Network Science (VU) (706.703)

Empirical Analysis of Networks

Denis Helic

ISDS, TU Graz

November 24, 2023

Outline

Basic Statistics

	Network	Type	n	m	c	S	ℓ	α	C	C_{WS}	r	Ref(s).
Social	Film actors	Undirected	449 913	25 516 482	113.43	0.980	3.48	2.3	0.20	0.78	0.208	16, 323
	Company directors	Undirected	7 673	55 392	14.44	0.876	4.60	–	0.59	0.88	0.276	88, 253
	Math coauthorship	Undirected	253 339	496 489	3.92	0.822	7.57	–	0.15	0.34	0.120	89, 146
	Physics coauthorship	Undirected	52 909	245 300	9.27	0.838	6.19	–	0.45	0.56	0.363	234, 236
	Biology coauthorship	Undirected	1 520 251	11 803 064	15.53	0.918	4.92	–	0.088	0.60	0.127	234, 236
	Telephone call graph	Undirected	47 000 000	80 000 000	3.16			2.1				9, 10
	Email messages	Directed	59 812	86 300	1.44	0.952	4.95	1.5/2.0		0.16		103
	Email address books	Directed	16 881	57 029	3.38	0.590	5.22	–	0.17	0.13	0.092	248
	Student dating	Undirected	573	477	1.66	0.503	16.01	–	0.005	0.001	−0.029	34
	Sexual contacts	Undirected	2 810					3.2				197, 198
Information	WWW `nd.edu`	Directed	269 504	1 497 135	5.55	1.000	11.27	2.1/2.4	0.11	0.29	−0.067	13, 28
	WWW AltaVista	Directed	203 549 046	1 466 000 000	7.20	0.914	16.18	2.1/2.7				56
	Citation network	Directed	783 339	6 716 198	8.57			3.0/–				280
	Roget's Thesaurus	Directed	1 022	5 103	4.99	0.977	4.87	–	0.13	0.15	0.157	184
	Word co-occurrence	Undirected	460 902	16 100 000	66.96	1.000		2.7		0.44		97, 116
Technological	Internet	Undirected	10 697	31 992	5.98	1.000	3.31	2.5	0.035	0.39	−0.189	66, 111
	Power grid	Undirected	4 941	6 594	2.67	1.000	18.99	–	0.10	0.080	−0.003	323
	Train routes	Undirected	587	19 603	66.79	1.000	2.16	–		0.69	−0.033	294
	Software packages	Directed	1 439	1 723	1.20	0.998	2.42	1.6/1.4	0.070	0.082	−0.016	239
	Software classes	Directed	1 376	2 213	1.61	1.000	5.40	–	0.033	0.012	−0.119	315
	Electronic circuits	Undirected	24 097	53 248	4.34	1.000	11.05	3.0	0.010	0.030	−0.154	115
	Peer-to-peer network	Undirected	880	1 296	1.47	0.805	4.28	2.1	0.012	0.011	−0.366	6, 282
Biological	Metabolic network	Undirected	765	3 686	9.64	0.996	2.56	2.2	0.090	0.67	−0.240	166
	Protein interactions	Undirected	2 115	2 240	2.12	0.689	6.80	2.4	0.072	0.071	−0.156	164
	Marine food web	Directed	134	598	4.46	1.000	2.05	–	0.16	0.23	−0.263	160
	Freshwater food web	Directed	92	997	10.84	1.000	1.90	–	0.20	0.087	−0.326	209
	Neural network	Directed	307	2 359	7.68	0.967	3.97	–	0.18	0.28	−0.226	323, 328

Table 8.1: Basic statistics for a number of networks. The properties measured are: type of network, directed or undirected; total number of vertices n; total number of edges m; mean degree c; fraction of vertices in the largest component S (or the largest weakly connected component in the case of a directed network); mean geodesic distance between connected vertex pairs ℓ; exponent α of the degree distribution if the distribution follows a power law (or "–" if not; in/out-degree exponents are given for directed graphs); clustering coefficient C from Eq. (7.41); clustering coefficient C_{WS} from the alternative definition of Eq. (7.44); and the degree correlation coefficient r from Eq. (7.82). The last column gives the citation(s) for each network in the bibliography. Blank entries indicate unavailable data.

Components

- In an unidirected network, there is typically a large component that fills most of the network
- Very often over 90%
- Sometimes, it is 100%, e.g. the Internet
- Sometimes it depends also on how we collect data

Components in a directed network

- Weakly connected components correspond to components in an undirected network, i.e. we simply ignore link directions
- Otherwise, we have strongly connected components with corresponding in- and out-components
- Apart from the largest scc we have also a number of smaller ones with their in- and out-components
- Typically, all components form a so-called “bow-tie” model

Components in a directed network

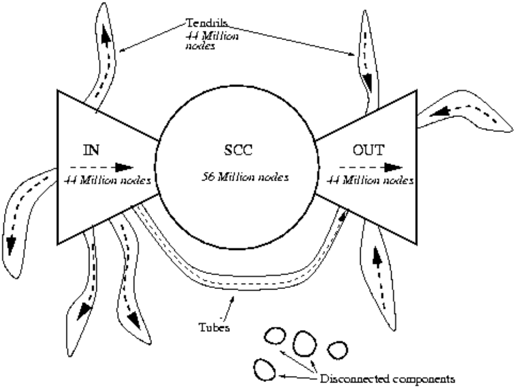


Figure: Bow-tie model of the Web graph

Small-worlds

- In many networks the typical network distance between nodes is very small
- This phenomenon was first observed in the letter-passing experiment by Milgram
- It is called *small-world effect*
- Typically, the average network distance ℓ scales as $\log n$

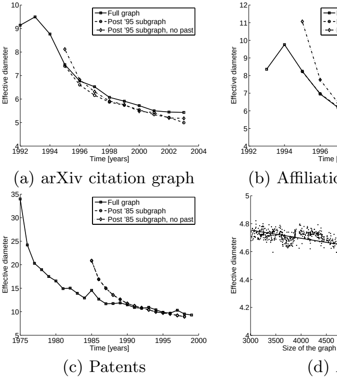
Diameter

- Sometimes we are also interested in the network diameter
- The extreme of the distance distribution, i.e. the longest shortest path in the network
- In many networks, the core of the network is very dense with the average network distance scaling as $\log \log n$
- Whereas at the periphery the diameter scales as $\log n$

Effective diameter

- Effective diameter, or 90-percentile effective diameter, i.e. 90% of shortest paths is smaller than the effective diameter
- Graphs over Time: Densification Laws, Shrinking Diameters and Possible Explanations by Leskovec et al.
- The empirical analysis has shown that when the networks grow the diameter becomes smaller

Effective diameter



(a) arXiv citation graph

(b) Affiliatio

(c) Patents

(d)

Degree distributions

- Frequency distribution of node degrees
- One of the most fundamental properties of networks
- p_k is the fraction of nodes in a network that has degree k
- p_k is also a probability that a randomly chosen node has a degree k
- Typically, we visualize a distribution with a histogram

Degree distributions

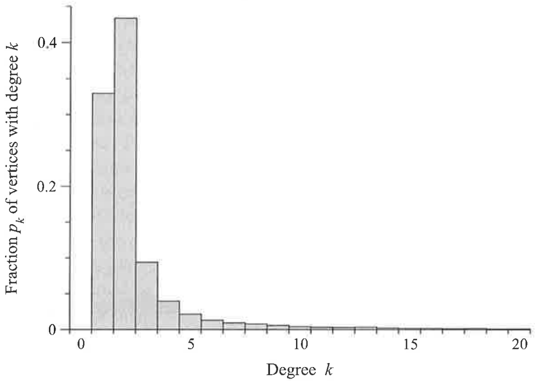


Figure: Degree distributions of the Internet graph at the level of autonomous systems

Degree distributions

- Most of the nodes have small degrees: one, two, or three
- There is a *tail* to the distribution corresponding to the high-degree nodes
- The plot cuts off but the tail is much longer
- The highest degree node is connected to about 12% of other nodes
- Such well-connected nodes are called *hubs*

Degree distributions

- It turns out that most of the real-world networks have such long-tailed distributions
- Such distributions are called *right-skewed*
- For directed networks we have two distributions
- In-degree and out-degree distribution

Degree distributions

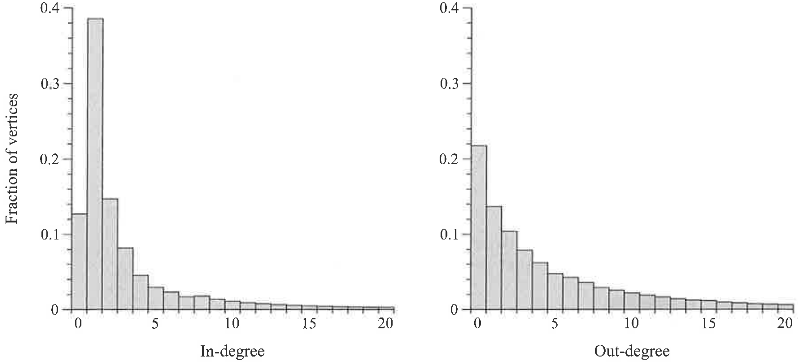


Figure: Degree distributions on the Web, from Broder et al.

Power laws and scale-free networks

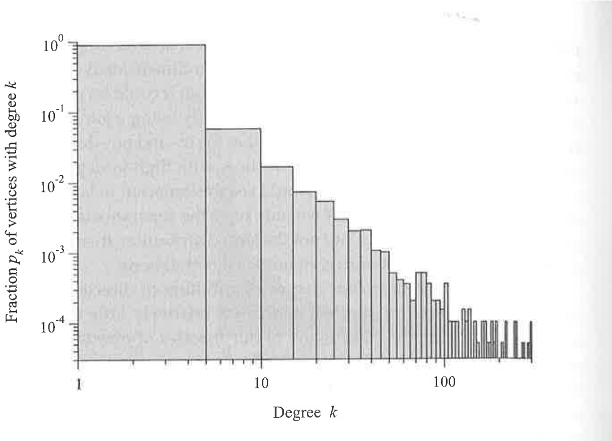


Figure: Degree distributions of the Internet graph on logarithmic scales

Power laws

- The degree distribution on logarithmic scales follows roughly a straight line

$$\ln p_k = -\alpha \ln k + c \tag{1}$$

- α and c are constants

$$p_k = Ck^{-\alpha} \tag{2}$$

- $C = e^c$ is another constant

Power laws

- Distributions of this form that vary as a power of k are called *power laws*
- This is a common pattern seen in many different networks
- The constant α is called the *exponent* of the power law
- Typical values are in the range: $2 \leq \alpha \leq 3$

Power-law (Zipf) random variable

- Power-law distribution is a very commonly occurring distribution
- Word occurences in natural language
- Friendships in a social network
- Links on the web
- PageRank, etc.

Power-law (Zipf) random variable

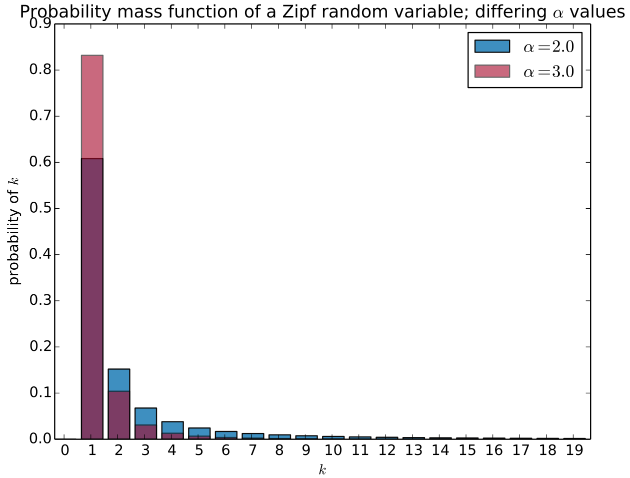
PMF

$$p(k) = \frac{k^{-\alpha}}{\zeta(\alpha)}$$

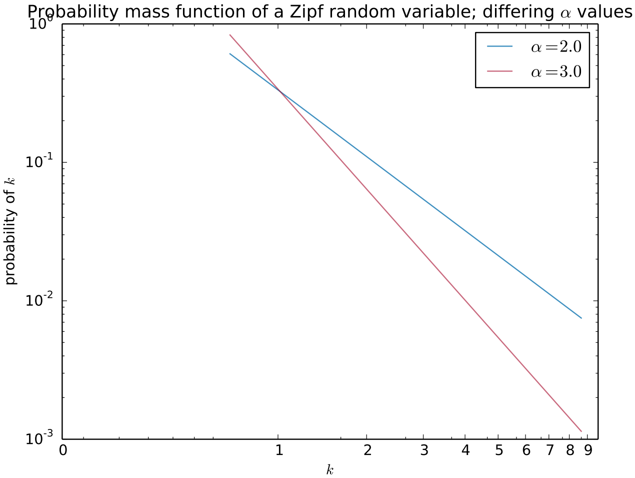
- $k \in \mathbb{N}$, $k \geq 1$, $\alpha > 1$
- $\zeta(\alpha)$ is the Riemann zeta function

$$\zeta(\alpha) = \sum_{k=1}^{\infty} k^{-\alpha}$$

Power-law (Zipf) random variable



Power-law (Zipf) random variable



Power-law (Pareto) random variable

- Power-law distribution is a very commonly occurring distribution
- 80%-20% rule
- Wealth distribution
- The sizes of the human settlements
- File size of internet traffic, etc.

Power-law (Pareto) random variable

PDF

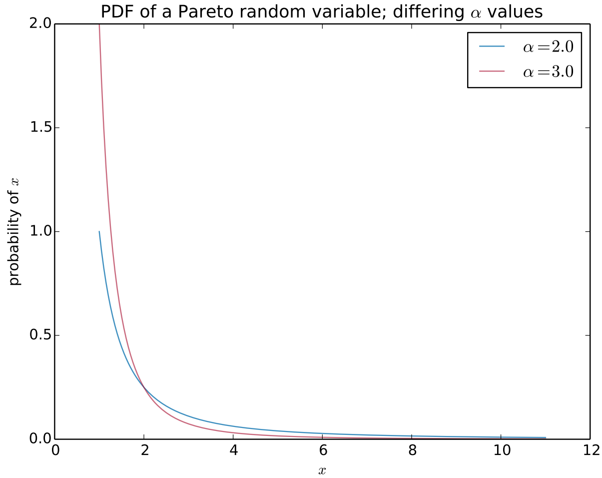
$$f(x) = \begin{cases} (\alpha - 1)\frac{x_{min}^{\alpha-1}}{x^{\alpha}}, x \geq x_{min} \\ 0, x < x_{min} \end{cases}$$

- $\alpha > 1$ is the exponent of the power-law distribution

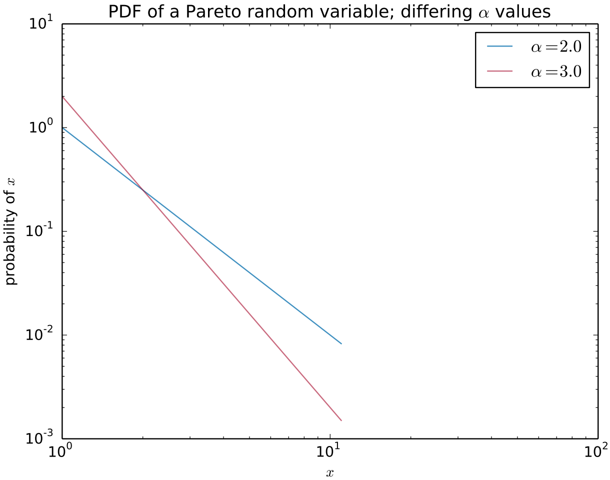
CDF

$$f(x) = \begin{cases} 1 - (\frac{x_{min}}{x})^{\alpha-1}, x \geq x_{min} \\ 0, x < x_{min} \end{cases}$$

Power-law (Pareto) random variable



Power-law (Pareto) random variable



Power laws

- Degree distributions do not follow power law equation over their entire range
- For example, for small k we typically observe some deviation
- Thus, power laws are typically observed in the tail for high degrees
- Sometimes, there is also deviation in the tail because there is some cut-off that limits the maximum degree of nodes
- Network with power law degree distributions are called *scale-free* networks

Detecting power laws

- Another common solution to visualizing power laws is to construct *cumulative distribution function*

$$P_k = \sum_{k'=k}^{\infty} p_{k'} \tag{3}$$

- P_k is the fraction of nodes that have degree k or higher

Detecting power laws

- Suppose the degree distribution p_k follows power law in the tail
- $p_k = Ck^{-\alpha}$, for $k \geq k_{min}$, for some k_{min}. Then for $k \geq k_{min}$:

$$P_k = \sum_{k'=k}^{\infty} k'^{-\alpha} \simeq C \int_{k}^{\infty} k'^{-\alpha} \mathrm{d}k' = \frac{C}{\alpha - 1} k^{-(\alpha-1)} \tag{4}$$

- Approximation of the sum by the integral is possible if we assume $\alpha > 1$ and is reasonable since the power law slowly varies for large k

Detecting power laws

- Thus, cumulative degree distribution is also a power law but with an exponent $\alpha - 1$
- We can visualize the cumulative degree distribution on log-log scales and look for the straight line behavior
- This has some advantages over visualizing p_k
- E.g. we do not need to bin the histogram and throw away information

Cumulative degree distributions

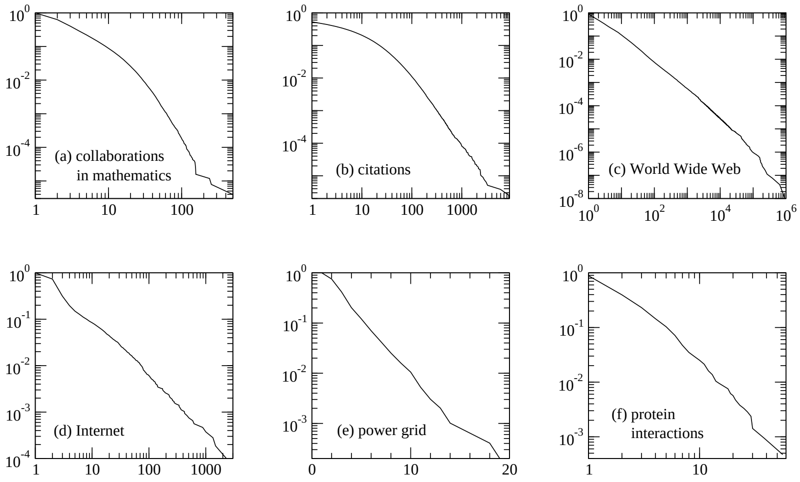


Figure: Cumulative degree distributions on logarithmic scales

Cumulative degree distributions

- Cumulative degree distribution is easy to calculate
- The number of nodes greater or equal to that of the rth-highest degree is r
- The fraction of nodes with degree greater or equal to that of the rth-highest degree is r/n and that is P_k
- Thus, we calculate degrees, sort them in descending order and then number them from 1 to n
- These numbers are *ranks* r_i and we plot $\frac{r_i}{n}$ as a function of k_i

Cumulative degree distributions

Degree k	Rank r	$P_k = \frac{r}{n}$
4	1	0.1
3	2	0.2
3	3	0.3
2	4	0.4
2	5	0.5
2	6	0.6
2	7	0.7
1	8	0.8
1	9	0.9
1	10	1.0

Table: Example of cumulative degree distribution for degrees $\{0,1,1,2,2,2,2,3,3,4\}$

Cumulative degree distributions

- Cumulative distribution have some disadvantages
- Successive points on a cumulative plot are not independent
- It is not valid to extract the exponent by fitting the slope of the line
- E.g. least squares method assumes independence of between the data points
- Also, which line to fit?

Parameter estimation

- It is better to calculate α directly from the data

$$\alpha = 1 + N \left[\sum_i \ln \frac{k_i}{k_{min} - \frac{1}{2}} \right]^{-1} \tag{5}$$

- where, k_{min} is the minimum degree for which the power low holds and N is the number of nodes with $k \geq k_{min}$

Parameter estimation

- Statistical error

$$\sigma = \frac{\alpha - 1}{\sqrt{N}} \tag{6}$$

- The derivation is based on *maximum likelihood* techniques
- Power law distributions in empirical data by Clauset et al.
- http://tuvalu.santafe.edu/~aaronc/powerlaws/

Likelihood

- We observe some data, e.g. number of heads in m experiments with n coin flips
- We **choose** a probabilistic model to describe the dataset
- E.g. a Binomial r.v. with parameters (p, n)
- p is the probability of heads on a single coin flip

PMF

$$p(x) = \binom{n}{x}(1-p)^{n-x}p^x \tag{7}$$

Likelihood

- Let us denote with $X_1, \ldots, X_m$ r.v. associated with our m experiments
- Each of them is a Binomial r.v. with parameters (p, n)
- They are mutually independent
- Independent and identically distributed (i.i.d.)

Likelihood

- We are interested in probability of observing the results of our m experiments
- For a single experiment:

Probability of a single experiment

$$p(x_i) = \binom{n}{x_i}(1-p)^{n-x_i}p^{x_i} \tag{8}$$

Likelihood

- For all m experiments (since experiments are i.i.d. r.v.)

Probability of all experiments

$$p(x_1, \ldots, x_m | p) = \prod_{i=1}^{m} \binom{n}{x_i} (1-p)^{n-x_i} p^{x_i} \tag{9}$$

- This probability is called **likelihood**
- It is the probability of data given the parameter p
- Another name is likelihood function (function of parameter p)

Log-likelihood

- Typically, we take a logarithm and work with logs since it simplifies the analysis

Log-likelihood

$$\mathcal{L}(p) = ln(\prod_{i=1}^{m} \binom{n}{x_i}(1-p)^{n-x_i}p^{x_i}) \quad (10)$$

$$= \sum_{i=1}^{m}(ln\binom{n}{x_i} + (n-x_i)ln(1-p) + x_i ln(p)) \quad (11)$$

$$= \sum_{i=1}^{m} ln\binom{n}{x_i} + ln(p)\sum_{i=1}^{m} x_i + ln(1-p)(mn - \sum_{i=1}^{m} x_i) \quad (12)$$

Maximum Likelihood Estimation (MLE)

- Now, we are interested in p that most likely generated the data
- The data are most likely to have been generated by the model with p that maximizes the log-likelihood function
- Setting $\frac{d\mathcal{L}}{dp} = 0$ and solving for p we obtain the *maximum likelihood estimate*

MLE

$$\frac{d\mathcal{L}}{dp} = \frac{1}{p}\sum_{i=1}^{m} x_i - \frac{1}{1-p}(mn - \sum_{i=1}^{m} x_i) = 0 \tag{13}$$

$$p = \frac{\sum_{i=1}^{m} x_i}{mn} = \frac{1}{m}\sum_{i=1}^{m}\frac{x_i}{n} \tag{14}$$

Parameter estimation

- We consider the continuous power law distribution

$$p(x) = \frac{\alpha - 1}{x_{min}} \left(\frac{x}{x_{min}} \right)^{-\alpha} \tag{15}$$

- Given a data set with n observations $x_i > x_{min}$ we would like to know the value of α that is most likely to have generated the data

Parameter estimation

- The probability that the data are drawn from the model

$$p(x|\alpha) = \prod_{i=1}^{n} \frac{\alpha - 1}{x_{min}} \left(\frac{x_i}{x_{min}} \right)^{-\alpha} \tag{16}$$

- This probability is called *likelihood* of the data given model

Parameter estimation

- The data are most likely to have been generated by the model with α that maximizes this function
- Commonly, we work with *log-likelihood* $\mathcal{L}$
- $\mathcal{L}$ has the maximum at the same place likelihood

$$\mathcal{L} = \ln p(x|\alpha) = \ln \prod_{i=1}^{n} \frac{\alpha - 1}{x_{min}} \left(\frac{x_i}{x_{min}} \right)^{-\alpha} \tag{17}$$

Parameter estimation

$$\mathcal{L} = n\ln(\alpha - 1) - n\ln x_{min} - \alpha \sum_{i=1}^{n} \ln \frac{x}{x_{min}} \tag{18}$$

- Setting $\frac{\partial \mathcal{L}}{\partial \alpha} = 0$ and solving for α we obtain the *maximum likelihood estimate*

$$\hat{\alpha} = 1 + n \left[\sum_{i=1}^{n} \ln \frac{x_i}{x_{min}} \right]^{-1} \tag{19}$$

Properties of power law distributions

- Normalization
- The constant C that appears in the power law equation is determined by the normalization requirement

$$\sum_{k=1}^{\infty} p_k = 1 \tag{20}$$

- $k^{-\alpha} = \infty$, for $k = 0$ and therefore we start at $k = 1$

Properties of power law distributions

$$C \sum_{k=1}^{\infty} k^{-\alpha} = 1 \tag{21}$$

$$C = \frac{1}{\sum_{k=1}^{\infty} k^{-\alpha}} = \frac{1}{\zeta(\alpha)} \tag{22}$$

- $\zeta(\alpha)$ is the Riemann zeta function

Properties of power law distributions

- Correctly normalized power law distribution for $k > 0$ and $p_0 = 0$

$$p_k = \frac{k^{-\alpha}}{\zeta(\alpha)} \tag{23}$$

- If the power law behavior holds only for $k > k_{min}$ we obtain (with $\zeta(\alpha, k_{min})$ being incomplete zeta function)

$$p_k = \frac{k^{-\alpha}}{\sum_{k=k_{min}}^{\infty} k^{-\alpha}} = \frac{k^{-\alpha}}{\zeta(\alpha, k_{min})} \tag{24}$$

Properties of power law distributions

- Alternatively, we can approximate the sum with an integral

$$C \simeq \frac{1}{\int_{k_{min}}^{\infty} k^{-\alpha} \mathrm{d}k} = (\alpha - 1) k_{min}^{\alpha - 1} \tag{25}$$

$$p_k \simeq \frac{\alpha - 1}{k_{min}} \left(\frac{k}{k_{min}} \right)^{-\alpha} \tag{26}$$

Properties of power law distributions

- Top-heavy distributions
- Another interesting property is the fraction of links that connect to the nodes with the highest degrees
- For a pure power law W is a fraction of links attached to a fraction P of the highest degree nodes

$$W = P^{\frac{\alpha-2}{\alpha-1}} \tag{27}$$

Properties of power law distributions

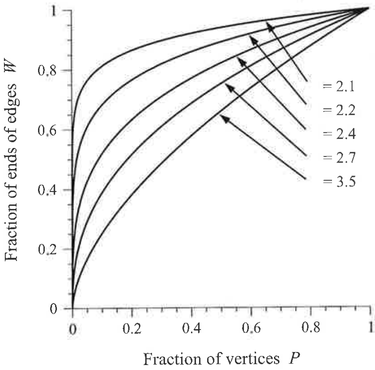


Figure: Lorenz curves for power law networks

Properties of power law distributions

- The curves have a very fast initial increase (especially if α is slightly over 2)
- This means that a large fraction of links is connected to a small fraction of the highest degree nodes
- For example, in-degrees on the Web have $k_{min} = 20$ and $\alpha = 2.2$
- For $P = 0.5$ we have $W = 0.89$, for $W = 0.5$ we have $P = 0.015$

Properties of power law distributions

- These calculations assume perfect power law
- We can still calculate W and P directly from the data
- For example, on the Web for $W = 0.5$ we have $P = 0.011$
- Similarly, in citation networks for $W = 0.5$ we have $P = 0.083$

Centralities

- Eigenvector centralities have often a highly right-skewed distributions
- Also, variants of the eigenvector centralities such as PageRank exhibit often power law behavior
- E.g. the Internet, WWW, or citation networks
- Betweenness centrality also tends to have right-skewed distributions

Centralities

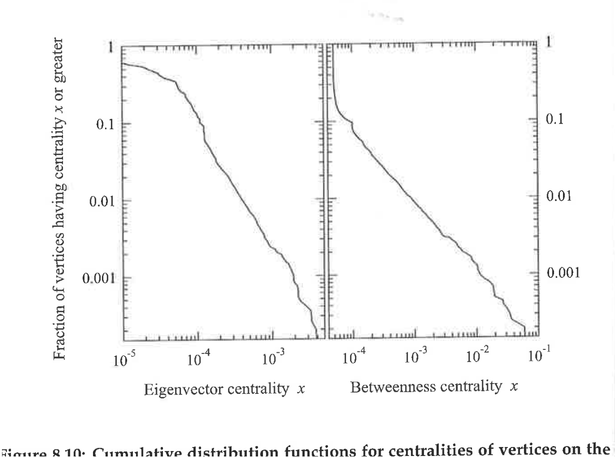


Figure: Cummulative distibutions of centralities on the Internet

Centralities

- An exception to this pattern is closeness centrality
- Values for closeness centralities are limited by 1 at the lower end and $\log n$ at the upper end
- Therefore their distributions cannot have a long tail
- Typically, closeness centrality distributions are multimodal, whit multiple peaks and dips

Centralities

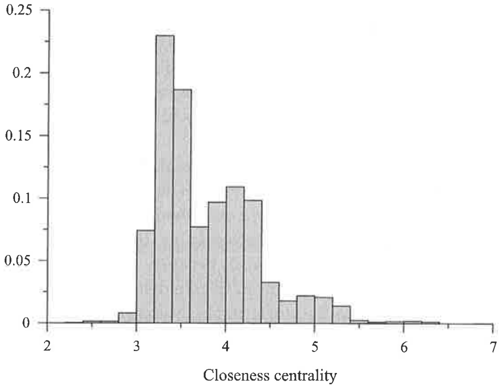


Figure: Histogram of closeness centralities on the Internet

Global clustering coefficient

- The clustering coefficient measures the average probability that two neighbors of a node are themselves neighbors
- It measures the density of triangles in the networks
- In real networks the clustering coefficient takes values in the order of tens of percent, e.g. 10% or even up to 60%
- This is much larger than what we would expect if the links are created by chance, e.g. 0.01%
- E.g. in collaboration networks of physicists expectation is 0.23% but the real value is 45%

Global clustering coefficient

- This large difference is indicative of social effects
- For example, it might be that people introduce the pairs of their collaborators to each other
- In social networks this process is called *triadic closure*
- An open triad of nodes is closed by the introduction of the last third link
- We can study the triadic closure processes directly if we have different version of datasets in time
- E.g. a study showed that it is much more likely (45 times) for people to collaborate in future if they had common collaborators in the past

Global clustering coefficient

- In some networks we have the opposite phenomenon
- The expected value of clustering exceeds the observed one
- For example, on the Internet we measure 1.2% and the expected value is 84%
- Thus, on the Internet we have mechanisms that prevent forming of triangles
- On the Web the measured clustering coefficient is of the order of the expected one

Global clustering coefficient

- It is not completely clear why different types of networks exhibit such different behaviors in respect to the clustering coefficient
- One theory connects these observations with the formation of communities in networks
- Social networks tend also to have positive degree correlations as opposed to other types of networks
- Thus, in social networks homophily and assortative mixing by degree plays a more important role than in other networks
- This tends to formation of communities and therefore the clustering coefficient becomes greater

Local clustering coefficient

- Local clustering coefficient of a node i is the fraction of neighbors of i that are themselves neighbors
- In many networks there is a phenomenon that high degree nodes tend to have lower local clustering
- One possible explanation for this behavior is that nodes tend to form highly connected communities
- Communities of low degree nodes are smaller that work as small disconnected networks, i.e. cliques
- Probability that higher degree nodes form such huge cliques is rather small

Local clustering coefficient

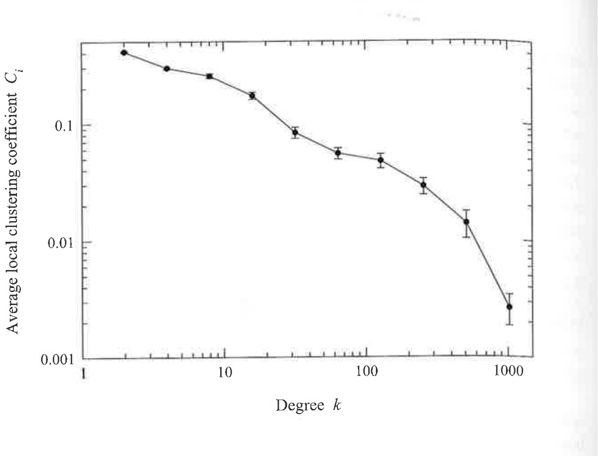


Figure: Local clustering as a function of degree on the Internet

Assortative mixing by degree

- Assortative mixing by degree can be quantified by the correlation coefficient r
- Typically, r is not of a large magnitude in real world networks
- There is clear tendency of social networks to have positive r (homophily)
- Technological, information, biological networks tend to have negative r
- Simple graphs bias: the number of links between high-degree nodes is limited because they connect to low degree nodes
- Social networks: communities

Network analysis project

- Software
- C++: SNAP http://snap.stanford.edu/
- Python: NetworkX http://networkx.github.io/
- Python wrapper for Boost: Graph-Tool http://graph-tool.skewed.de/
- Python, R, C: IGraph https://igraph.org/

Network analysis project

- SNAP: http://snap.stanford.edu/
- KONECT: http://konect.cc/
- Dataset of choice
- From SNAP or KONECT Web site
- Your own dataset