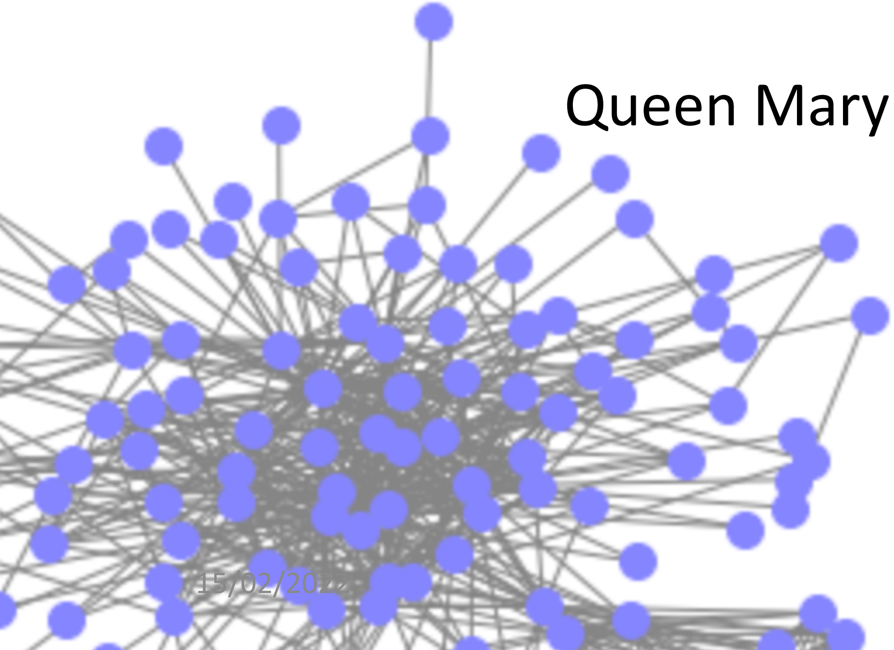
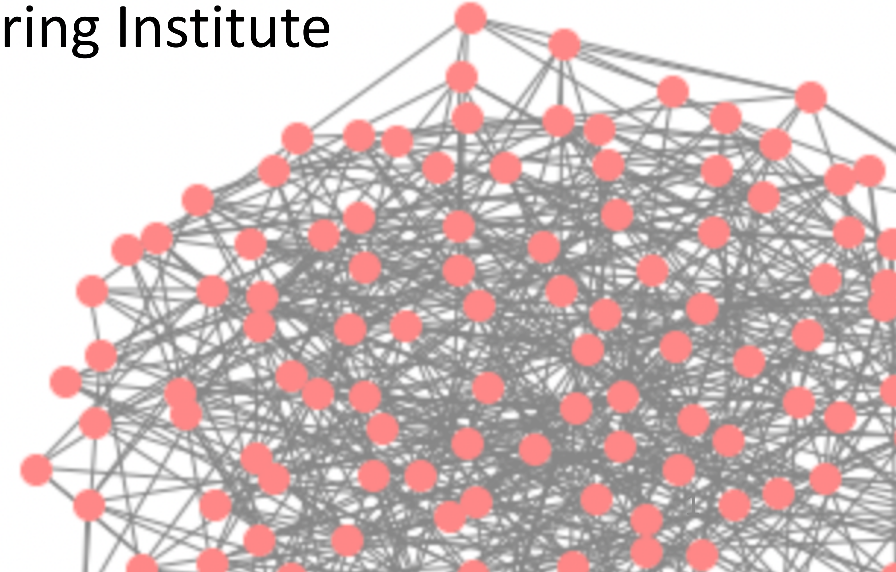
Networks and Random Graphs

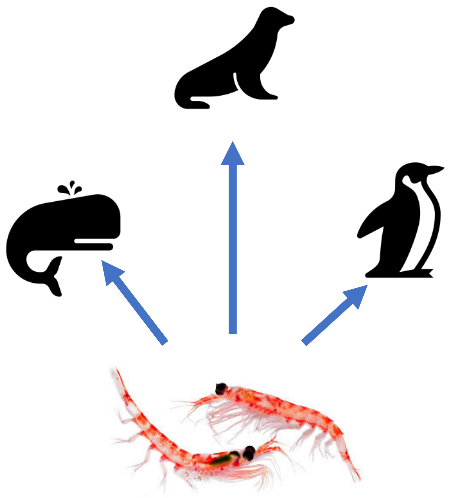
Naomi Arnold

Queen Mary University of London, Alan Turing Institute



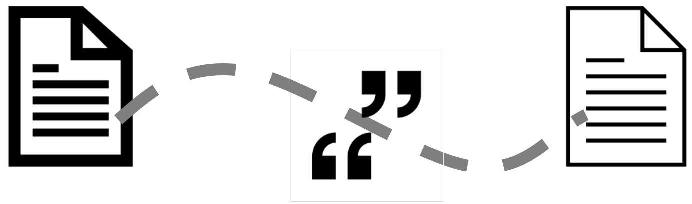


My journey with networks

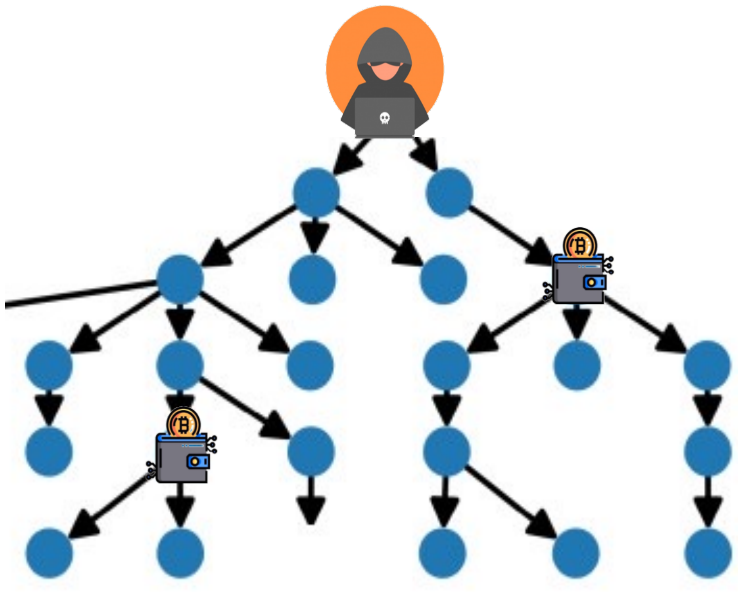


Food webs: nodes are species and edges are "being eaten"

Online social networks: nodes are users and edges interactions



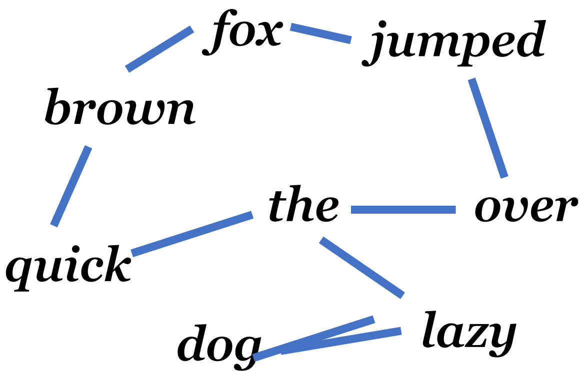
Citation networks: nodes are papers and edges are citations



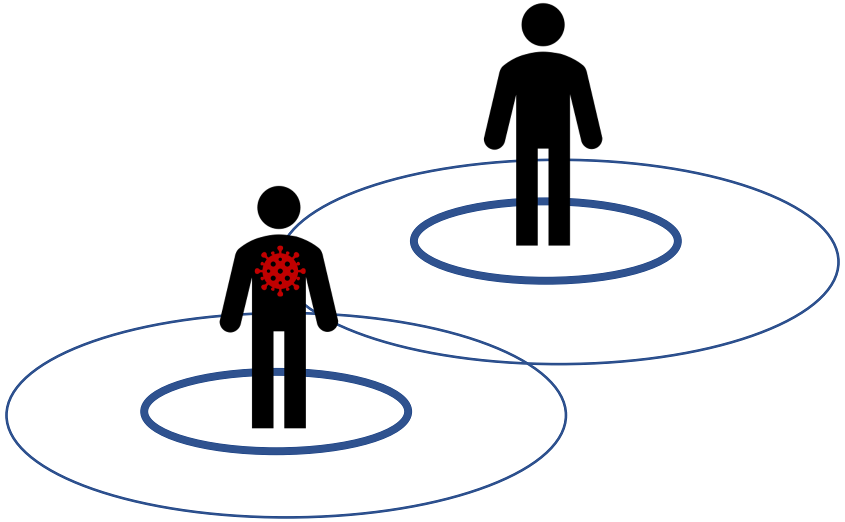
Bitcoin network: nodes are wallets and edges transactions

Other types of graph

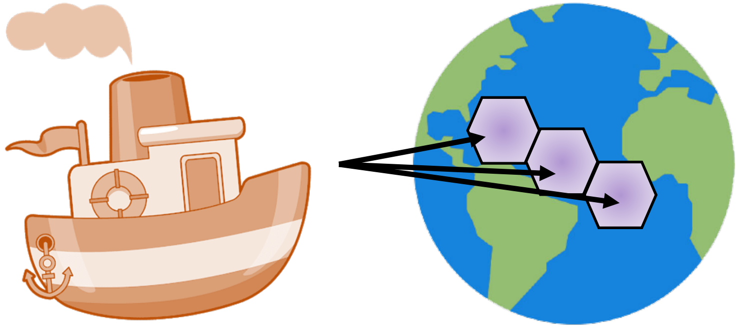
"The quick brown fox jumped over the lazy dog"



Word co-occurrence network (NLP)



Co-location network – Covid Track and Trace



Co-location network – large vessels

In this tutorial we will cover

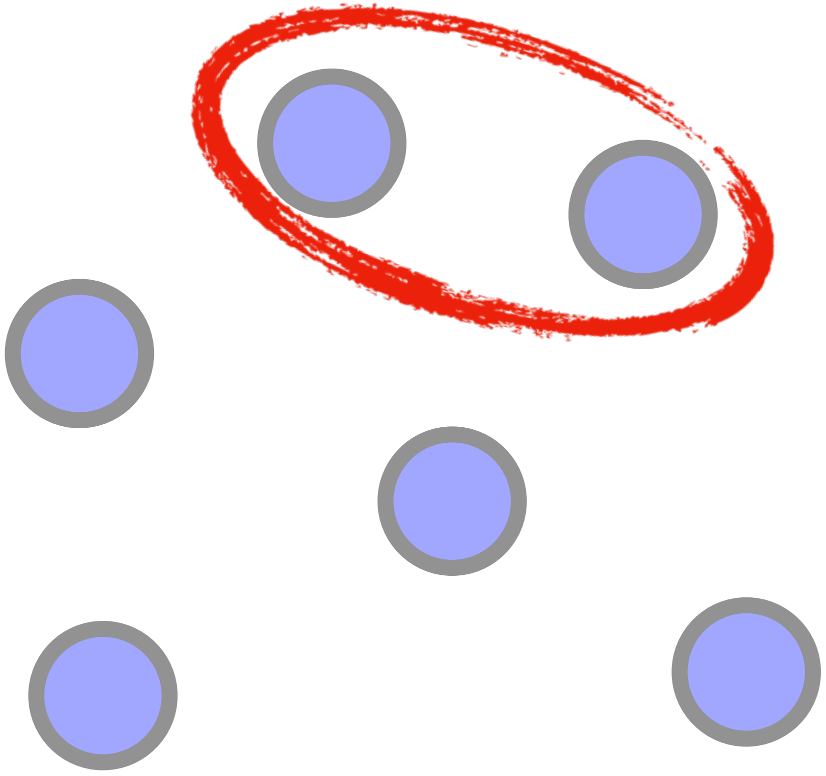
- What is a **random graph** and why are they useful?
- The **Erdős-Rényi** random graph model: the theory and implementation in Python NetworkX
- **Differences** between real and random graphs
- Watts-Strogatz **small-world** model

Why random graphs?

Why random graphs?

- 🔧 **Null model for network features** – test whether a feature of a network dataset is really a "feature" or a common network property
- 🥸 **Replacement for sensitive data** – e.g. financial transactions, Covid track and trace contact networks
- ⁉️ **Modelling unknown networks** – many systems just don't have datasets available e.g. offline friendship networks, brain connectomes

Erdos-Renyi $G(n, p)$ Model

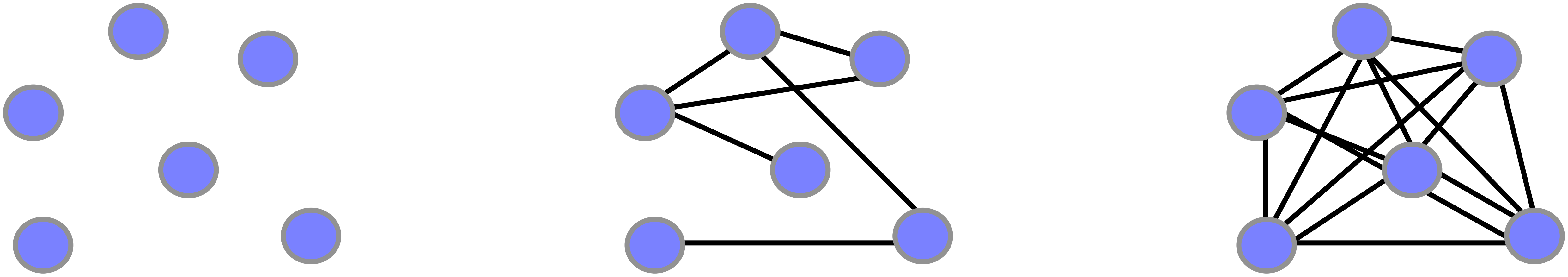


1. Start with an empty graph of n nodes

2. Acquire a biased coin with head probability p

3. For each pair of nodes, do a coin toss. If heads, draw an edge between them. If not, move on.

Erdos-Renyi G(n,p) Model

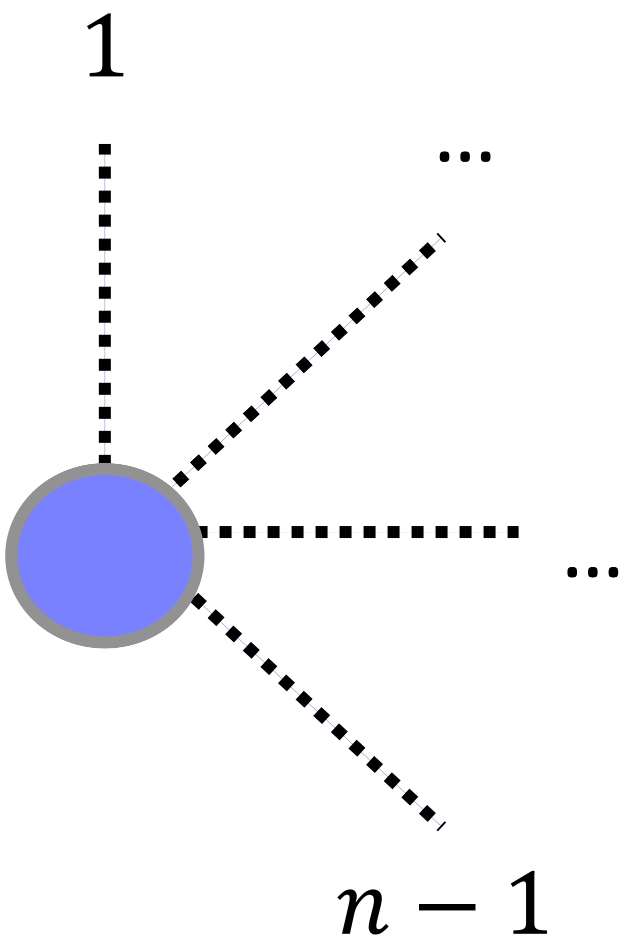


$p = 0$ $\longrightarrow$ $p = 1$

Increasing p

What are some properties of random graphs?

Expected degree of nodes in ER networks



For each node, there are $n-1$ others in the graph it could connect to.

Each of those connections can happen with probability p

So average degree is $\underline{(n-1)p}$, or approximately $\underline{np}$

Expected Clustering coefficient in ER networks

Node clustering coefficient C(v)

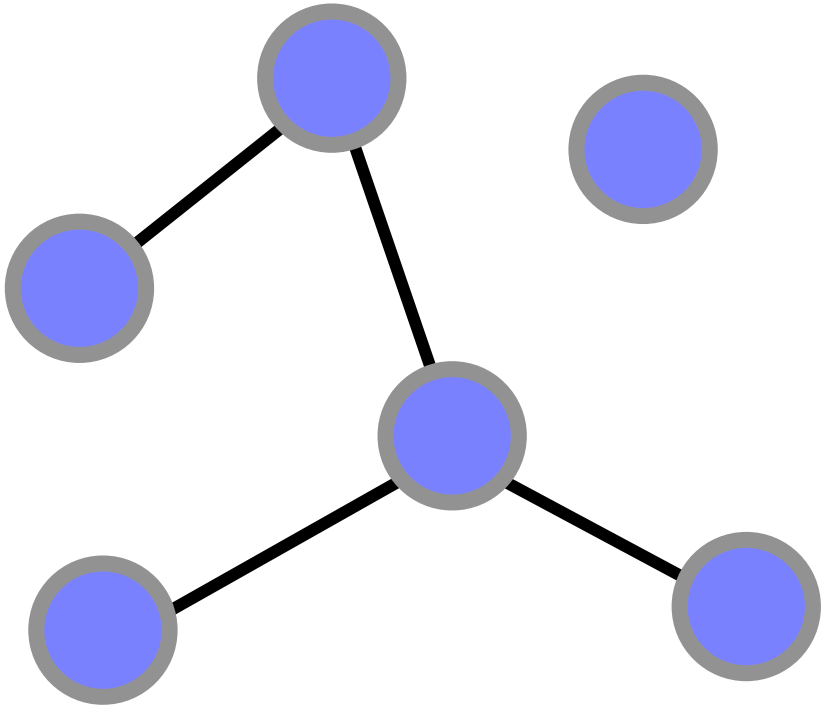
$$C(v) = \frac{|\{(u,w)|u,w \in N(v)\}|}{\frac{1}{2}k(v)(k(v)-1)} = \frac{p * \frac{1}{2}k(v)(k(v)-1)}{\frac{1}{2}k(v)(k(v)-1)}$$

Pairs of neighbours of v that are connected

Possible pairs of v's neighbours, "k(v) choose 2"

$$C(v) = p$$

Alternative: Erdos-Renyi $G(n, m)$ Model



e.g. $m = 4$

1. Start with an empty graph of n nodes.
2. Place m edges uniformly at random among these nodes

Fact: this is equivalent in large graphs to the $G(n, p)$ model via

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

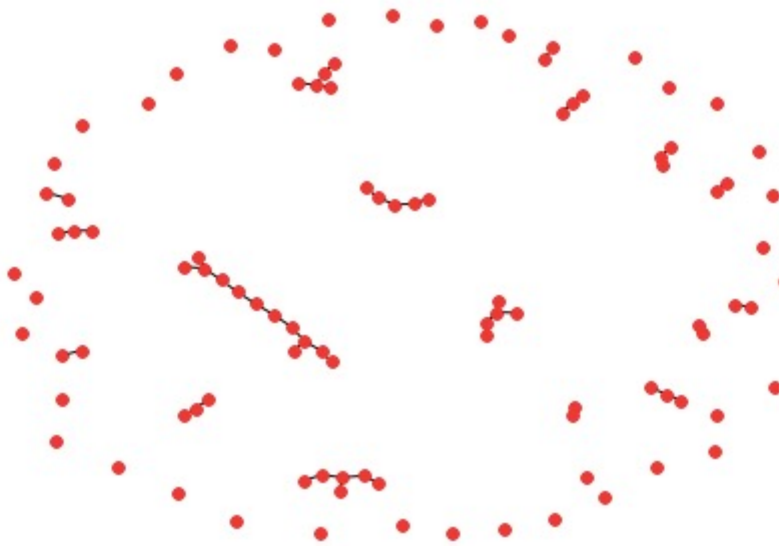
What does this mean?

- Directly controlling the **size** (number of nodes of the graph) by the parameter n
- Directly controlling the **density** by the parameter p (or number of edges m)
- **Where** the edges occur is at uniformly random – every possible graph with n nodes and m edges occurs with **equal probability.**

Jupyter notebook demo
(Lord of the Rings)

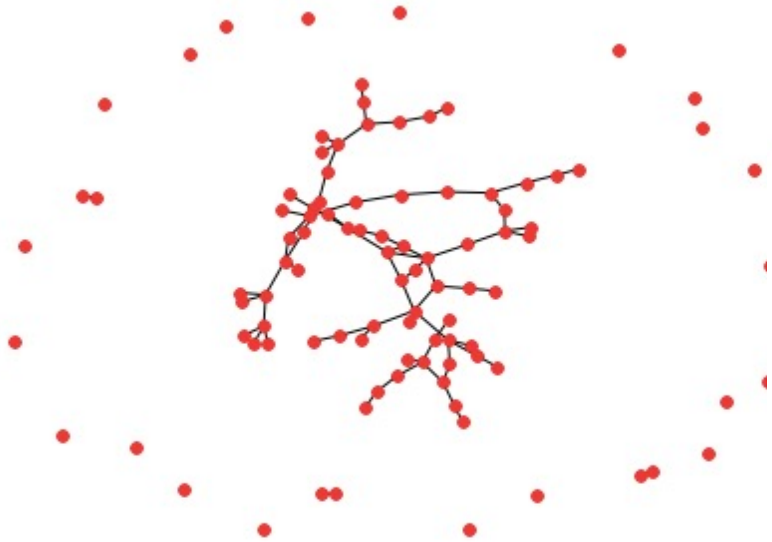
What do ER graphs look like?

$$p < \frac{1}{n}$$



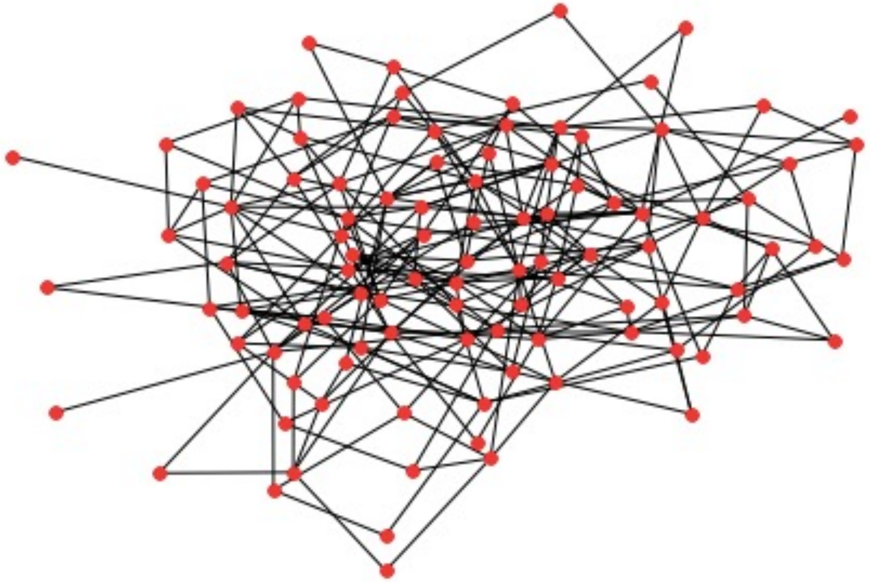
Very disconnected graph, only tiny connected components

$$p = \frac{1}{n} + \epsilon$$



A giant component appears, no/very few cycles

$$p > \frac{\log(n)}{n}$$

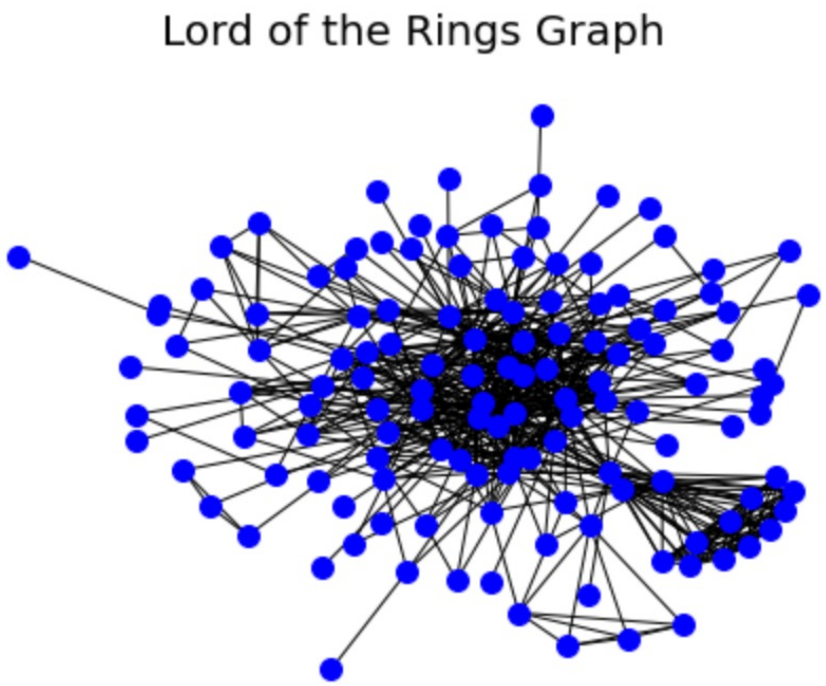


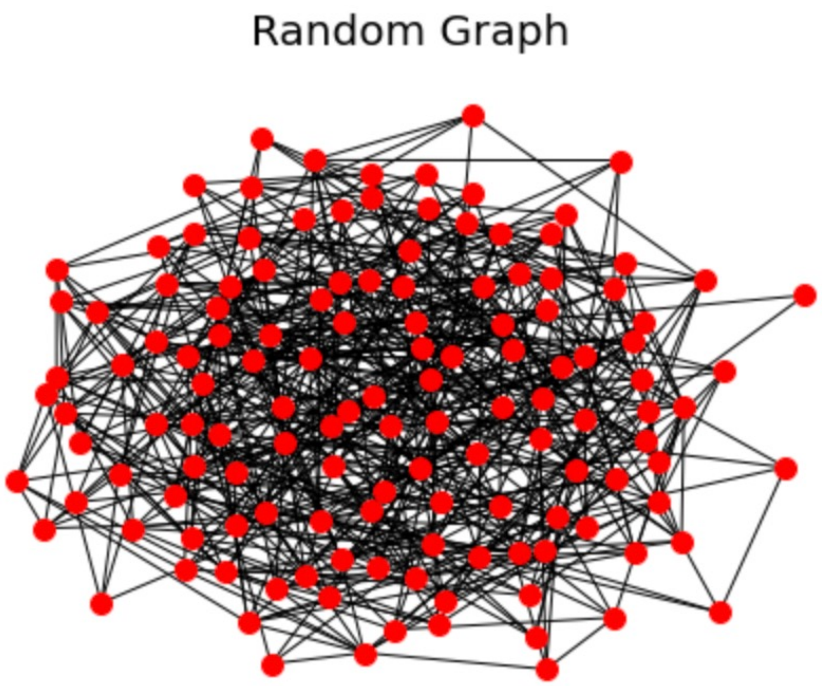
Whole graph is connected, some cycles present

Workflow for random graphs comparison

1. Compute quantities of interest like the **number of nodes and edges** for the real network.
2. Generate a **number** of networks (for taking averages etc) from **random graph models** using the number of nodes and edges as model parameters.
3. Perform analysis on the **real** and **generated** networks and compare.

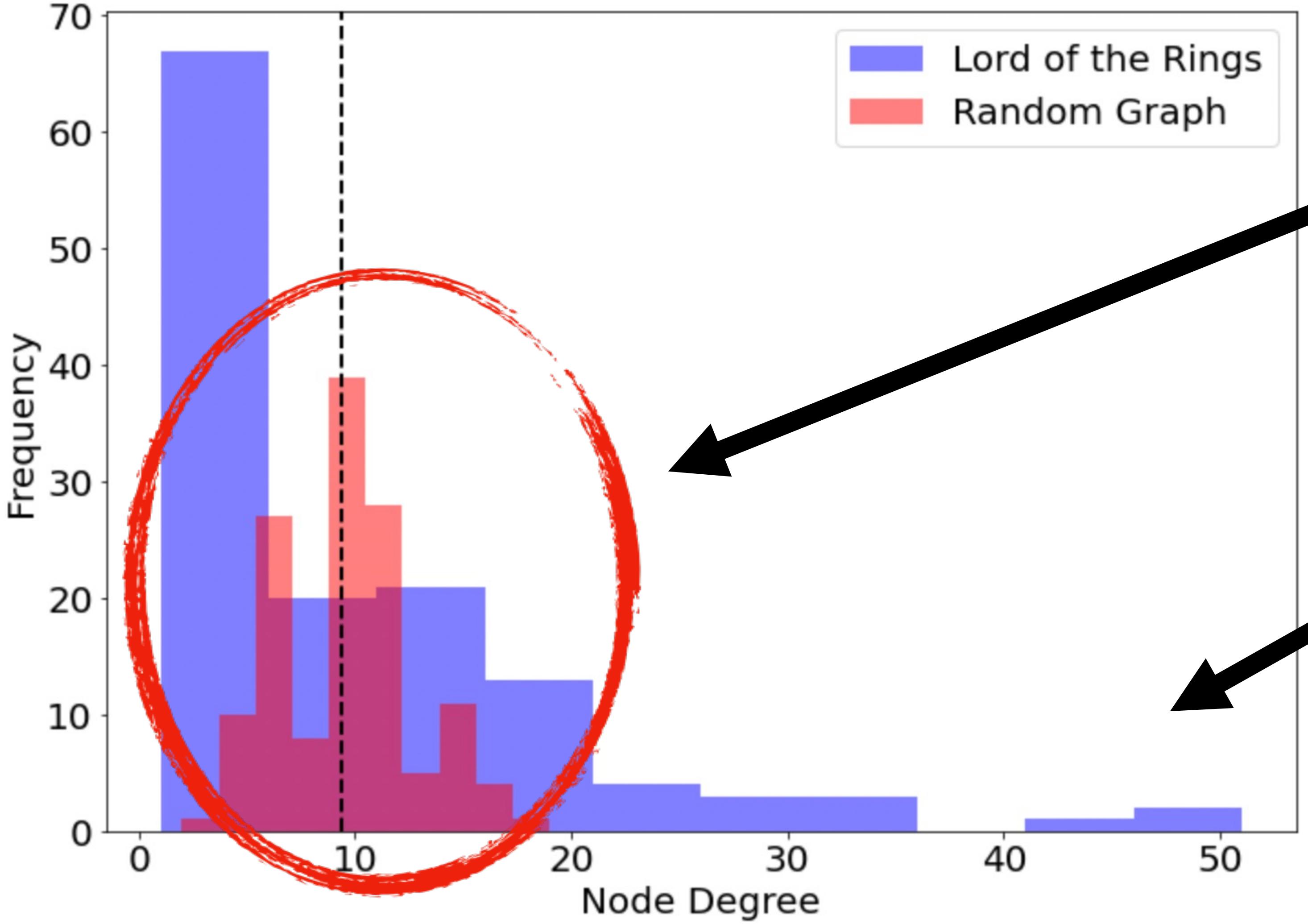
At a glance: Real vs Random networks





Real networks more **heterogeneous** with **community** and **hub/spoke** structure

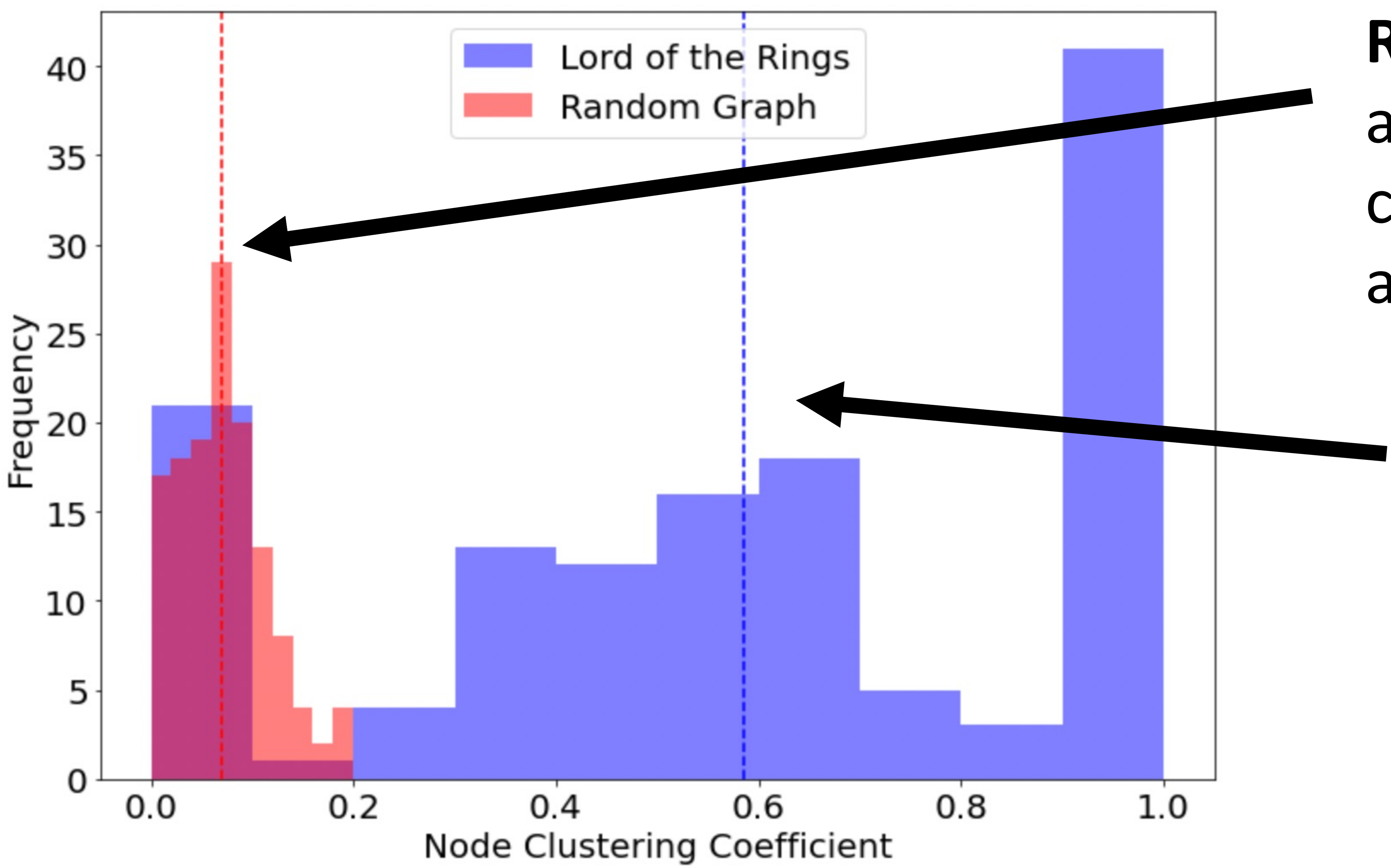
Degree Distribution: Real vs Random



Random: node degrees all clustered round the average value

Real: small number of high degree nodes, large number of low degree nodes

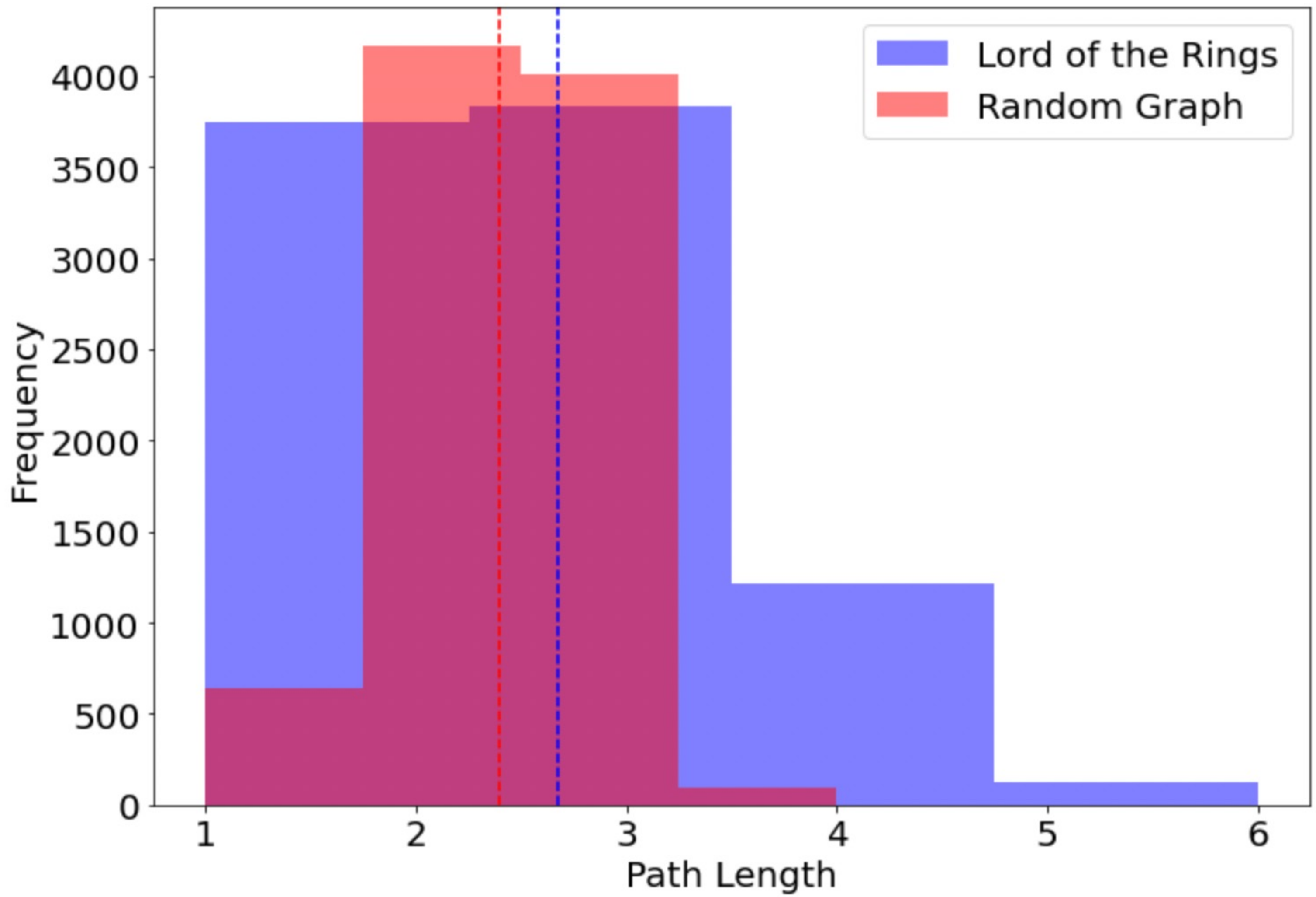
Clustering Coefficient: Real vs Random



Random: very low average clustering coefficient, tightly banded around this number

Real: much higher average clustering coefficient, values much wider distributed

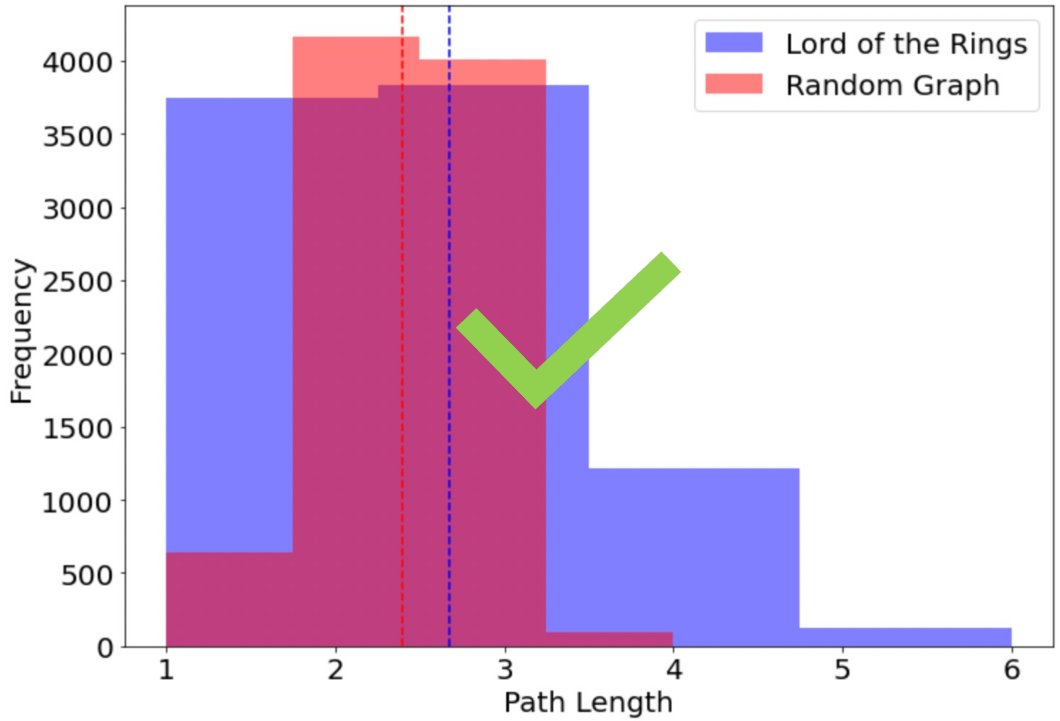
Path lengths: Real vs Random

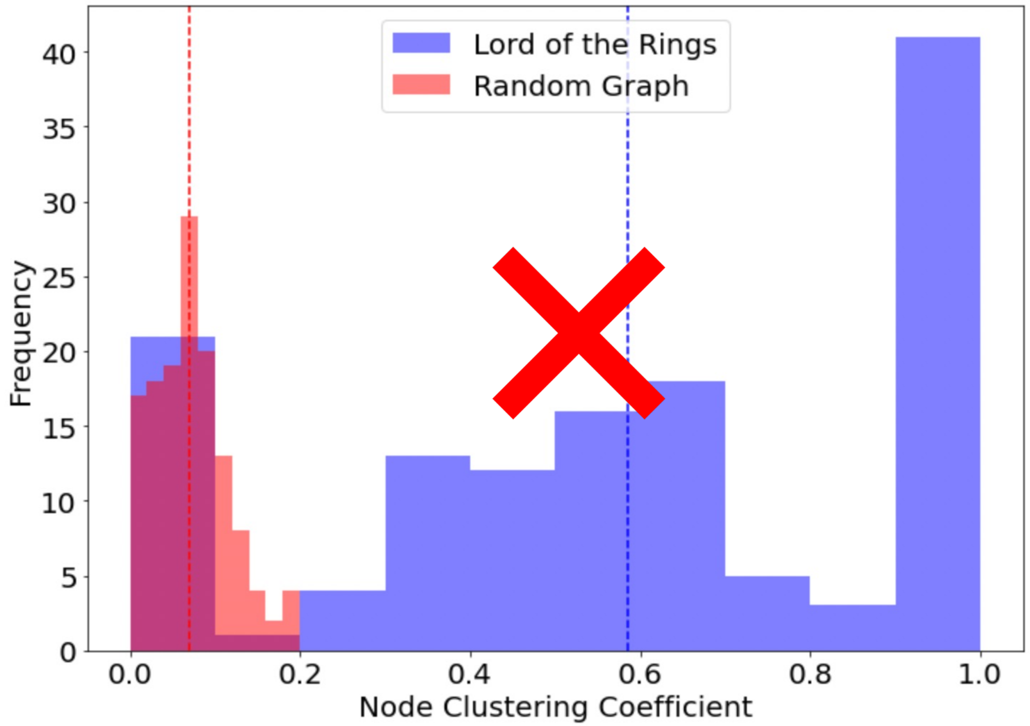


Averages are close but **real** network has **higher variance** in path lengths

How can we make a more realistic model?

Motivation





ER Random Graphs are good at reproducing **average path lengths** but very bad at capturing the **clustering coefficient**

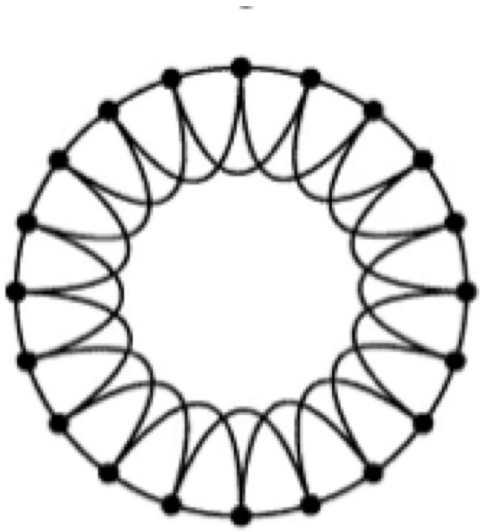
Motivation

Network	Size	$\langle k \rangle$	ℓ	ℓ_{rand}	C	C_{rand}	Reference	Nr.
WWW, site level, undir.	153 127	35.21	3.1	3.35	0.1078	0.00023	Adamic, 1999	1
Internet, domain level	3015–6209	3.52–4.11	3.7–3.76	6.36–6.18	0.18–0.3	0.001	Yook *et al.*, 2001a, Pastor-Satorras *et al.*, 2001	2
Movie actors	225 226	61	3.65	2.99	0.79	0.00027	Watts and Strogatz, 1998	3
LANL co-authorship	52 909	9.7	5.9	4.79	0.43	1.8×10^{-4}	Newman, 2001a, 2001b, 2001c	4
MEDLINE co-authorship	1 520 251	18.1	4.6	4.91	0.066	1.1×10^{-5}	Newman, 2001a, 2001b, 2001c	5
SPIRES co-authorship	56 627	173	4.0	2.12	0.726	0.003	Newman, 2001a, 2001b, 2001c	6
NCSTRL co-authorship	11 994	3.59	9.7	7.34	0.496	3×10^{-4}	Newman, 2001a, 2001b, 2001c	7
Math. co-authorship	70 975	3.9	9.5	8.2	0.59	5.4×10^{-5}	Barabási *et al.*, 2001	8
Neurosci. co-authorship	209 293	11.5	6	5.01	0.76	5.5×10^{-5}	Barabási *et al.*, 2001	9
E. coli, substrate graph	282	7.35	2.9	3.04	0.32	0.026	Wagner and Fell, 2000	10
E. coli, reaction graph	315	28.3	2.62	1.98	0.59	0.09	Wagner and Fell, 2000	11
Ythan estuary food web	134	8.7	2.43	2.26	0.22	0.06	Montoya and Solé, 2000	12
Silwood Park food web	154	4.75	3.40	3.23	0.15	0.03	Montoya and Solé, 2000	13
Words, co-occurrence	460.902	70.13	2.67	3.03	0.437	0.0001	Ferrer i Cancho and Solé, 2001	14
Words, synonyms	22 311	13.48	4.5	3.84	0.7	0.0006	Yook *et al.*, 2001b	15
Power grid	4941	2.67	18.7	12.4	0.08	0.005	Watts and Strogatz, 1998	16
C. Elegans	282	14	2.65	2.25	0.28	0.05	Watts and Strogatz, 1998	17

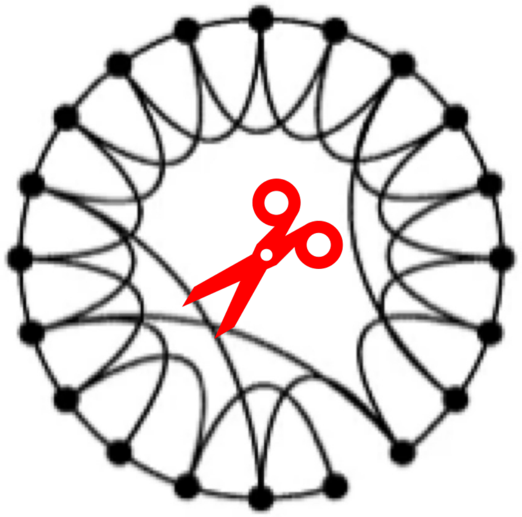
ER Random Graphs are good at reproducing **average path lengths** but very bad at capturing the **clustering coefficient**

Watts and Strogatz: “Can we keep the short path lengths but have higher clustering?”

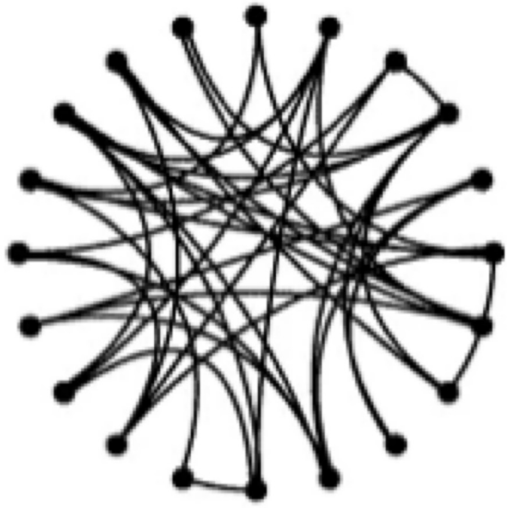
The model



Start with a **ring graph** where each node is connected to the k nodes closest to it. This has a **high clustering coefficient**.

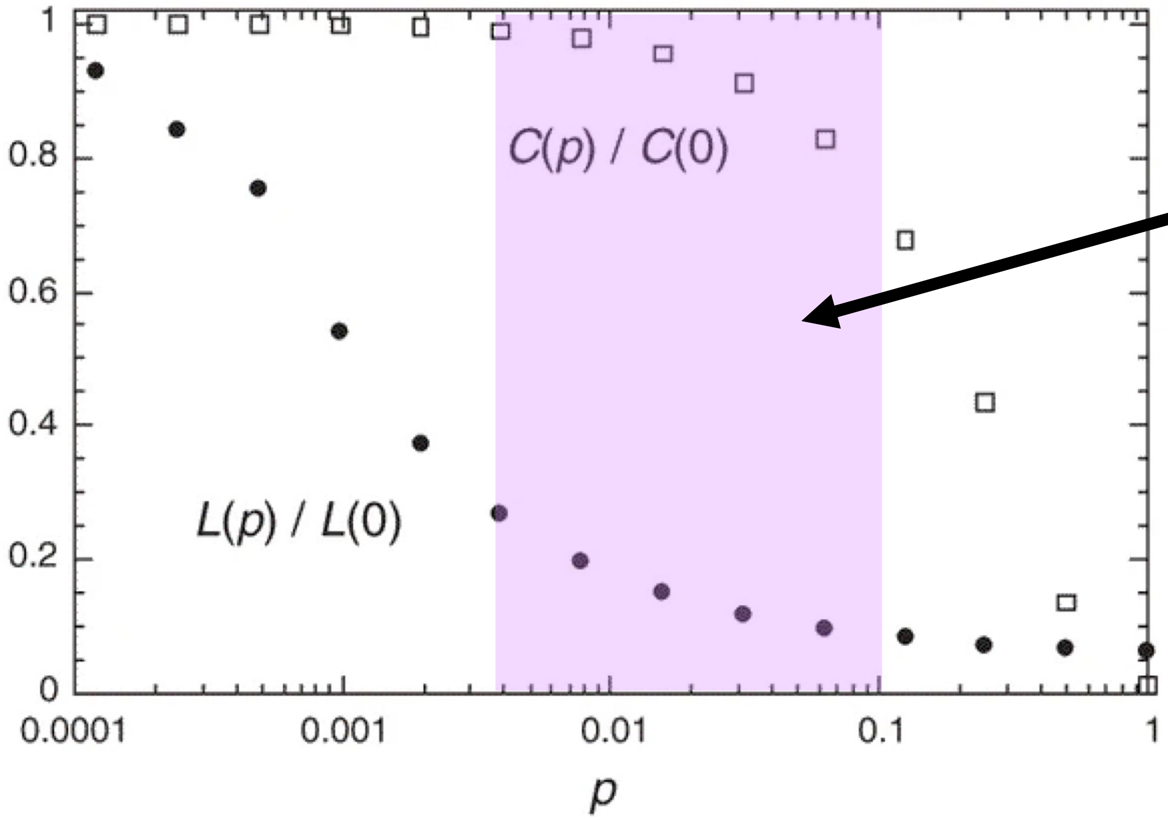


For each node and attached edge, with probability p, **reconnect** it to a randomly chosen node, otherwise leave alone.



When p is **very high**, this looks like a **random graph** again

Tuning between structure and randomness



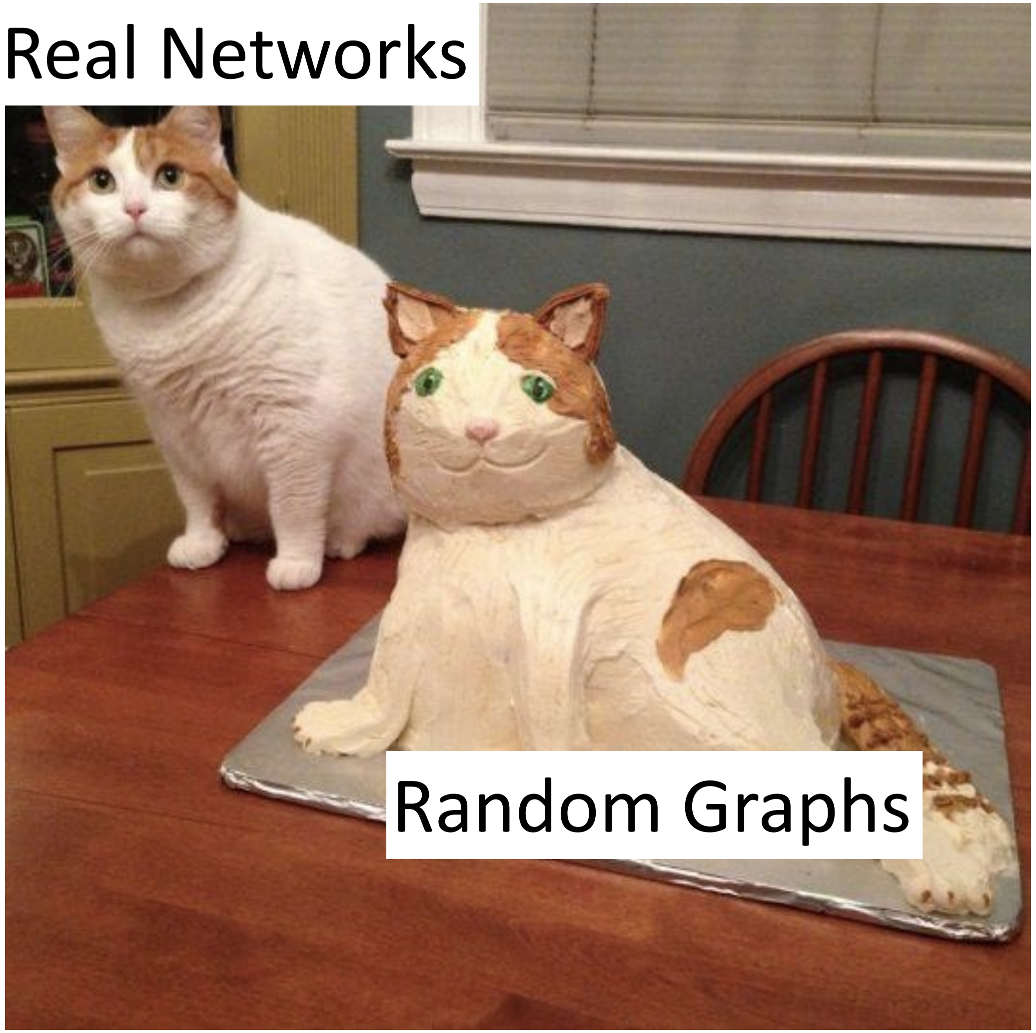
Zone where we have both **high clustering** and **low average path length**

The Goldilocks zone

Lord of the Rings Revisited

Summary

- Real networks have a **heavy-tailed** degree distribution, **high clustering coefficient** and **short path lengths**
- Random graph models provide a **useful comparison** point for experiments, and can be a **good substitute** if no real data available
- **BUT** getting network models to produce networks that have similar property values to real networks is **hard**, and an open problem!



Thank you for listening! What are your questions?