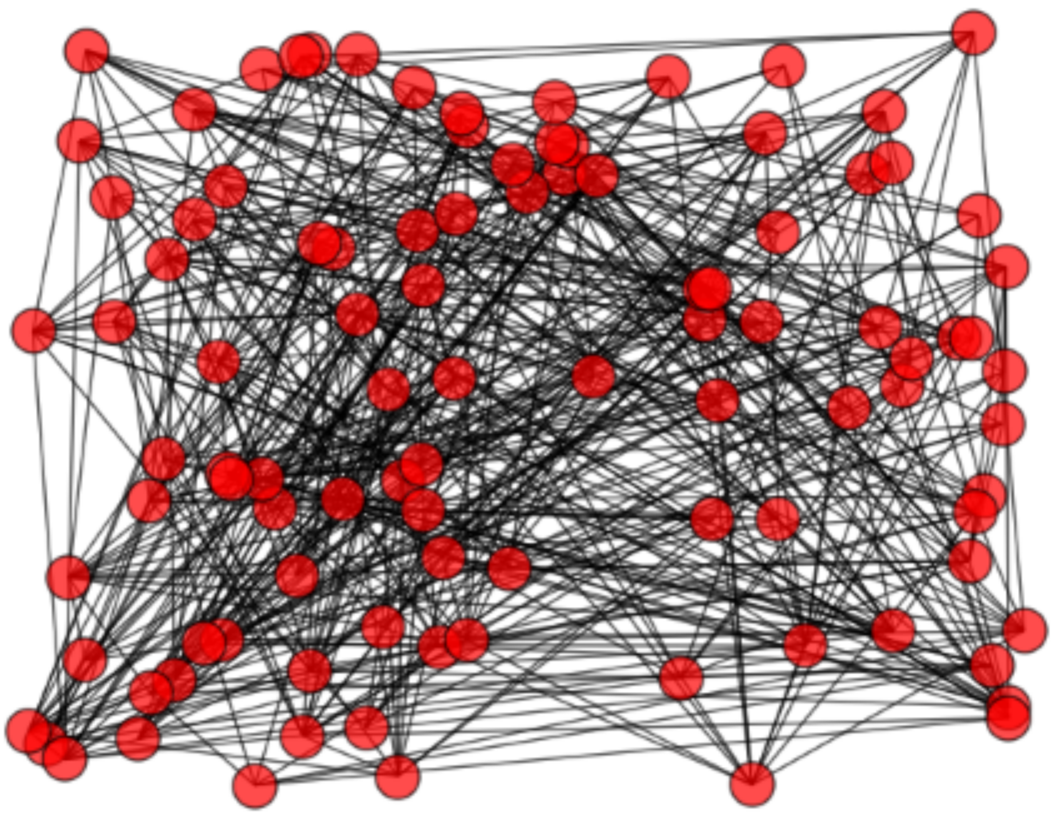
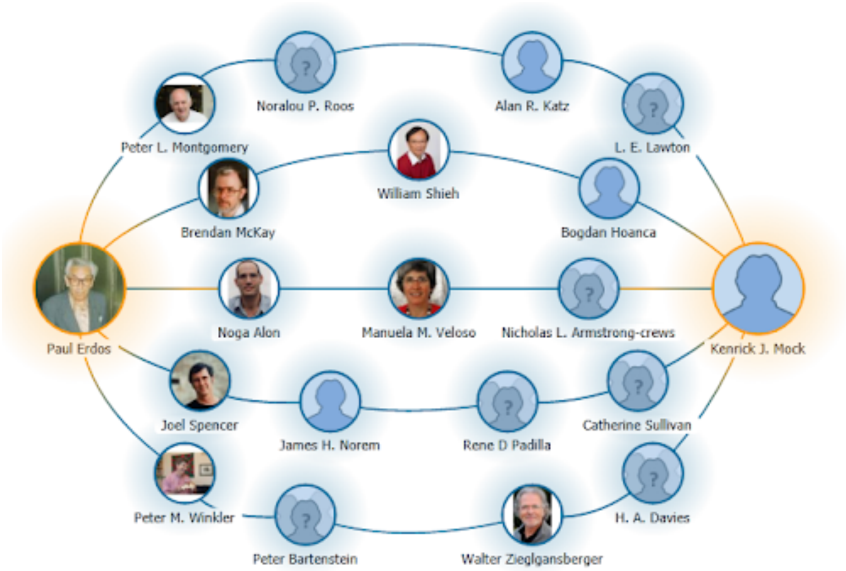
Random Networks

The Erdós-Rényi Model



The Pioneers

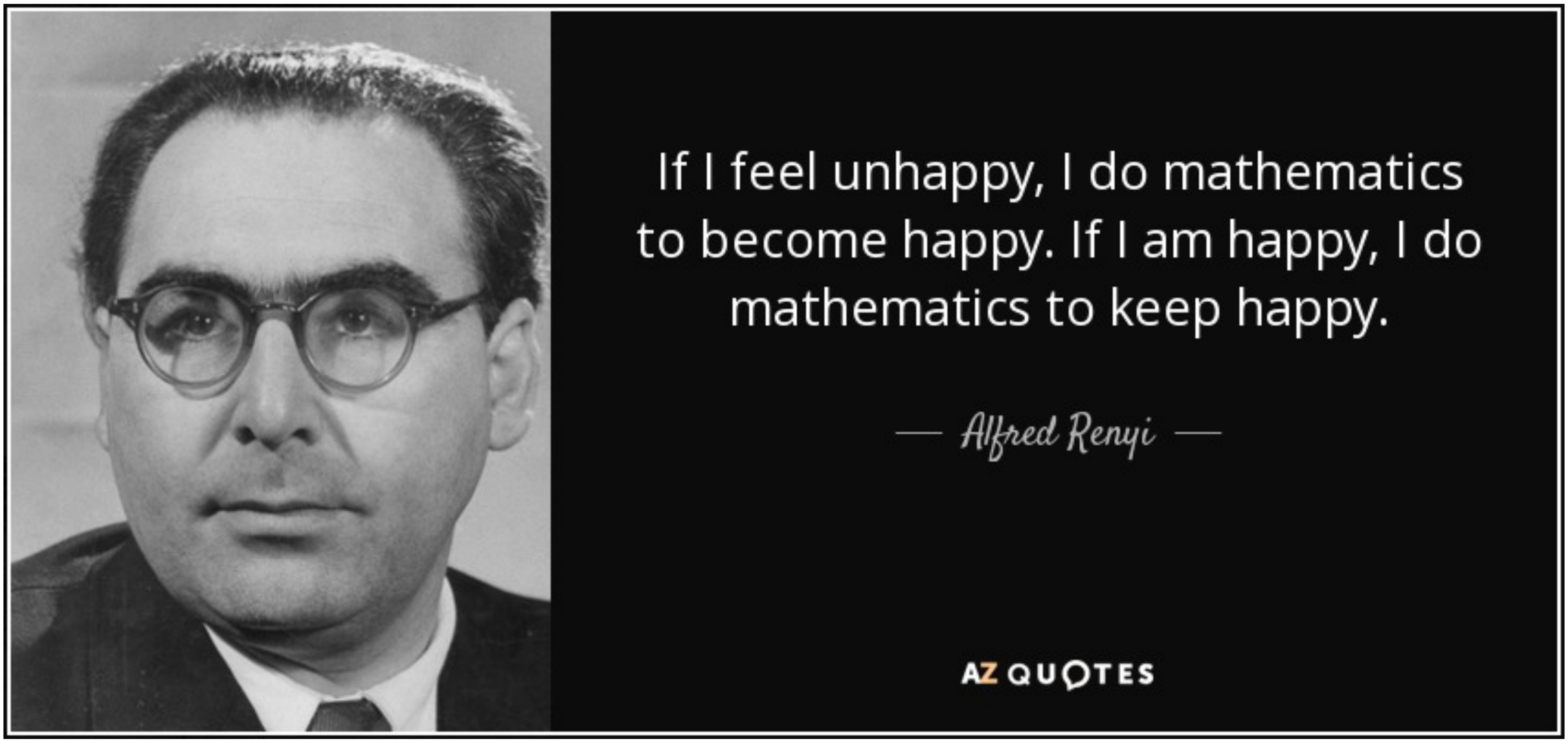
Paul Erdós (1913-1996) was a Hungarian mathematician who published around 1500 mathematical papers over his long career.



The Erdós number refers to the length of the geodesic path connecting an author to Erdós, where an edge means a collaborative paper.

The Pioneers

Alfréd Rényi (1921-1970) was a Hungarian mathematician who published in the area of graph theory and probability, including a number of papers on random graphs with Erdós.

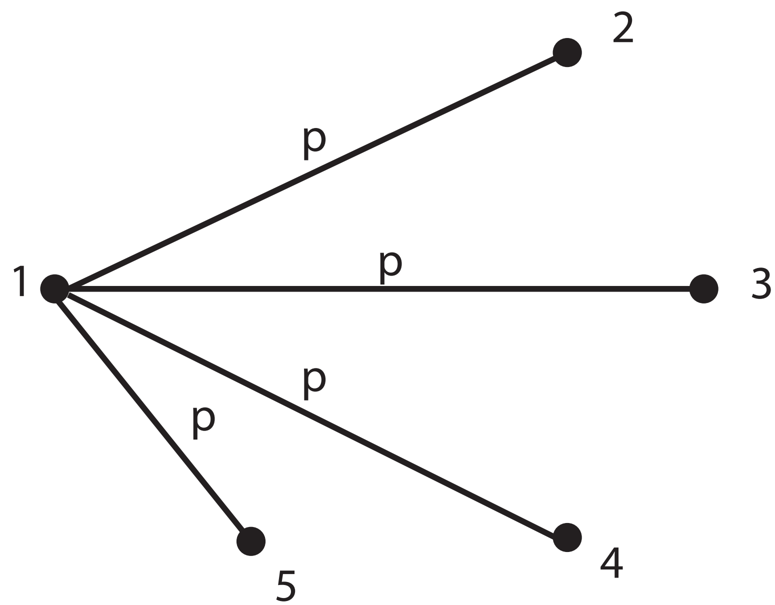


Reason for Studying Random Networks

There are many ways to generate random graphs that have different degree distribution, clustering, community number, etc. The reason for examining these is that they allow us to examine the effects of such properties rather than the specific connectivity pattern among nodes.

An Erdós-Rényi (E-R) Random Graph

For each of the *n* nodes, connect with each other node with probability *p*. This is denoted as G(*n*,*p*). This typically does not allow for self-edges.



This is an example of the first step in creating a G(5,*p*) graph, where edges connecting to node 1 are determined. This would proceed onto node 2, noting that the connection (or not) to node 1 has already been determined.

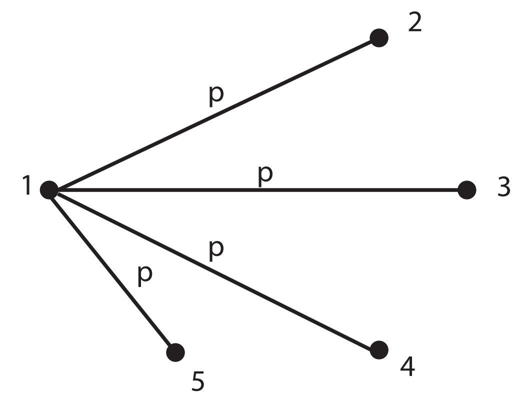
One could create an ensemble of such graphs, each time using a different set of random numbers to pick edges.

Mean Number of Edges

How many node pairs in G(*n*,*p*)? $\binom{n}{2}$

What is the probability that a pair will be connected? p

What is the mean number of edges? $\binom{n}{2}p$



$$\boxed{\langle m \rangle = \binom{n}{2}p}$$

Mean edge number of G(*n*,*p*)

Mean Degree

With this algorithm, what will be the degree of a typical node in an E-R graph? That is, what is the mean degree?

What is the mean degree of a graph with *n* nodes and *m* edges? c= $\frac{2m}{n}$

What is the average of this over the ensemble of networks?

$$\langle d \rangle = \left\langle \frac{2m}{n} \right\rangle$$

$$= \frac{2}{n}\langle m \rangle$$ since *n* is constant across the ensemble

$$= \frac{2}{n}\binom{n}{2}p$$

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

$$=(n-1)p$$

$$\boxed{c = (n-1)p}$$

Mean degree of G(*n*,*p*)

Mean Degree

$$\boxed{c = (n-1)p}$$

Mean degree of G(n,p)

This is very intuitive: each node can connect to $n-1$ other nodes, and the probability of doing so is p. So the formula for mean degree makes good sense.

Scaling of Means with *n*

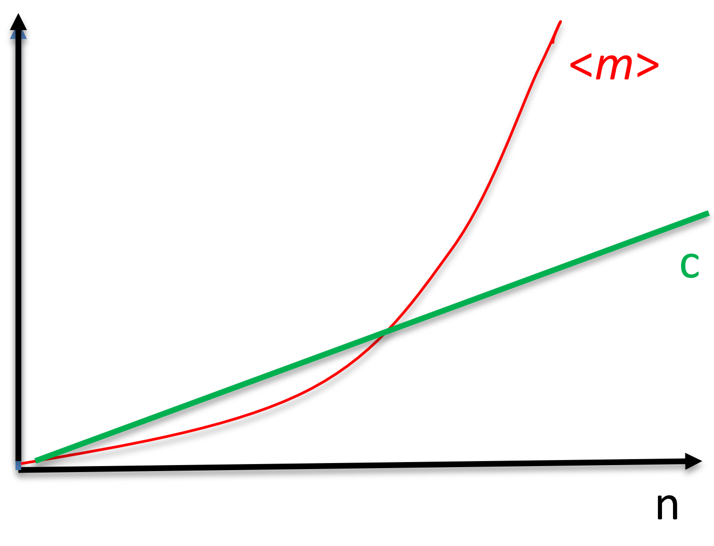
Mean number of edges: $\langle m \rangle = \binom{n}{2} p$

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

$$\approx \frac{1}{2} n^2 p \qquad \text{for large } n$$

Mean degree: $c = (n-1)p$

$$\approx np \qquad \text{for large } n$$



The Full Degree Distribution

We have a formula for the mean degree, but can we come up with the full degree distribution for a graph G(n,p)?

For any node, the probability of connecting with exactly k other nodes is

$$p^k(1-p)^{n-1-k}$$

How many ways are there to pick the k nodes to connect to? $\binom{n-1}{k}$

So probability of having degree k is

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

Degree distribution of G(n,p)

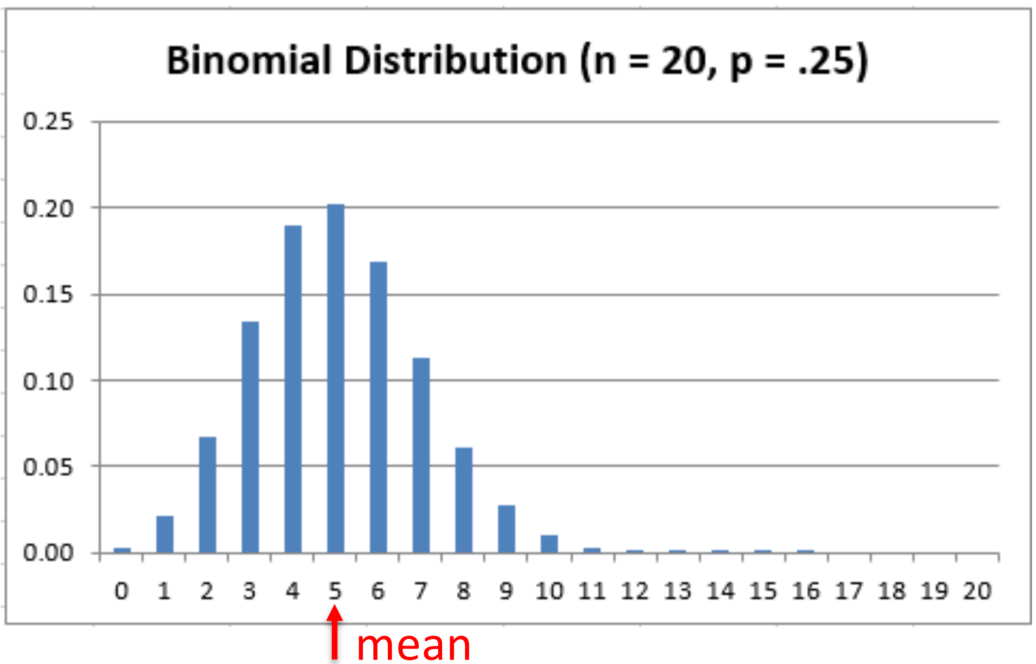
Binomial Degree Distribution

This is just the **binomial distribution**, with $N = n - 1$

$$P_k = \binom{N}{k} p^k (1-p)^{N-k}$$

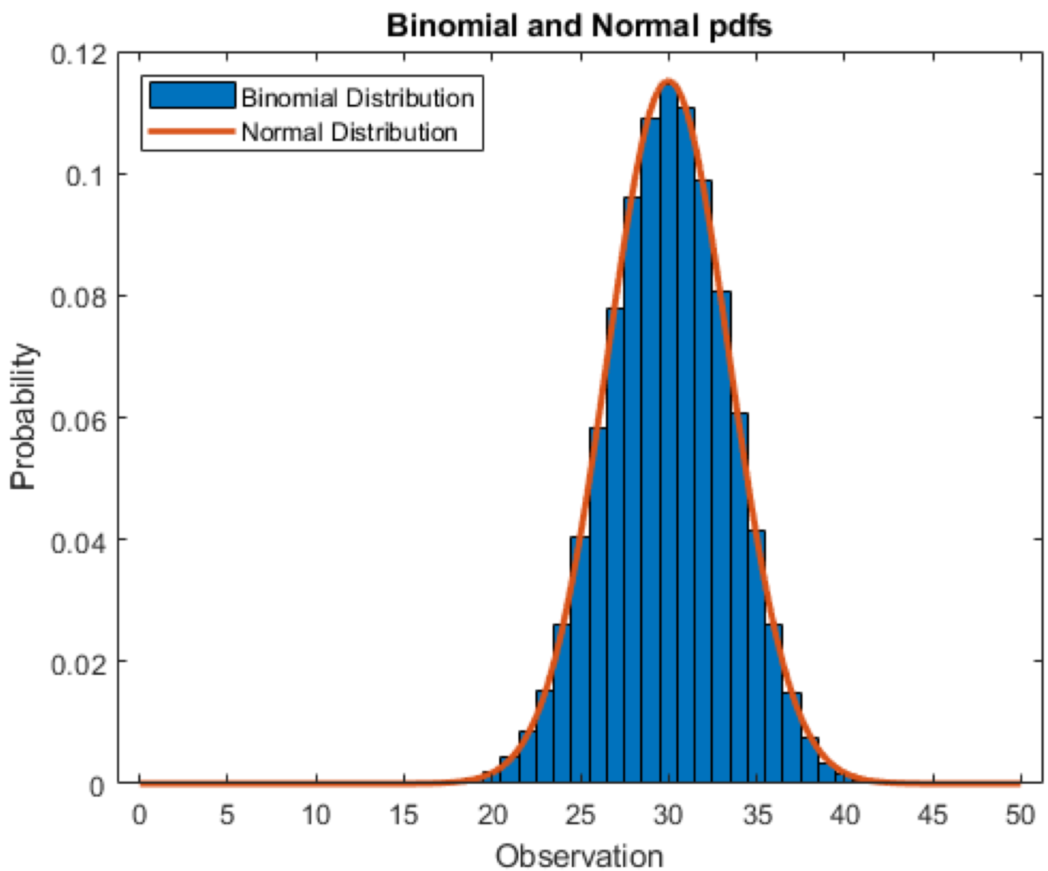
Degree distribution of G(*n*,*p*)

and the mean of such a distribution is *Np*, as we determined for G(*n*,*p*).



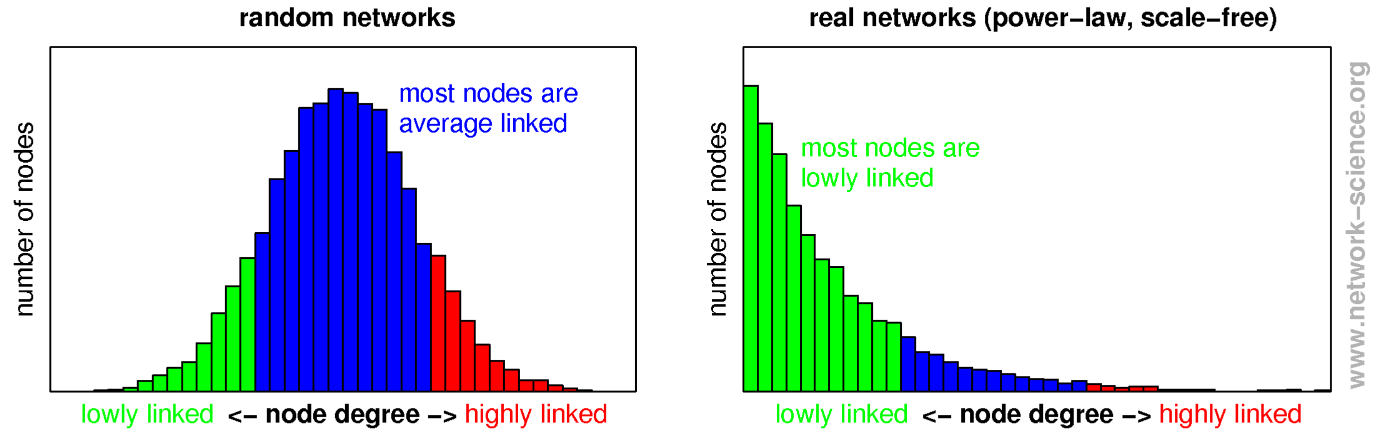
Comparison with Real-World Networks

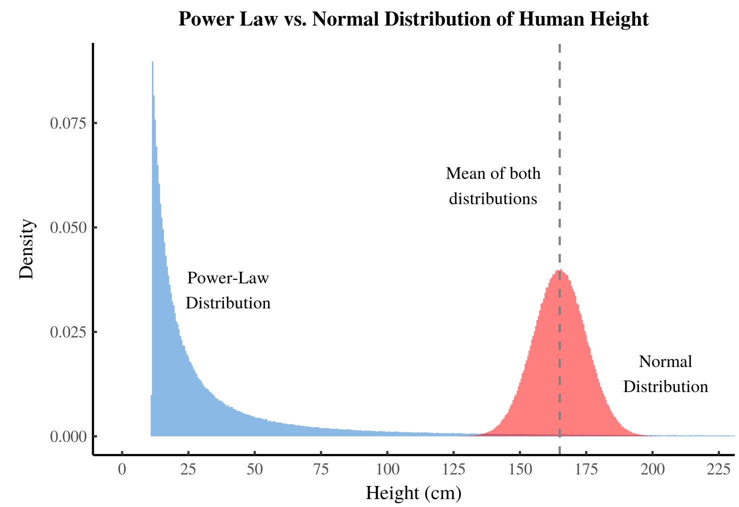
In the limit of large *n*, a binomial distribution converges to a normal distribution



How does this degree distribution compare to that of real-world networks?

Comparison with Real-World Networks





In an E-R network, most nodes have degree near the mean. In scale-free networks like many real-world networks, most nodes have degree far below the mean of the distribution. There are many nodes with low degree and a few with very high degree.

Clustering Coefficient of an E-R Network

Recall that the clustering coefficient is the probability that two neighbors of a node are neighbors of each other. What is the mean clustering coefficient for an E-R network?

The mean degree is $c = (n-1)p$, so $p = \frac{c}{n-1}$. Since *p* is the probability that any two nodes are neighbors in an E-R network, this is the clustering coefficient.

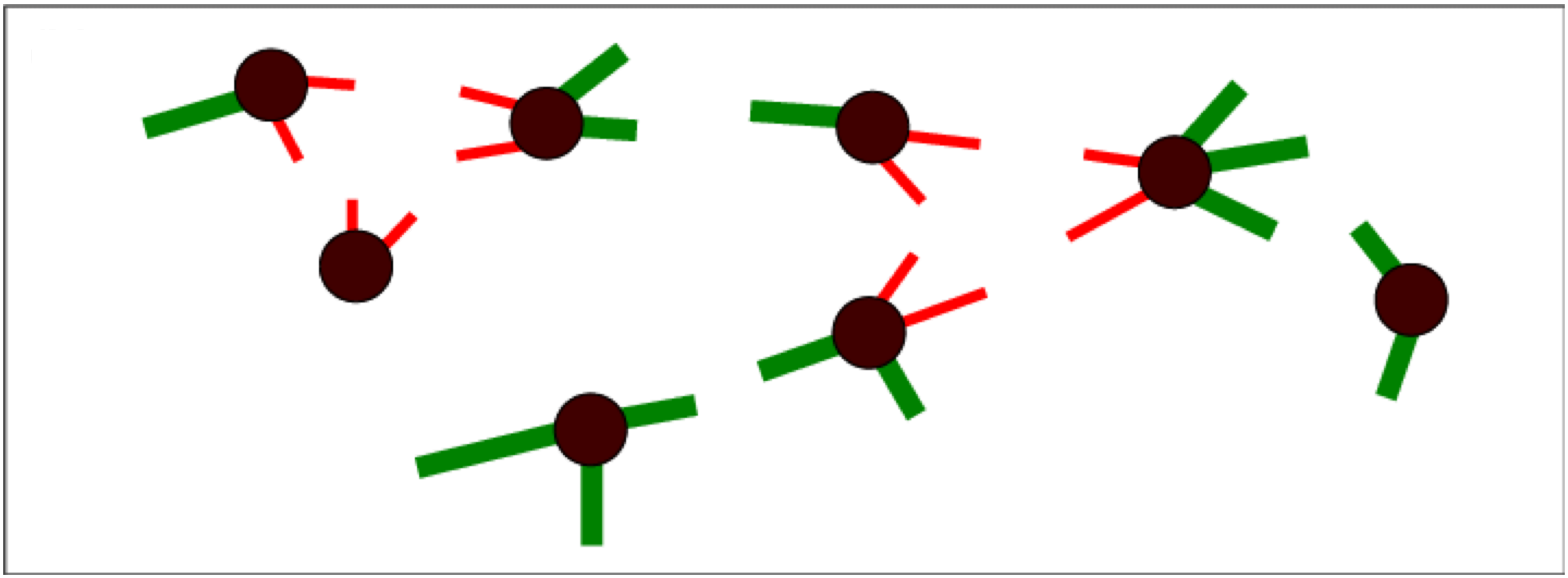
$$\boxed{C = p = \frac{c}{n-1}}$$

Clustering coefficient of an E-R network

Note that the clustering coefficient, *p*, is the same regardless of the number of nodes in the network.

Most real-world networks have relatively large clustering coefficients. For example: for co-authorship of scholarly papers, $C = 0.6$
for the film actor network, $C = 0.2$
With these large values, it is extremely unlikely that these are E-R networks.

The Configuration Model



Another Type of Random Network

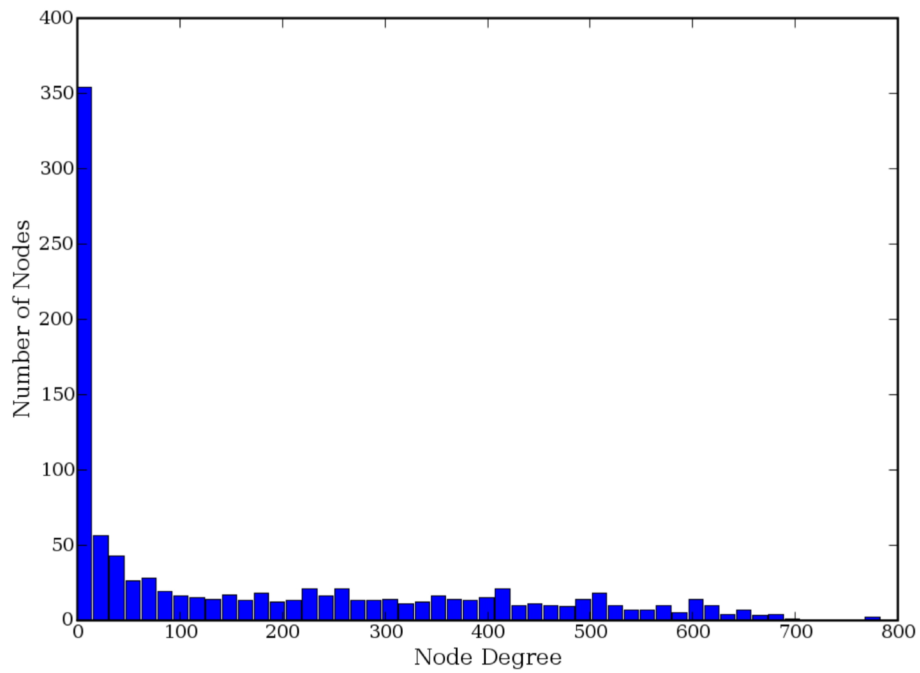
The E-R model is just one type of random network, where the node degrees have a binomial distribution. That is, each node is pretty similar to any other node. There are no hubs.

Real-life networks, in contrast, typically have a power law degree distribution; most nodes have low degree, but there are a few hubs with very high degree.

The configuration model is an algorithm for constructing networks with any degree distribution, including power law distributions. It was first used in 1980.

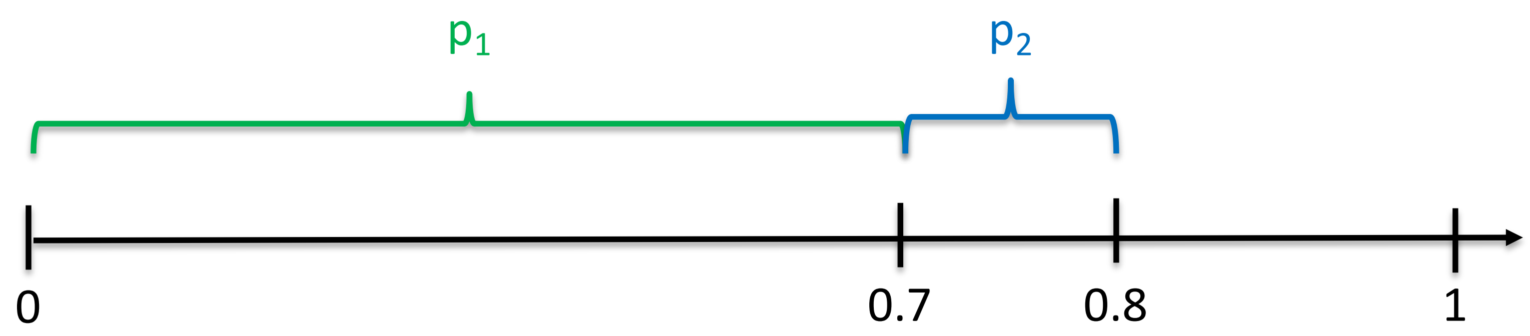
Step 1: Select a Degree Sequence

Number the nodes from 1 to *n*. For each node, determine its degree by sampling from the desired degree distribution.



In this example, with 500 nodes, the probability that a node has degree 1 is $p_1 = \frac{350}{500} = 0.7$. The probability it has degree 2 is $p_2 = \frac{50}{500} = 0.1$. Because these are probabilities, $p_1 + p_2 + \cdots + p_{n-1} = 1$.

Step 1: Select a Degree Sequence



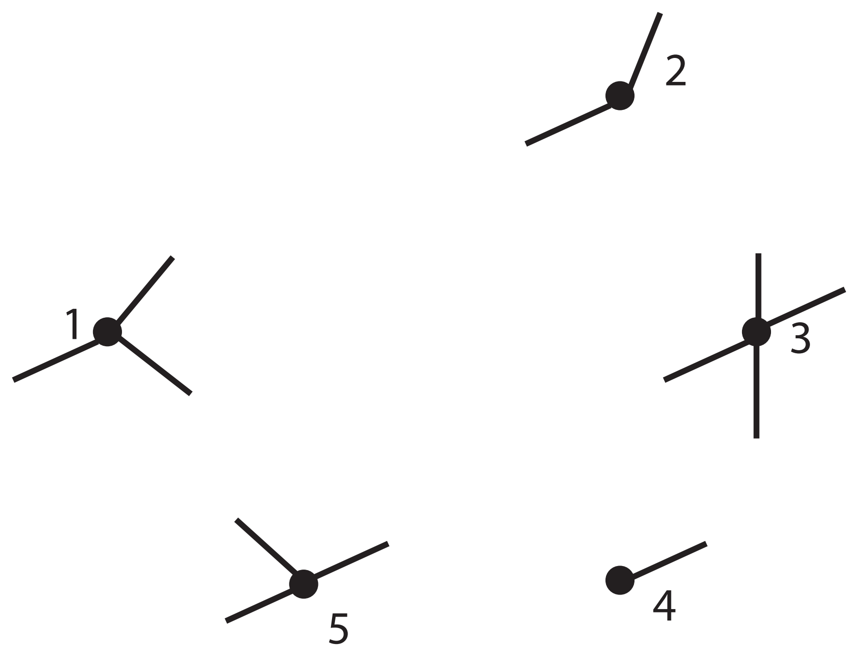
For each node, pick a random number $r \in [0,1]$. If $r < 0.7$ then the degree of the node is 1. If $r \in [0.7,0.8]$ then the degree of the node is 2. Similarly for other values of *r*.

Sampling in this way will give a network with degree distribution similar to what was desired.

The sum of the degrees must be even. If not, increase the degree of a randomly-chosen node by 1.

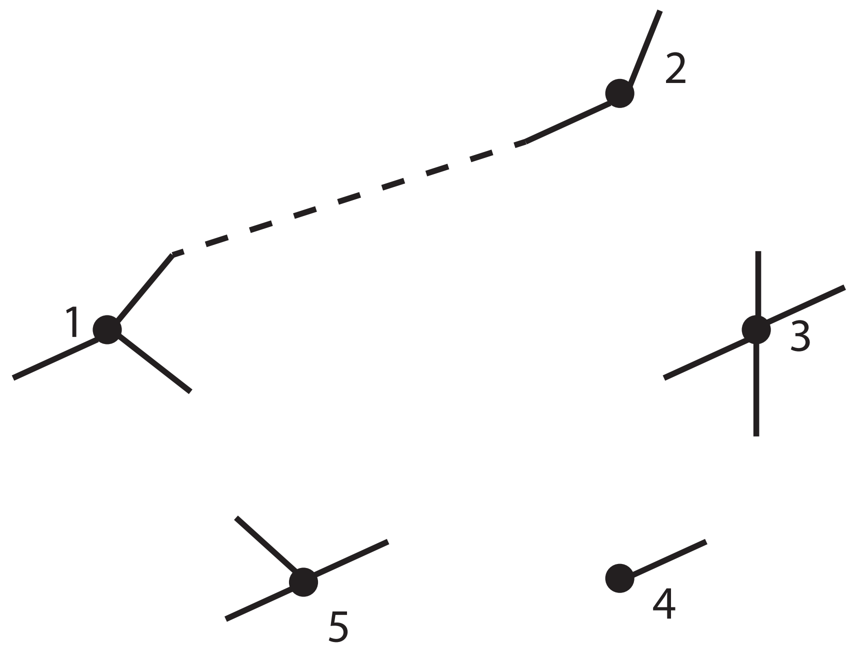
Step 2: Attach "Stubs" to the Nodes

If a node has degree *d*, then it gets *d* stubs.



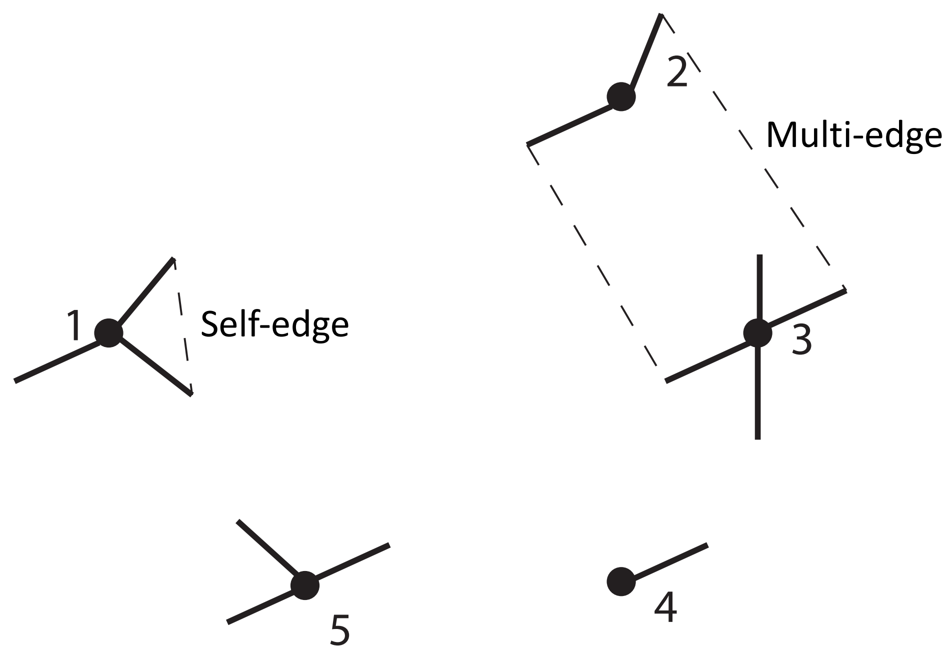
Step 3: Match the “Stubs” to Form Edges

At each stub, randomly match it with another stub. Connect these with an edge.

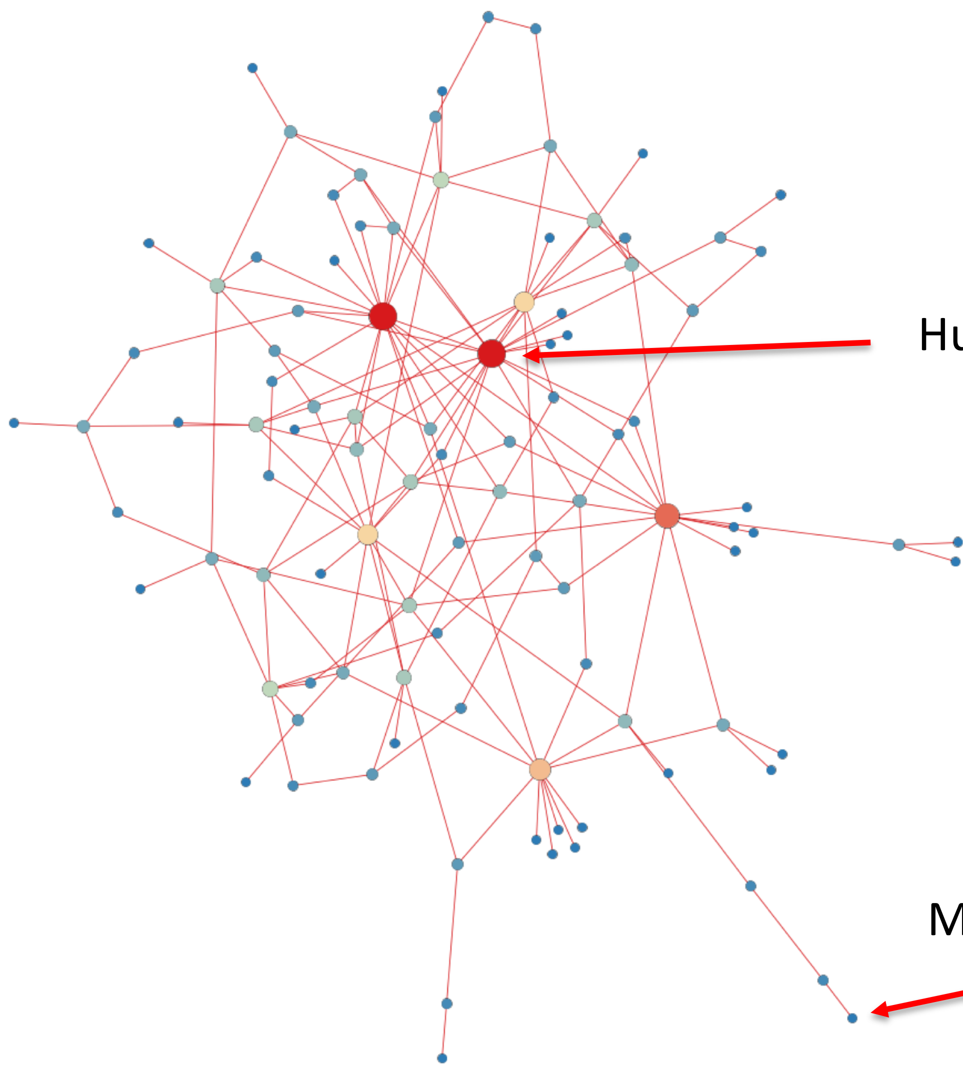


Step 3: Match the "Stubs" to Form Edges

Some self-edges and multi-edges will likely be formed, and this is probably not desired. However, for large networks these are negligible, so don't worry about them.



A Scale-Free Network with $\alpha = 2$ and $n = 100$

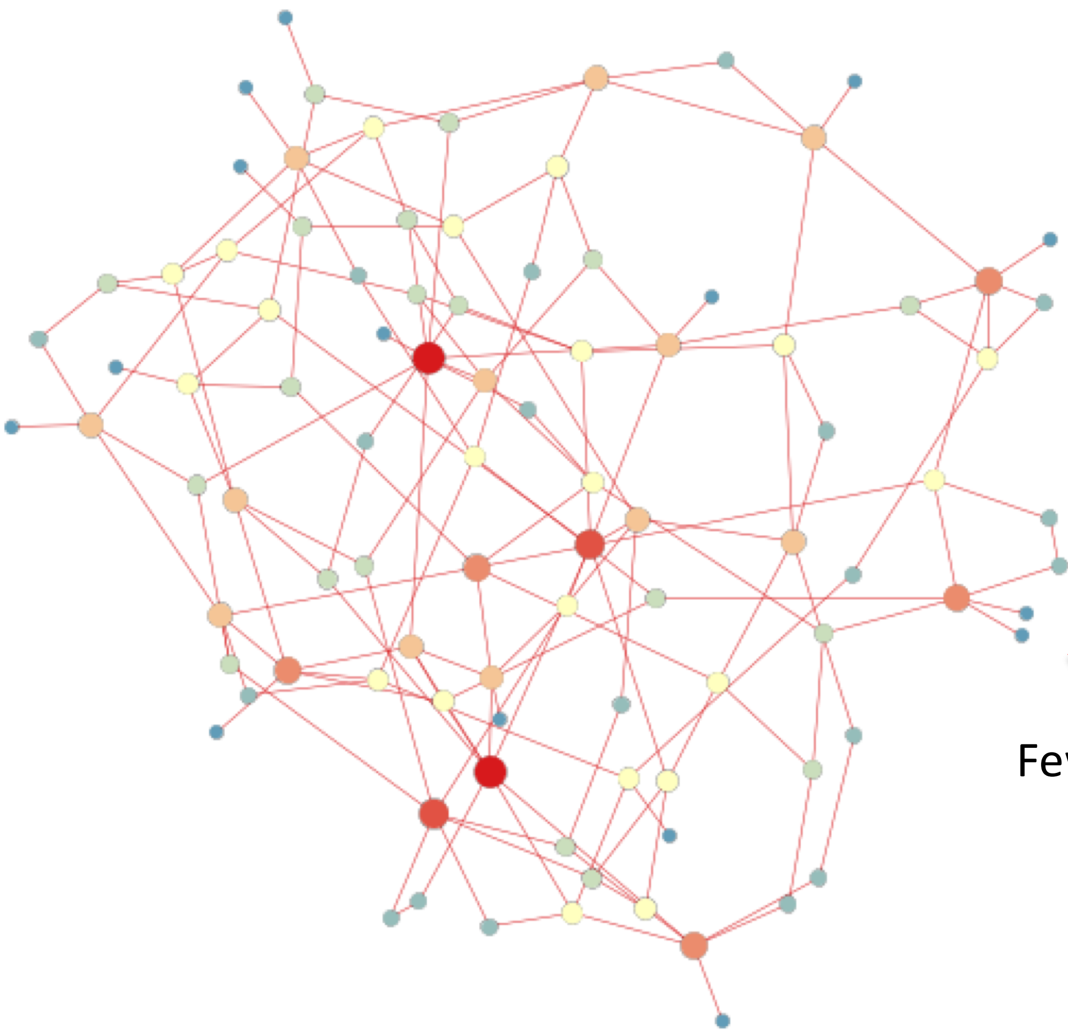


Hubs have high degree

Many low-degree nodes

164 edges

An E-R Network with $p = 0.035$ and $n = 100$



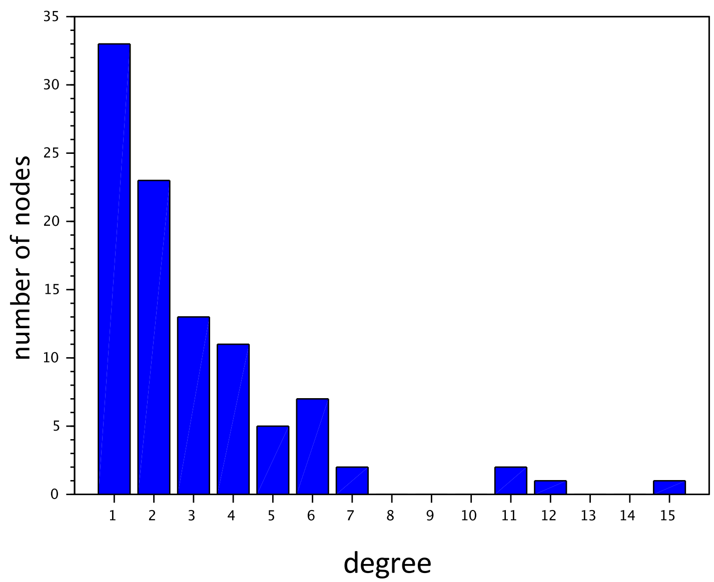
Degree not very high for any node

Fewer low-degree nodes

157 edges

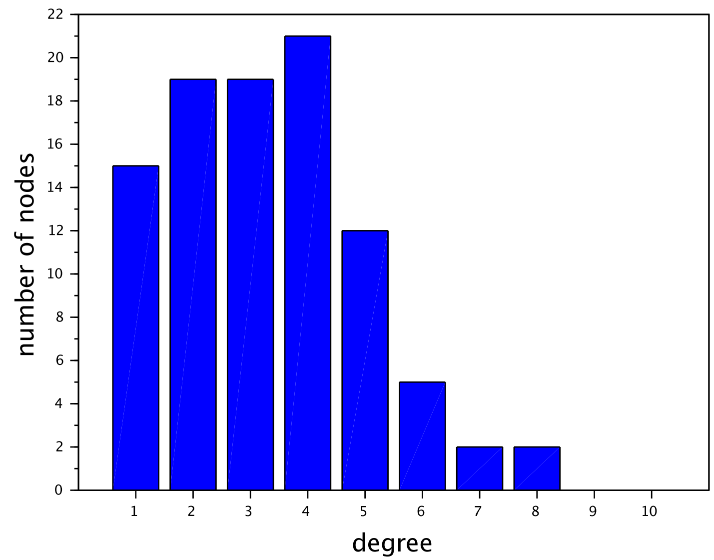
Comparison of Degree Distribution

Scale-free network



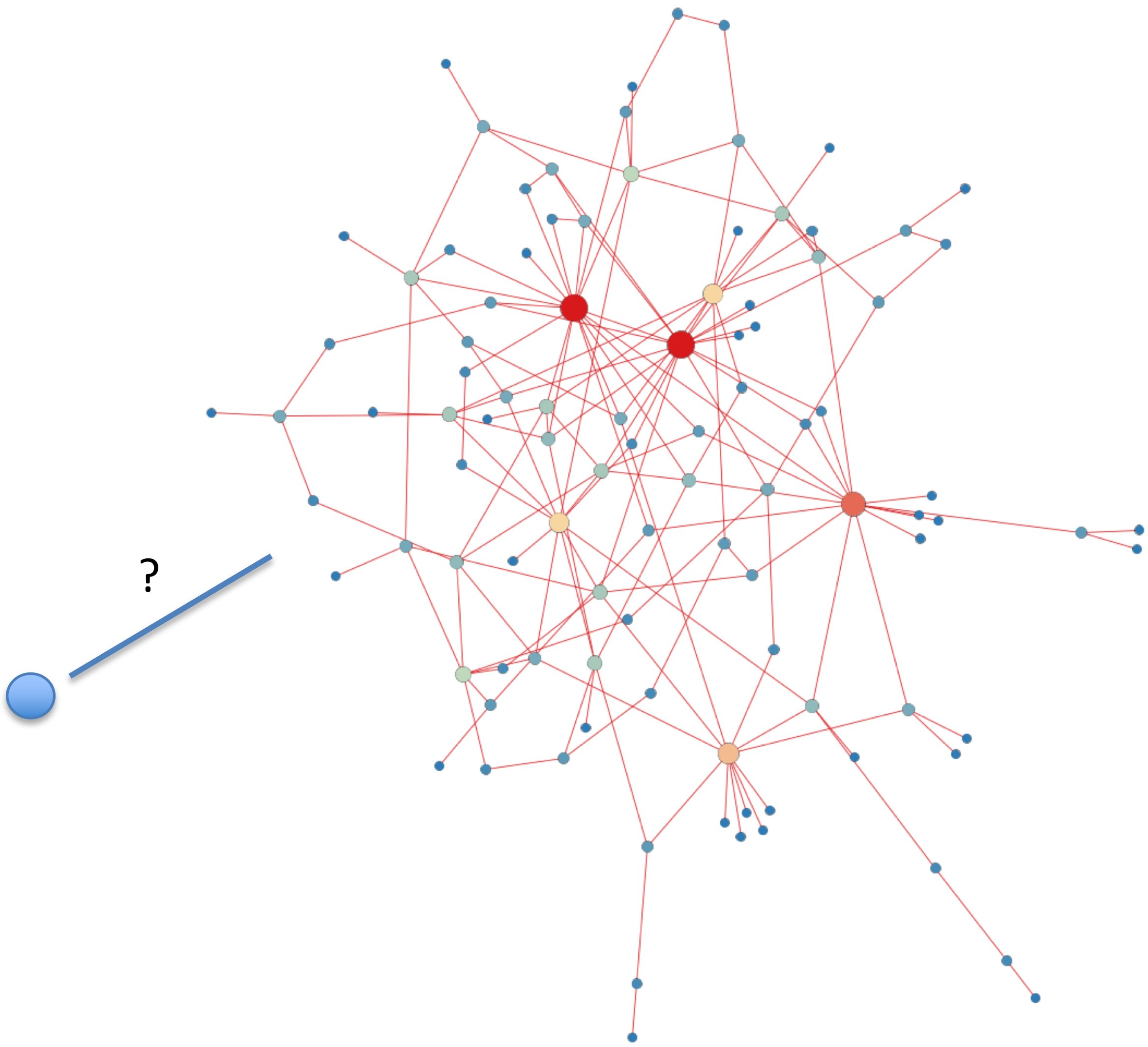
Mean degree=3.08

E-R network



Mean degree=3.14

Dynamic Networks



Where do Scale Free Networks Come From?

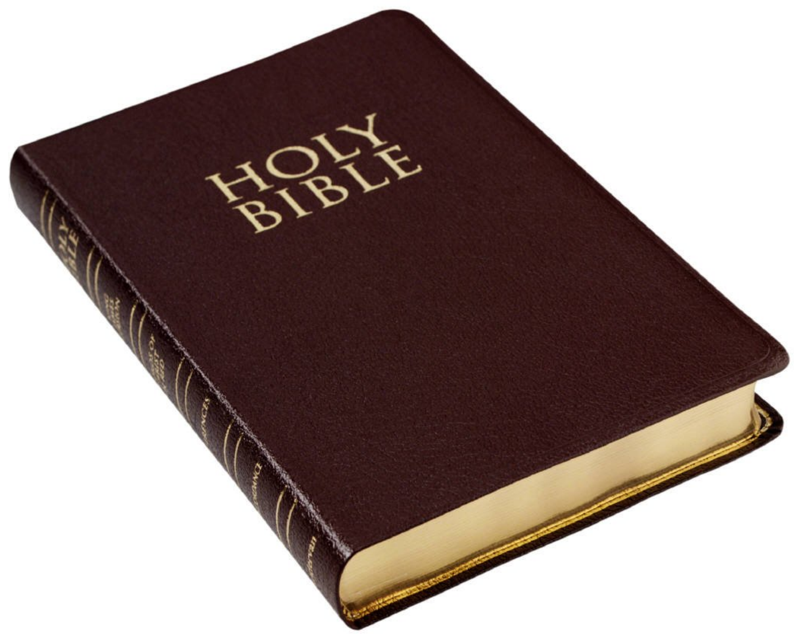
They are present in virtually every type of network. Why?

The “rich get richer” phenomenon really does seem to apply in many situations, including networks.

We have focused so far on static networks, where the number of nodes is fixed. But how does the network develop? When a new node is added, how does it get attached to other nodes? Such networks, where nodes or edges change over time, are called dynamic networks.

Preferential Attachment

Preferential attachment was first used in the Bible in the Gospel of Matthew: “For every one who has will more be given, and he will have abundance; but from him who has not, even what he has will be taken away”. For this reason, preferential attachment is sometimes called the “Matthew Effect”.



When an edge is connected from a new node to some other node in the existing network, target those nodes that already have a lot of edges.

Barabási-Albert Model

The Barabási-Albert (BA) model, published in 1999, is the best-known dynamic network model that incorporates preferential attachment.

Start with a complete graph with n_o nodes. At each iteration do the following two steps:

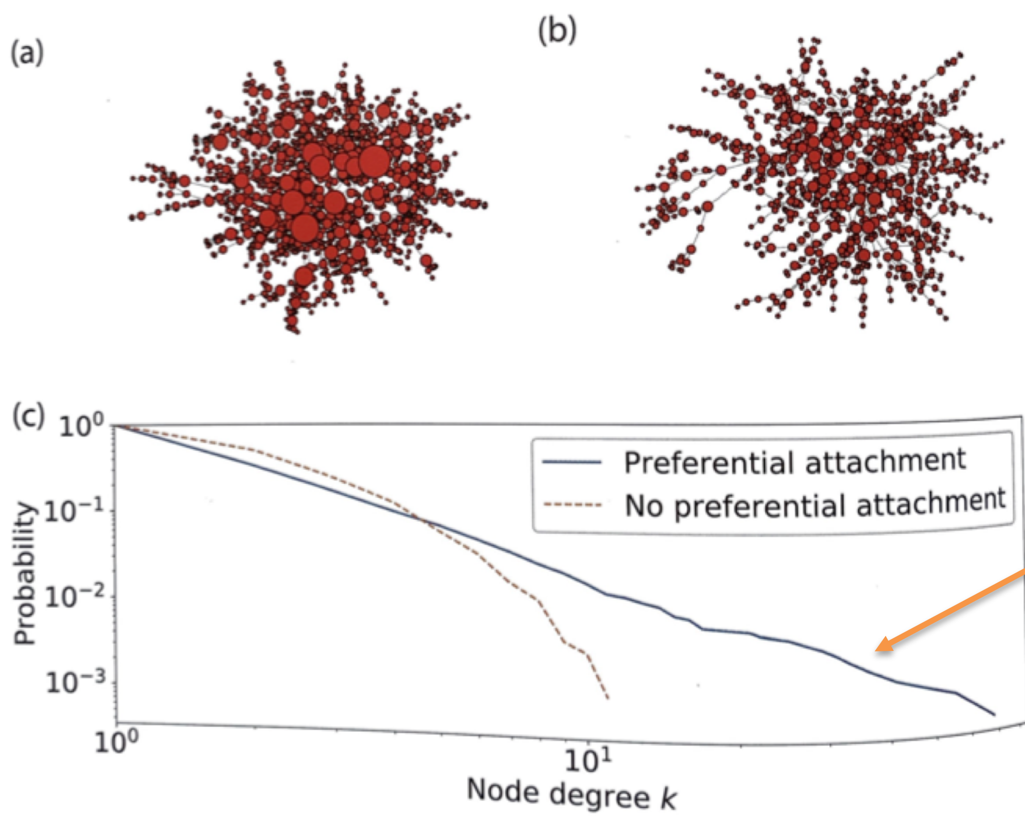
1. A new node *i* is added to the network, with $m \leq n_o$ links attached to it. This *m* will become the average degree of the full network.
2. Each new link is wired to an old node *j* with probability

$$P_{ij} = \frac{k_j}{\sum_l k_l}$$

where k_j is the degree of node *j* and the denominator is the sum of the degrees of all nodes except *i*.

Continue until the desired number of nodes *N* and their edges have been added.

Barabási-Albert Model



N=2000 nodes
$< k > = 2$

Linear in log-log plot means scale free

(a) Network build using the BA model (with preferential attachment)
(b) Network built with uniform probability of attachment

The Rank Model

As a model for the growth of a network, the BA model suffers from one seemingly unrealistic assumption: It assumes that the new node knows the degrees of all the other nodes in the network.

When a new web site is established, it likely links to several other popular web sites, simply because the designer of the site knows about them. Less popular sites just are not known, so no linkage will be made. This is preferential attachment, but the designer did not know the degree of the nodes, just an implicit ranking of the nodes. Links are made to those at the top of the ranking.

The Rank Model

Start with a small graph with n_o nodes. Some property of the nodes, such as degree or fitness, is selected to rank the nodes. At each iteration, do the following:

1. All nodes are ranked. Nodes are assigned ranks $R = 1, 2, \ldots$ Node l receives a rank of R_l.
2. A new node *i* is added to the network, with $m \leq n_o$ new links attached.
3. Each new link from *i* is wired to an old node *j* with probability

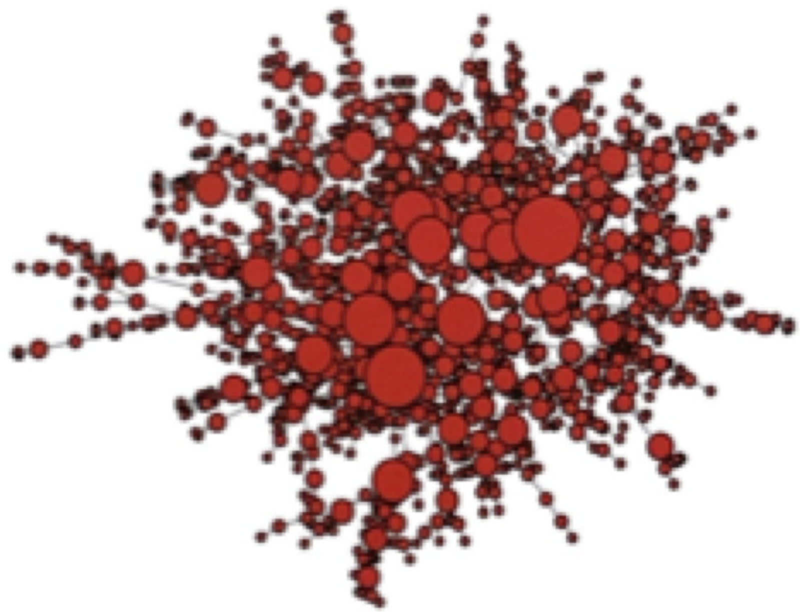
$$p_{ij} = \frac{R_j^{-\alpha}}{\sum_l R_l^{-\alpha}}$$

The nodes may have to be re-ranked after an iteration if the addition of the new node changed the ranking.

The parameter $\alpha > 0$ adjusts the extend of preferential attachment. With larger values, only the most highly ranked nodes will get linked to.

The Random Walk Model

Besides a scale-free degree distribution, real networks often have high local clustering. Does a BA or rank model provide this feature? No

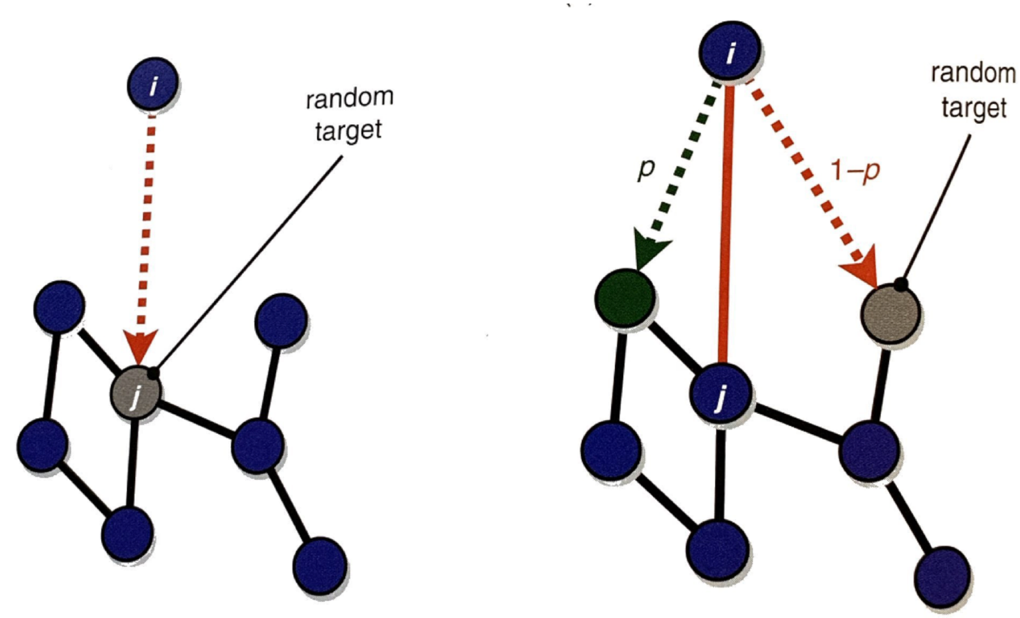


The random walk model enforces a desired level of local clustering.

The Random Walk Model

Start from any small network. For each iteration do the following steps:

1. A new node *i* is added, with $m > 1$ new links attached to it.
2. The first link is wired to an old node *j*, chosen at random.
3. Each other link is attached to a randomly selected neighbor of *j*, with probability *p*, or to another randomly selected node, with probability $1 - p$.



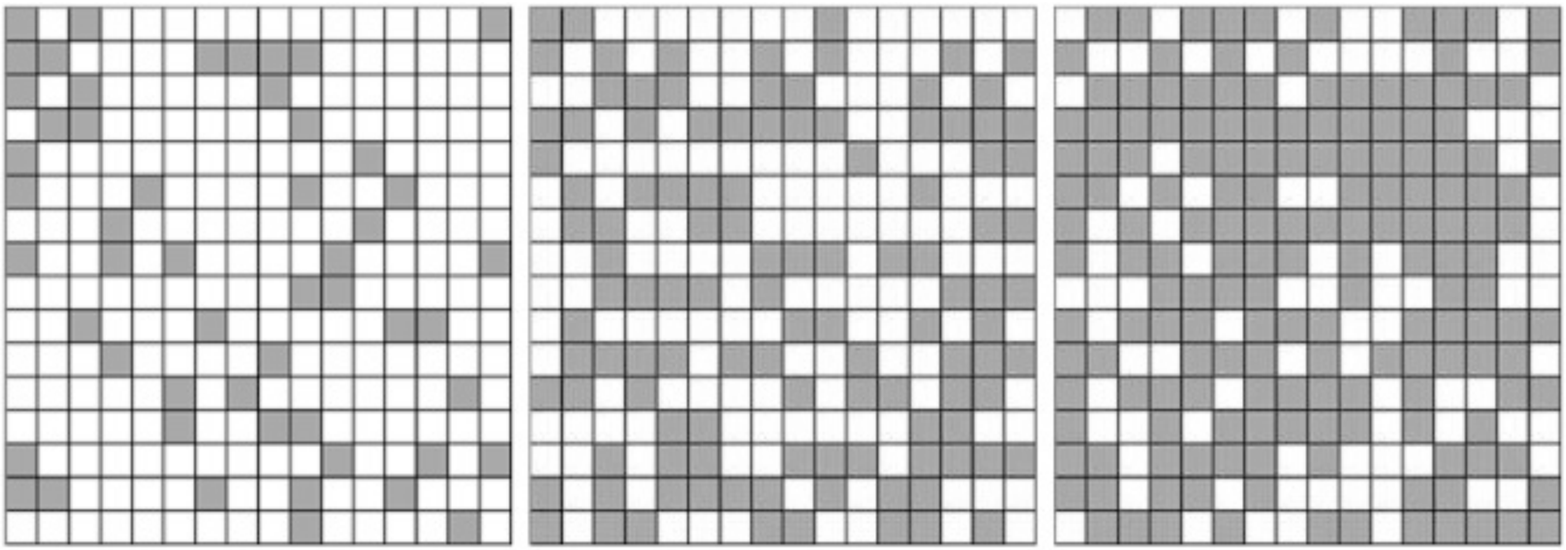
The Random Walk Model

The degree of triadic closure is specified through parameter p.
$p = 0$: There is no enforced closure so nodes are linked at random.
$p = 1$: All links except for the first one are wired to neighbors of the initially selected node, thus maximally closing triangles.

What about the degree distribution? Does this model produce preferential attachment? Yes

Nodes with high degree are neighbors to many other nodes, so they will get chosen more frequently as "neighbors of *j*" than nodes with low degree and not many neighbors.

Network Resilience



(a) $p \approx 0.21$ (b) $p \approx 0.44$ (c) $p \approx 0.66$

How Resilient is a Network to Loss of Nodes?

Most real-life networks have a single giant component, and then several minor components. If one starts removing nodes either randomly or according to some algorithm, how many nodes can be removed before the giant component disintegrates?

The sequential removal of nodes from a network (and the associated edges) is called node percolation, where the fraction of nodes *not removed* (ϕ) is called the **occupation probability**. If $\phi = 1$, then all nodes remain, if $\phi = 0$, then all nodes are removed.

When the network contains a giant component it is said that it percolates. A percolation transition occurs when this giant component disintegrates. The value of ϕ at which the percolation transition occurs is called the percolation threshold.

Publication date: 2000

Citation number: 11,219 by 2025

Error and attack tolerance of complex networks

Réka Albert, Hawoong Jeong & Albert-László Barabási

Department of Physics, 225 Nieuwland Science Hall, University of Notre Dame, Notre Dame, Indiana 46556, USA

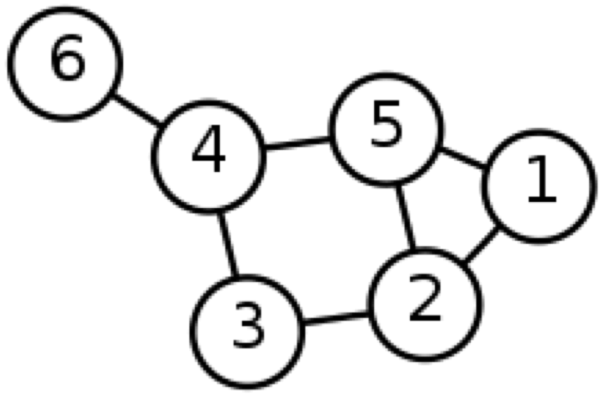
Many complex systems display a surprising degree of tolerance against errors. For example, relatively simple organisms grow, persist and reproduce despite drastic pharmaceutical or environmental interventions, an error tolerance attributed to the robustness of the underlying metabolic network[1]. Complex communication networks[2] display a surprising degree of robustness: although key components regularly malfunction, local failures rarely lead to the loss of the global information-carrying ability of the network. The stability of these and other complex systems is often attributed to the redundant wiring of the functional web defined by the systems' components. Here we demonstrate that error tolerance is not shared by all redundant systems: it is displayed only by a class of inhomogeneously wired networks,

Approach and Central Question

Approach: Use the configuration model to create random scale-free networks and the E-R model to create random networks with binomial degree distribution (they call these “exponential networks” because the fraction of nodes with degree *d* declines approximately exponentially when $d \gg \langle d \rangle$).

Question: Which type of network is most resilient to node removal? More precisely, which type has a higher percolation threshold?

Their Definition of Network Diameter



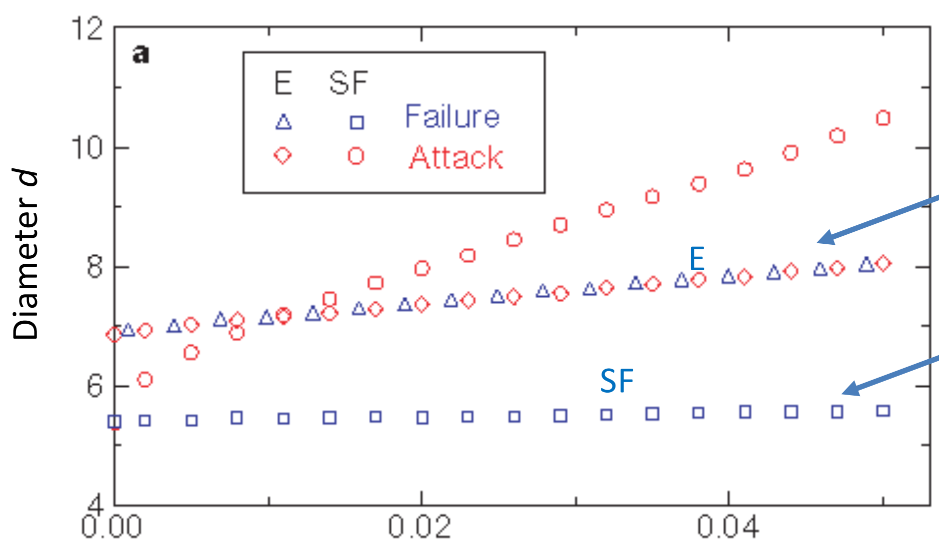
Shortest path from node 1 to node 6 is (1,5,4,6), path length=3

Do this calculation for all node pairs and sum

Divide by the number of node pairs to get diameter *d*
(this is the mean geodesic path length of the network)

Only computed for nodes within the giant component

The Scale-Free Network is **More Resilient** to Random Failure Than the Exponential Network



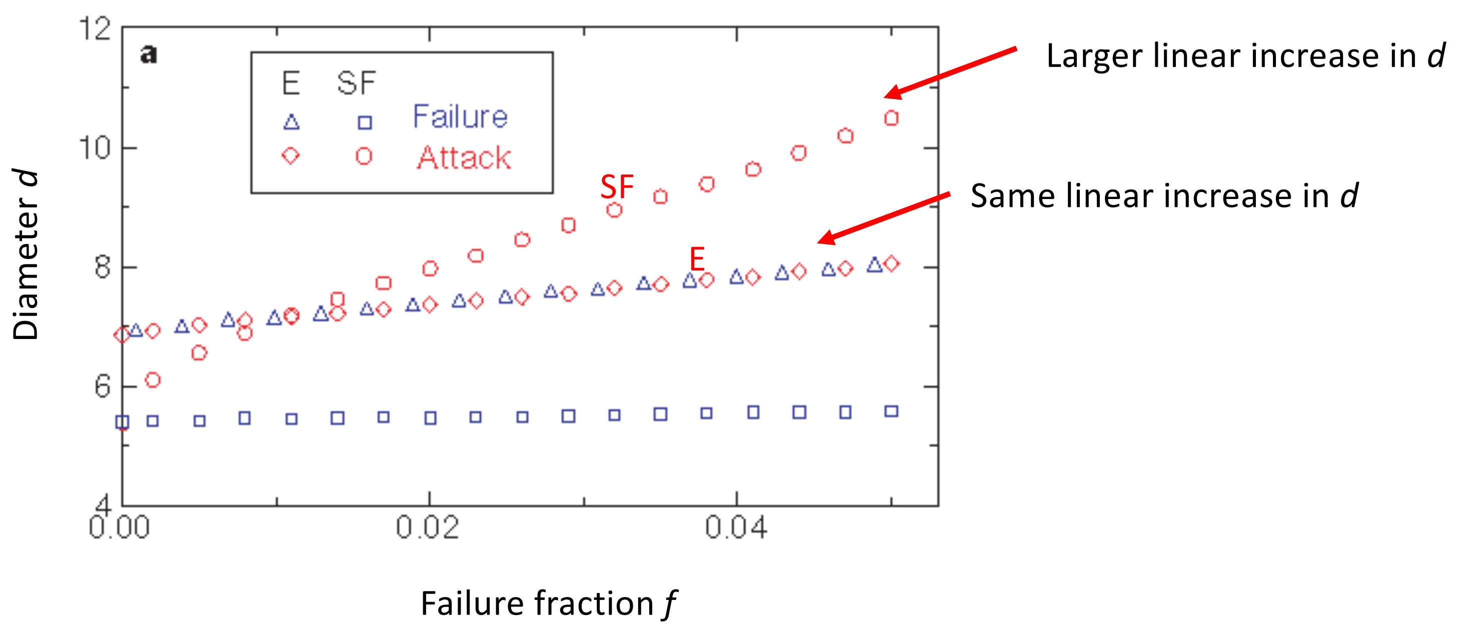
Failure fraction $f\,(= 1 - \phi)$

Linear increase in d

Almost no change

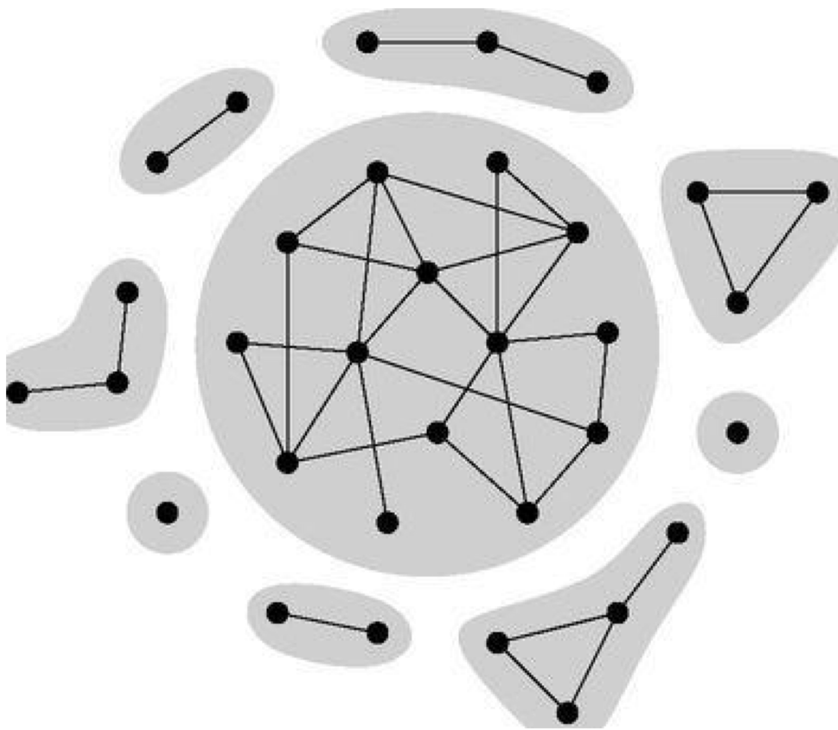
Increase in diameter means that the network is becoming disconnected

But the Scale-Free Network is **More Vulnerable** to Attack Than the Exponential Network



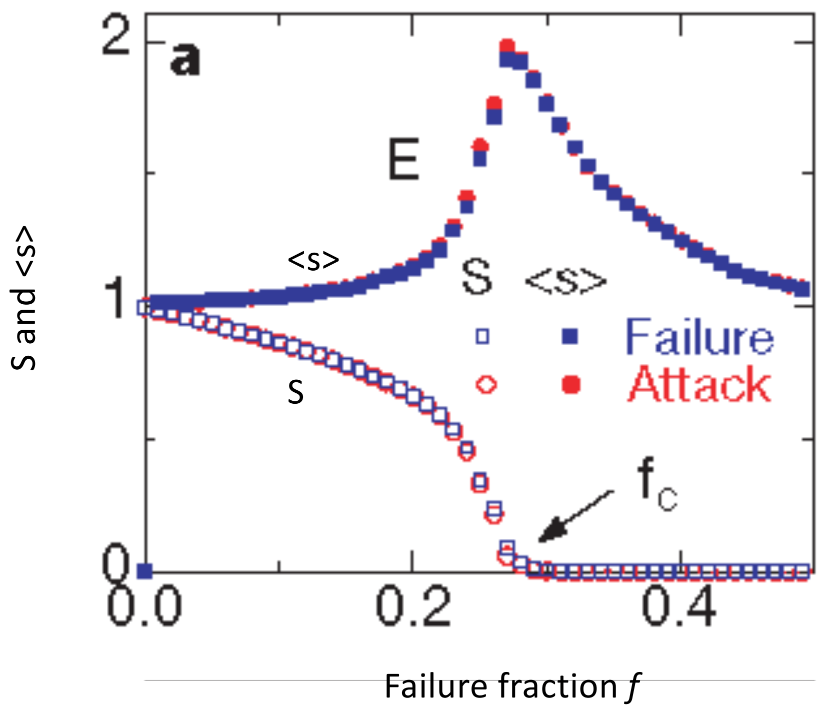
Attack means sequentially taking out the nodes with highest degree

Giant Components



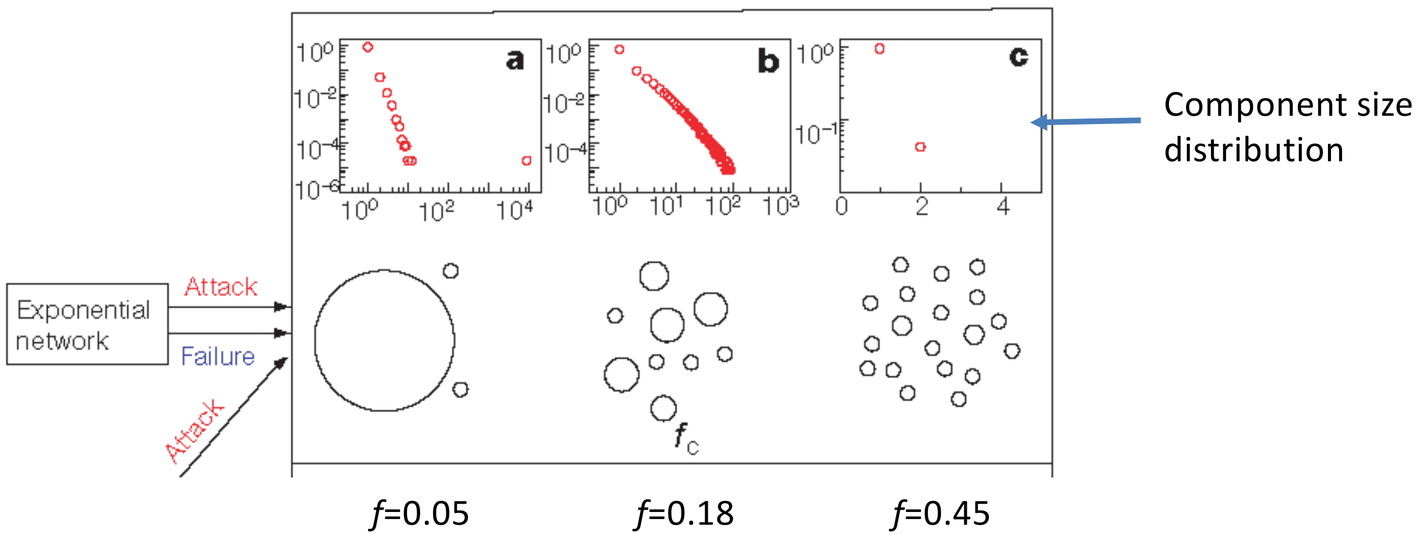
S = Fraction of nodes in the giant component
<s> = mean size of an isolated (non-giant) component

In an Exponential Network There is **No Difference** Between Failure and Attack on Giant Component Size

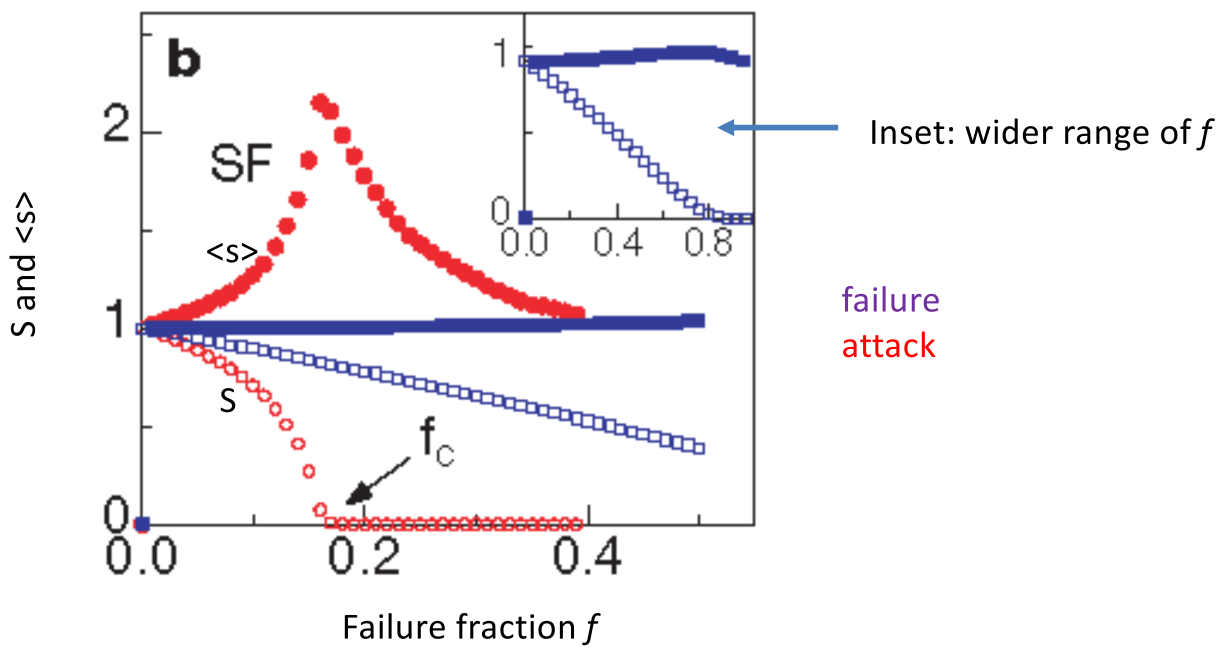


f_c is the percolation threshold

In an Exponential Network the Network Disintegration is **Roughly Homogenous**

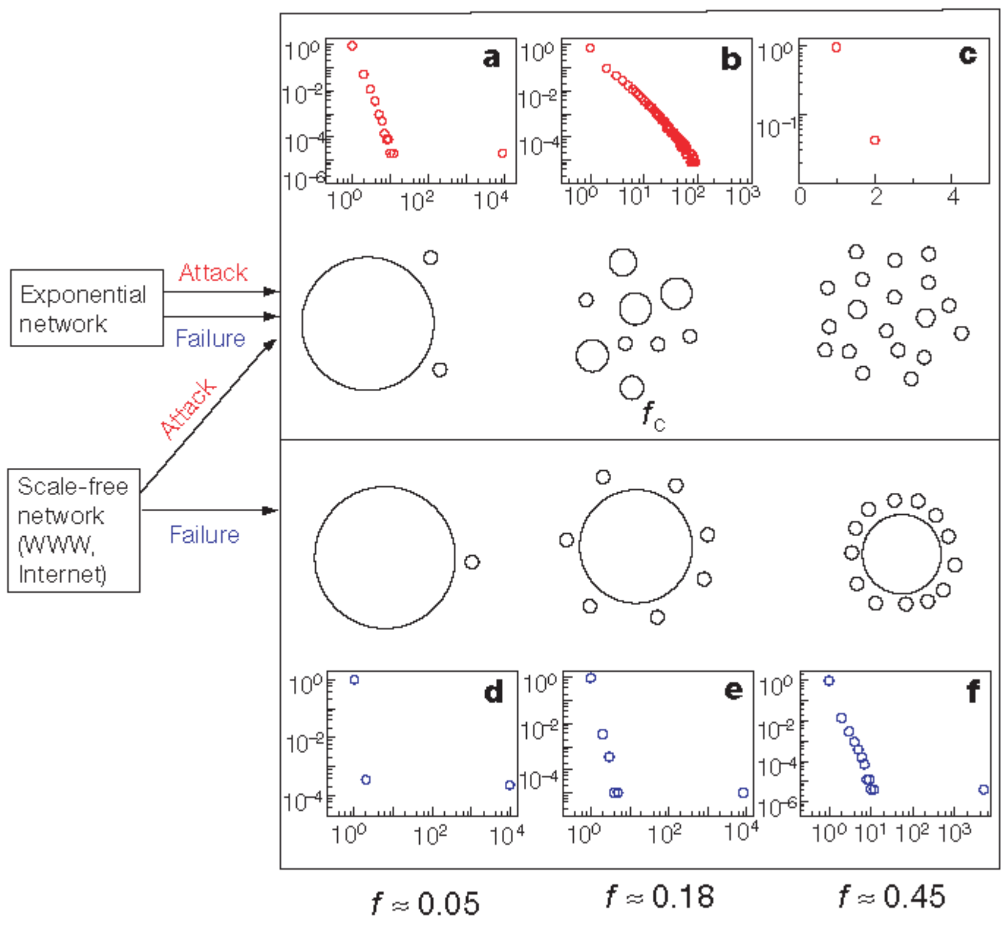


In a Scale-Free Network There is a **Big Difference** Between Failure and Attack on Giant Component Size



f_c is the percolation threshold

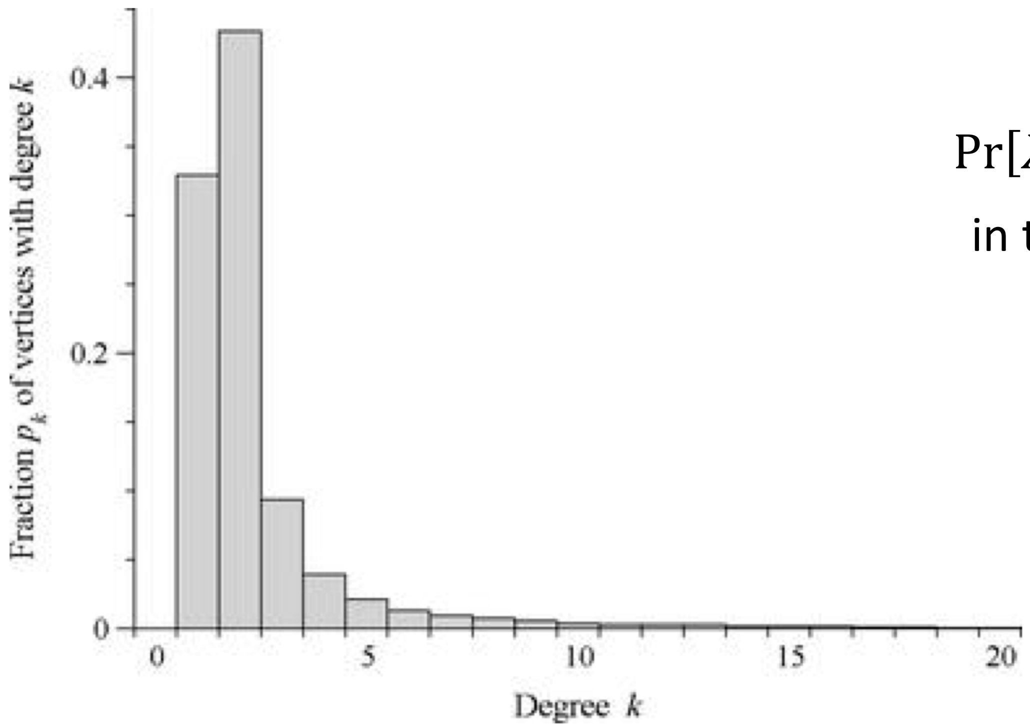
In a Scale-Free Network the Network Disintegration in Response to Failure is **<u>Very Heterogeneous</u>**



Homogenous disintegration

Heterogeneous disintegration

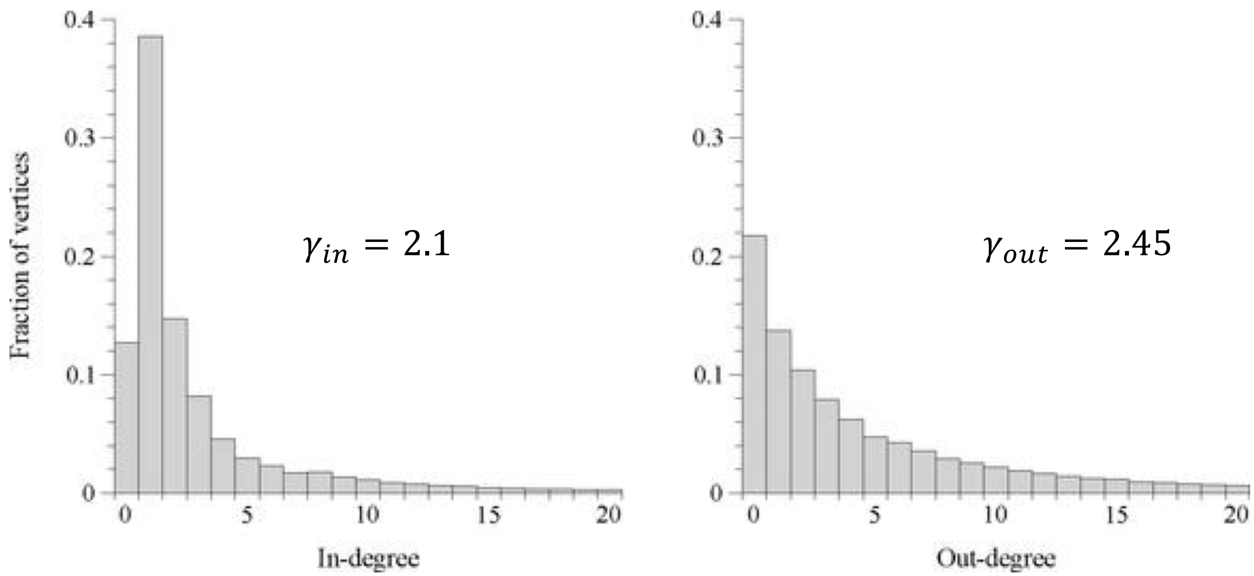
Degree Distribution of the Internet Obeys a Power Law for k>1



$$\Pr[X = k] = Ck^{-2.48}$$

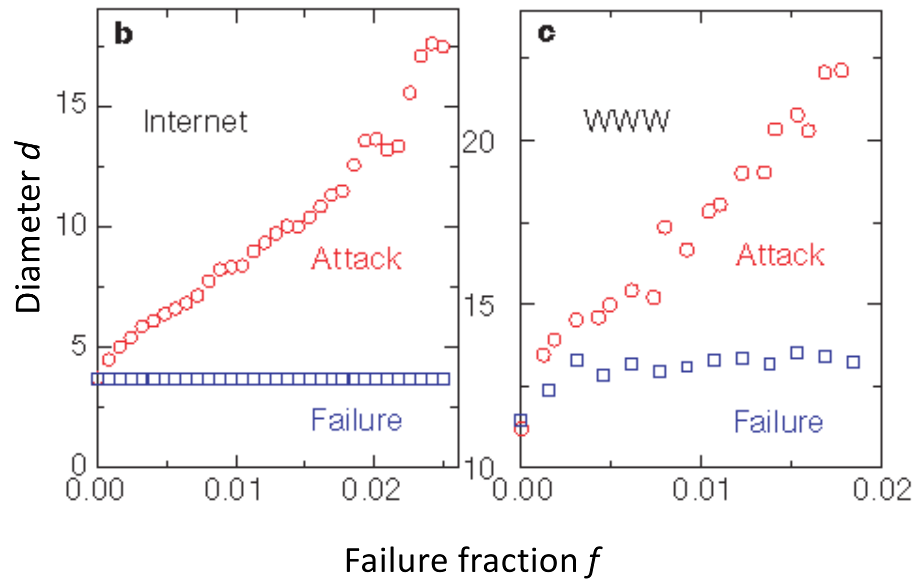
in the year 2000

In-Degree and Out-Degree Distributions of the WWW Obey Power Laws



Exponent data from the year 2000

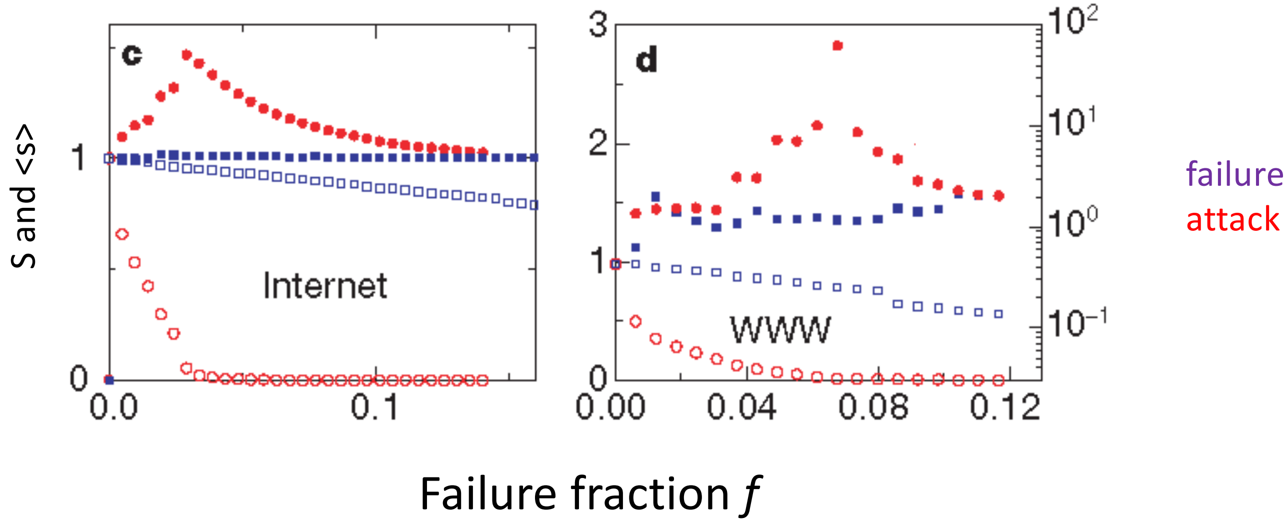
The Internet and WWW are Resilient to Random Failure, but Sensitive to Attack



Internet data: 6,209 nodes and 12,200 edges

WWW data: 325,729 nodes and 1,498,353 edges

In the Internet and WWW There is a **Big Difference** Between Failure and Attack on Giant Component Size



The End