



University of Tehran  
ECE

# Social Network

*Instructor: Masoud Asadpour*

*Teacher Assistant:  
Faezeh Mozaffari  
Farzad Jannati*



Homework 1  
Fall 2025

## Submission Guidelines and Policies

Submit your final file as a single **ZIP** file that includes the report in **PDF** format and all code files, and upload it to the **eLearn platform**. The name of the ZIP file should follow the pattern: SN\_HW#\_StudentNumber

- If you have any questions, contact the assignment TAs via **email**. Please avoid sending private messages on social media so that responses can remain organized and efficient.
- The length of the report is not a grading factor. For implementation questions, focus on providing clear explanations; clarity matters far more than word count.
- The procedure for submitting assignments is explained in detail separately in [this file](#) and in the [Git workshop video](#). Before submitting your code, ensure that you have watched the entire workshop video.
- Every submission must include both the report and the corresponding code. Any code submitted without a report will receive **zero points**.
- In your assignment report, you must describe how you used these tools. Include details such as the tools you used, their specific applications, and any other relevant information.
- At the end of your report, you must include the link to the prompts used: ([The ChatGPT conversation link](#)).
- Assignments may be uploaded to eLearn for up to 7 days after the official deadline, but a **5% penalty per day** will be deducted from the grade for each late day. After 7 days, submissions will not be accepted.
- To verify your understanding of each assignment, there will be a brief 5–10 minute in-person or virtual review session. You will be selected for this session **once during the semester**. If there are discrepancies between your submitted report and your presentation, the chance of being chosen again will increase.
- **Plagiarism is strictly prohibited.** Any similarity in the report or code that indicates copying or if cheating occurs during exams, will result in a score of **0.25 for all students involved for the course**.

# Contents

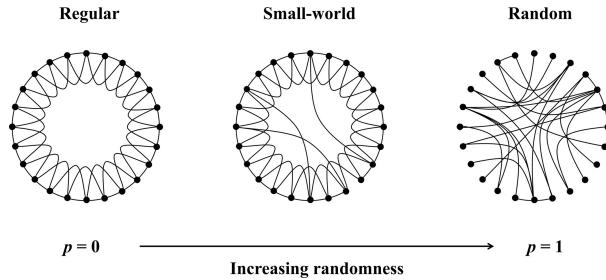
|  |   |
|--|---|
| Theoretical Questions . . . . .  | 3 |
| Question 1: The Watts-Strogatz Model . . . . .                             | 3 |
| Question 2: Snobbish Network . . . . .                                     | 3 |
| Implementation Questions . . . . .   | 4 |
| Question 1: Small World Phenomena in Random Networks . . . . .             | 4 |
| Question 2: Generative Graph Models & Genetic Encodings . . . . .          | 5 |
| Question 3: The Evolution of Random Networks & Phase Transitions . . . . . | 7 |

## Theoretical Questions

### Question 1: The Watts-Strogatz Model

- (a) Consider the small-world model of Watts and Strogatz with rewiring probability  $p$ . The model starts with a regular ring lattice where each node is connected to its  $\langle k \rangle / 2$  neighbors on either side for a total degree of  $\langle k \rangle$ .<sup>1</sup> Show that when  $p = 0$ , the overall clustering coefficient of this graph is given by:

$$C(0) = \frac{3(K - 2)}{4(K - 1)}$$



**Figure 1:** The Watts-Strogatz Model

- (b) Show that when  $p > 0$ , the overall clustering coefficient is given by:

$$C(p) \approx \frac{3(K - 2)}{4(K - 1)}(1 - p)^3$$

### Question 2: Snobbish Network

Consider a network of  $2N$  total nodes, which is divided into two equal-sized communities,  $N$  red nodes and  $N$  blue nodes. The probability that there is a link between nodes of identical color is  $p$  and the probability that there is a link between nodes of different color is  $q$ . A network is snobbish if  $p > q$ , capturing a tendency to connect to nodes of the same color. For  $q = 0$  the network has at least two components, containing nodes with the same color.

- (a) For a very large, snobbish network where  $p \gg q$ , determine the minimal values for  $p$  and  $q$  (each in terms of  $N$ ) required to ensure global connectivity.
- (b) Consider a large network that satisfies the two connectivity criteria you derived in the previous part.
1. What is the expected scaling of the average shortest path length  $\langle d_{\text{same}} \rangle$  between two nodes of the same color?
  2. What is the expected scaling of the average shortest path length  $\langle d_{\text{diff}} \rangle$  between two nodes of different colors?
  3. Based on your answers, does this network model exhibit the small-world property?<sup>1</sup> Justify your answer by explaining how this model does (or does not) satisfy the two key characteristics of small-world networks (path length and clustering).

## Implementation Questions

### Question 1: Small World Phenomena in Random Networks

The “Small World” property is characterized by a low average shortest path length between nodes, typically scaling logarithmically with the network size ( $\langle d \rangle \sim \ln N$ ). In contrast, regular lattices exhibit polynomial scaling ( $\langle d \rangle \sim N^{1/D}$ ). In this task, you will reproduce the scaling behavior of path lengths in simulations to verify these theoretical bounds.

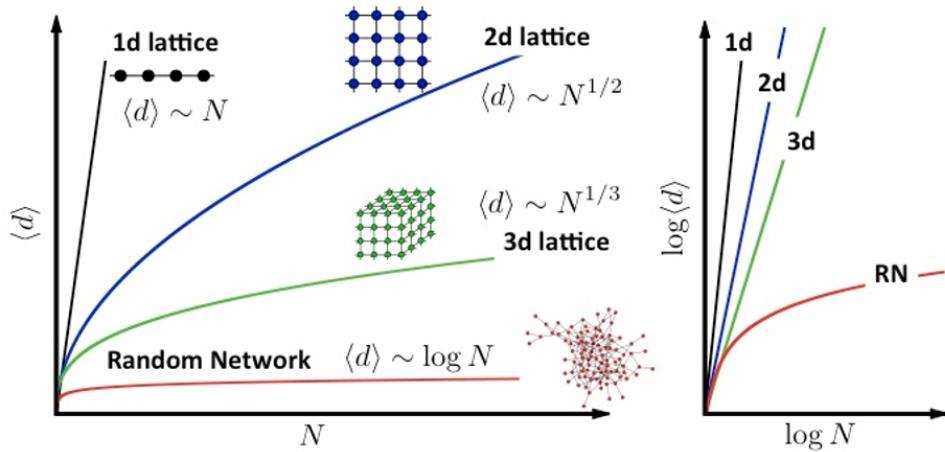
#### (a) Network Construction and Simulation

You are required to generate four types of undirected networks with varying system size  $N$ :

- **1D Lattice:** A ring graph where nodes are connected to  $k$  nearest neighbors (use  $k = 2$ ).
- **2D Lattice:** A grid graph (square lattice) with periodic boundary conditions (torus) or fixed borders.
- **3D Lattice:** A cubic grid graph.
- **Random Network (RN):** An Erdős–Rényi random graph ( $G_{N,p}$ ) with a fixed average degree  $\langle k \rangle \approx 4$  (set  $p = \langle k \rangle / (N - 1)$ ).

**Implementation Task:** Perform a parameter sweep over the number of nodes  $N$  (e.g., logarithmically spaced values from  $N = 500$  to  $N = 5000$ ). For each  $N$  and each topology, generate the graph and calculate the **Average Shortest Path Length**  $\langle d \rangle$ .

*Note: Calculating all-pairs shortest paths is computationally expensive. To keep runtime reasonable, limit  $N \leq 5000$  and use fewer repetitions for larger graphs.*



**Figure 2:** Theoretical scaling of average path length  $\langle d \rangle$  vs network size  $N$  for Lattices and Random Networks.

#### (b) Scaling Analysis

Visualize your simulation results by generating two plots similar to Figure 2:

- **Linear Plot:**  $\langle d \rangle$  vs  $N$  (linear scales).
- **Log-Log Plot:**  $\log(\langle d \rangle)$  vs  $\log(N)$ .

Perform a linear regression (fit) on the Log-Log data to extract the scaling exponent for each topology.

1. Compare your simulation results with the theoretical predictions shown in the lecture slide:

- 1D Lattice:  $\langle d \rangle \sim N$
- 2D Lattice:  $\langle d \rangle \sim N^{1/2}$
- 3D Lattice:  $\langle d \rangle \sim N^{1/3}$
- Random Network:  $\langle d \rangle \sim \ln N$  (or  $\log N$ )

Do your simulated exponents match the theoretical dimensionality ( $1/D$ )?

2. Explain why the Random Network follows a logarithmic scaling while lattices follow a polynomial scaling. What structural feature of the Random Network (often referred to as “shortcuts”) is responsible for this drastic reduction in path length?

## Question 2: Generative Graph Models & Genetic Encodings

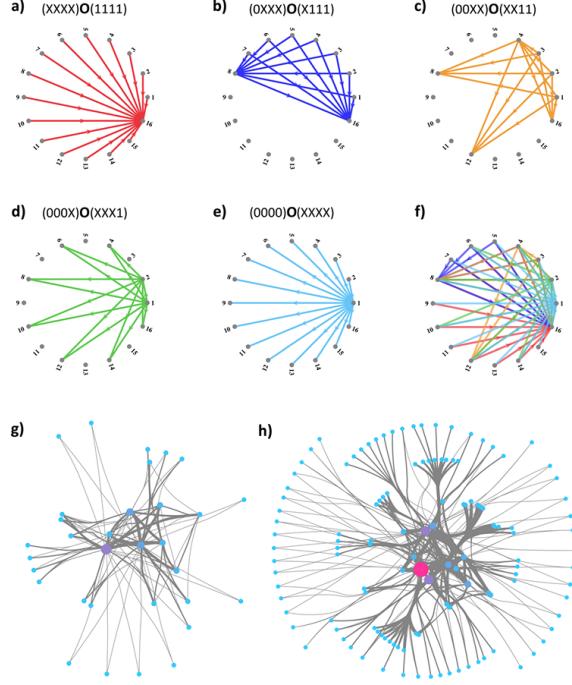
In this assignment, you will implement a **Genetic Generative Model** based on the framework proposed by [Barabási & Czégel \(2021\)](#). In this framework, nodes are assigned unique binary addresses (DNA/Barcodes), and connections are formed based on specific “wiring rules” that match these patterns. You will implement deterministic heuristics to recreate complex topologies and a stochastic process (the **Random Genetic** or **RG** model) to study phase transitions, density saturation, and structural robustness.

### (a) Deterministic Construction of Scale-Free Networks

In this model, we define a network of size  $N = 2^b$ , where each node is identified by a unique binary string of length  $b$ . Edges are created by specific **Rules** connecting a *Source Pattern* ( $S$ ) to a *Destination Pattern* ( $D$ ). You are required to implement a deterministic heuristic to generate a **Directed** Scale-Free network.

Apply the following algorithm iterating  $i$  from 0 to  $b$ :

- **Source Set ( $S_i$ ):** A pattern starting with  $i$  zeros ('0'), followed by  $(b - i)$  wildcards ('X'). (e.g., for  $b = 4, i = 1 \rightarrow 0XXX$ ).
- **Destination Set ( $D_i$ ):** A pattern starting with  $i$  wildcards ('X'), followed by  $(b - i)$  ones ('1'). (e.g., for  $b = 4, i = 1 \rightarrow X111$ ).
- **Action:** Create **Directed** edges from *every* node matching  $S_i$  to *every* node matching  $D_i$ . Note that self-loops are permitted and expected.



**Figure 3:** Scale-free network construction using wiring rules

Implement a function and generate a network with  $b = 10$  ( $N = 1024$  nodes). Visualize the Adjacency Matrix using a Heatmap (Spy Plot). Finally, plot the In-Degree and Out-Degree Distributions ( $P(k)$ ) on a Log-Log scale and perform a linear fit to estimate the power-law exponent  $\gamma$ .

1. Examine the Adjacency Matrix plot. Explain how the recursive, block-like structure visually represents the hierarchical nature of the hubs.
2. In standard Barabási–Albert models,  $\gamma \approx 3$ . However, this deterministic method produces a different exponent. Does your estimated  $\gamma$  converge closer to  $\gamma \approx 1$  (slope of -1)? Explain why this specific bit-matching heuristic creates such an extreme "winner-takes-all" topology.

### (b) The Random Genetic (RG) Model & Parameter Space

Unlike the deterministic model, the Random Genetic (RG) model selects rules stochastically. The model relies on parameters  $b$  (barcode length),  $x$  (number of 'X' wildcards per rule), and  $r$  (number of rules). For each of the  $r$  rules, randomly assign  $x$  positions as 'X' and the remaining  $(b - x)$  positions as '0' or '1' for both Source and Destination patterns. Connect all matching Source nodes to Destination nodes.

Implement the corresponding function. To analyze the parameter space, perform a parameter sweep varying  $x$  from 1 to 8 and  $r$  from 2 to 160 (using logarithmic spacing). Calculate the Density for each  $(x, r)$  pair and visualize the results using a 3D Surface Plot or Heatmap where  $Z$  represents Network Density.

1. The theoretical density is given by  $\rho = 1 - (1 - \pi^2)^{2r}$  where  $\pi = 2^{x-b}$ . Compare your simulation results with this formula. Does the density grow linearly with  $r$ , or does it saturate?
2. Explain the concept of "Rule Overlap." Why does adding more rules eventually yield diminishing returns in terms of new edges created?

### Question 3: The Evolution of Random Networks & Phase Transitions

As we add links to a set of isolated nodes, the network undergoes a dramatic topological change. It transforms from a fragmented collection of small clusters into a single cohesive system. This phenomenon, known as a **Phase Transition**, is characterized by the sudden emergence of a **Giant Component (GC)**. In this problem, you will simulate the evolution of an Erdős-Rényi network to identify the critical threshold of this transition.

#### (a) Simulating Network Evolution

You are required to simulate the growth of a random network by varying the average degree  $\langle k \rangle$  (which acts as the control parameter).

**Implementation Task:** Fix the network size at  $N = 1000$ . Perform a parameter sweep for the average degree  $\langle k \rangle$  ranging from 0 to 5.

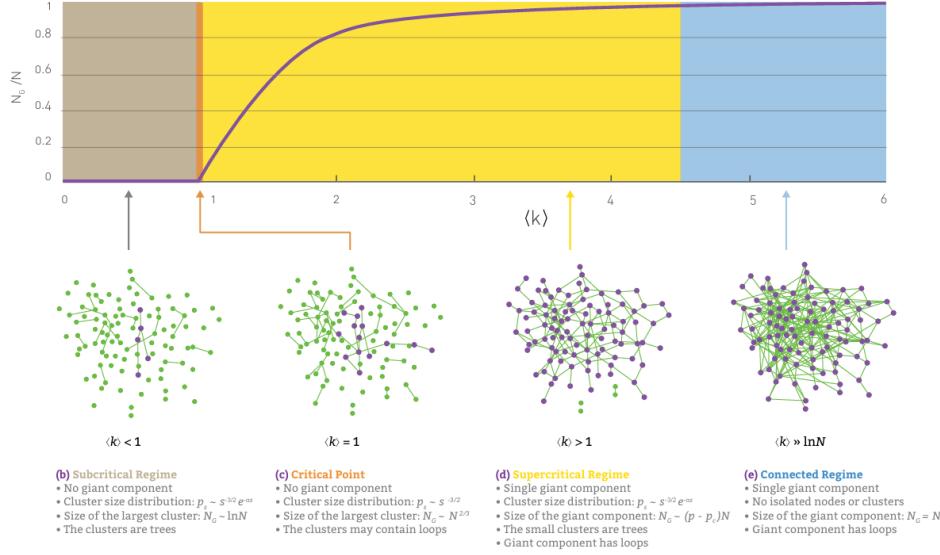
**Crucial:** To accurately capture the divergence at the critical point, use a variable step size:

- Use a step of 0.1 for non-critical regions.
- **Use a finer step (e.g., 0.02)** in the critical window  $\langle k \rangle \in [0.8, 1.2]$ .

For each  $\langle k \rangle$ :

- i. Calculate the corresponding probability  $p = \langle k \rangle / (N - 1)$ .
- ii. Generate an Erdős-Rényi graph  $G(N, p)$ .
- iii. Identify all connected components.
- iv. Record the size of the largest component ( $N_G$ ) and calculate the order parameter  $S = N_G/N$  (relative size of the giant component).
- v. Calculate the average size of the *remaining* small clusters,  $\langle s \rangle$ . This implies calculating the average size of all components *strictly excluding* the largest one.

*Note: To obtain smooth curves, average your results over at least 50 independent realizations for each  $\langle k \rangle$ .*



**Figure 4:** The evolution of the giant component size ( $N_G/N$ ) as a function of the average degree  $\langle k \rangle$ .

### (b) Analyzing the Critical Threshold

Generate two plots sharing the x-axis ( $\langle k \rangle$ ):

- **Plot 1:** The order parameter  $S = N_G/N$  vs.  $\langle k \rangle$ .
- **Plot 2:** The average size of isolated clusters  $\langle s \rangle$  vs.  $\langle k \rangle$ .

1. Identify the critical point  $\langle k \rangle_c$  from your simulations. Does the transition occur at the theoretical prediction  $\langle k \rangle = 1$ ? Explain any deviations (consider finite-size effects).
2. Examine the plot of  $\langle s \rangle$ . Why does the average cluster size diverge (peak) near the critical point and then decrease in the supercritical regime ( $\langle k \rangle > 1$ )? Explain the physical meaning of this peak.

### (c) Finite Size Effects

In thermodynamic theory, phase transitions are sharp only in the limit of infinite system size ( $N \rightarrow \infty$ ). In finite networks, the transition is smoother.

**Implementation Task:** Repeat the simulation from Part (a) for three different network sizes:  $N = 100$ ,  $N = 1,000$ , and  $N = 10,000$ . Plot  $S$  vs.  $\langle k \rangle$  for all three sizes on the same figure.

1. Observe the slope of the curve near  $\langle k \rangle = 1$ . How does the "sharpness" of the transition change as  $N$  increases?
2. At  $\langle k \rangle = 1$ , does the relative size of the giant component  $S$  tend towards zero or a finite value as  $N$  increases? Relate this to the theoretical prediction  $N_G \sim N^{2/3}$  at criticality.

### (d) The Critical State

The network structure at the critical point is unique. It is populated by trees of various sizes, following a power-law distribution.

**Implementation Task:** Generate a single large Random Network with  $N = 10,000$  exactly at the critical point  $\langle k \rangle = 1$ . Calculate the distribution of component sizes  $P(s)$ .

**requirement:** Plot  $P(s)$  vs.  $s$  on a Log-Log scale. You must use **Logarithmic Binning** (log-spaced bins) to visualize the power-law tail cleanly; linear binning will result in excessive noise.

1. Does the component size distribution follow a power law  $P(s) \sim s^{-\alpha}$ ? If so, estimate the exponent  $\alpha$ .
2. Compare your estimated exponent with the theoretical prediction for random graphs at criticality ( $\alpha = 3/2$ ).