Lab 03. Watts-Strogatz Networks and Markov-Chain Convergence

Jordi-Lluís Figueras

Modelling Complex Systems, Spring 2026

1 Introduction

In this lab we study two topics from the lectures:

- the structure of Watts-Strogatz small-world graphs, measured by clustering and distances,
- the convergence of a Markov chain, measured by the eigenvalues of its transition matrix.

The two parts are intentionally separate. First you will generate Watts-Strogatz graphs and compute standard network statistics. Then you will study the convergence speed of a Markov chain from an explicitly given transition matrix. The Markov-chain part does not use the Watts-Strogatz graphs.

For the Markov-chain part, use the column-stochastic convention from the lectures: if x_t is the probability distribution at time t, then

$$x_{t+1} = Px_t.$$

2 Exercises

Work in a clear and reproducible way. Your report should explain both the definitions and the numerical experiments. Unless explicitly stated otherwise, all exercises below are part of the lab.

Recommended Python libraries: use NetworkX for the Watts-Strogatz graph generation and graph statistics in Exercise 1. Use NumPy for the matrix and eigenvalue computations in Exercise 2, and Matplotlib for the plots.

1. Watts-Strogatz graphs and network statistics

The Watts-Strogatz model constructs an undirected graph $G(N, k, p)$ as follows. Start with N nodes arranged on a ring. Assume that k is even. Each node is connected to its $k/2$ nearest neighbors on each side, so every node initially has degree k. Then each edge is rewired with probability p, avoiding self-loops and duplicate edges.

The parameter p controls the amount of randomness:

- $p = 0$ gives a regular ring lattice,
- small $p > 0$ gives a graph with a few long-range shortcuts,
- $p = 1$ gives a highly randomized graph with the same number of edges.

Use, for example,

$$N = 200, \qquad k = 6,$$

and compare several values of p, such as

$$p = 0,\ 0.001,\ 0.003,\ 0.01,\ 0.03,\ 0.1,\ 0.3,\ 1.$$

Because the model is random when $p > 0$, repeat the experiment for many independent realizations for each value of p. Aim for 1000 realizations per value of p when the computation is feasible. If this is too slow on your machine, use a smaller number of realizations, but state the number clearly and explain the computational limitation. This matters especially for very small values such as $p = 0.001$, where many samples may be needed before the effect of rare rewired edges becomes visible.

For each graph, compute the following quantities. In your report, define each quantity before using it.

Degrees and degree distribution

The degree d_i of node i is the number of edges incident to i. The average degree is

$$\bar{d} = \frac{1}{N} \sum_{i=1}^{N} d_i.$$

The degree distribution is the empirical distribution of the numbers d_i. In practice, compute all degrees and make a histogram of their values.

For the standard Watts-Strogatz construction, the number of edges is preserved, so the average degree is exactly k. Explain why the degrees are exactly equal to k when $p = 0$, and why they may vary from node to node when $p > 0$.

Connected components

A path from node i to node j is a sequence of edges connecting i to j. An undirected graph is connected if every pair of nodes can be joined by a path. A connected component is a maximal connected subgraph.

For each generated graph:

- determine whether it is connected,
- record the number of connected components,
- record the size of the largest connected component.

If a generated graph is disconnected, report this in the component statistics and compute distance quantities on its largest connected component. This keeps the distance convention consistent across all values of p.

Shortest-path distance, diameter, and average path length

Assume first that G is connected. The shortest-path distance between two nodes i and j is denoted by $\mathrm{dist}(i, j)$ and is defined as the minimum number of edges in a path from i to j.

The diameter is the largest shortest-path distance:

$$\operatorname{diam}(G) = \max_{i,j} \operatorname{dist}(i,j).$$

The average path length is

$$\ell(G) = \frac{1}{N(N-1)} \sum_{i \neq j} \operatorname{dist}(i,j).$$

These quantities measure the global scale of the graph. A small average path length means that most nodes can be reached from most other nodes in only a few steps.

Compute $\operatorname{diam}(G)$ and $\ell(G)$ for each connected graph. If the graph is disconnected, compute these quantities on the largest connected component and replace N in the formula above by the number of nodes in that component.

Clustering coefficient

The local clustering coefficient of node i measures how many pairs of neighbors of i are themselves connected. Let d_i be the degree of node i. Among the d_i neighbors of i, there are at most

$$\frac{d_i(d_i-1)}{2}$$

possible edges. If e_i of these edges are present, define

$$C_i = \frac{2e_i}{d_i(d_i-1)}.$$

If $d_i < 2$, set $C_i = 0$.

The average clustering coefficient is

$$C(G) = \frac{1}{N} \sum_{i=1}^{N} C_i.$$

Compute $C(G)$ for each graph. Then plot the averages of $C(G)$ and $\ell(G)$ as functions of p. It is useful to normalize by the value at $p = 0$, plotting

$$\frac{C(p)}{C(0)}, \qquad \frac{\ell(p)}{\ell(0)}.$$

Discuss whether you observe the small-world effect: a range of small positive p where clustering remains relatively high but the average path length drops substantially.

2. Convergence speed from the second-largest eigenvalue

This exercise is the main Markov-chain part of the lab. It is written step-by-step because the goal is to connect computation with the spectral mechanism behind convergence. It is independent of the Watts-Strogatz graphs in Exercise 1.

Use the column-stochastic convention from the lectures: if x_t is the probability distribution at time t, then

$$x_{t+1} = P x_t.$$

As a concrete example, start with the transition matrix

$$P_0 = \begin{pmatrix} 0.60 & 0.20 & 0.15 & 0.10 \\ 0.20 & 0.50 & 0.20 & 0.20 \\ 0.10 & 0.20 & 0.45 & 0.20 \\ 0.10 & 0.10 & 0.20 & 0.50 \end{pmatrix}.$$

Then repeat the experiment for the lazy variants

$$P_\alpha = \alpha I + (1-\alpha)P_0, \qquad \alpha = 0.5,\ 0.8.$$

Increasing α means that the chain has a larger probability of remaining in its current state. The lazy variants have the same stationary distribution as P_0. If λ is an eigenvalue of P_0, then $\alpha + (1-\alpha)\lambda$ is the corresponding eigenvalue of P_α, so larger values of α move the nontrivial eigenvalues closer to 1 and slow down convergence.

Let $P \in \mathbb{R}^{N\times N}$ be a column-stochastic transition matrix. Assume that 1 is a simple eigenvalue of P, with normalized stationary eigenvector π, and assume that all other eigenvalues satisfy

$$|\lambda| < 1.$$

Together, these assumptions ensure convergence to a unique stationary distribution. They exclude periodic obstructions and reducibility-related obstructions. For a general Markov chain, if you encounter an eigenvalue very close to -1, explain what this means for convergence. You may then replace the chain by the lazy chain

$$P_{\text{lazy}} = \frac{1}{2}(I + P)$$

and repeat the computation.

Let the eigenvalues be ordered by decreasing modulus:

$$\lambda_1 = 1, \qquad |\lambda_2| \geq |\lambda_3| \geq \cdots \geq |\lambda_N|.$$

The number

$$\mu = |\lambda_2|$$

is called the second-largest eigenvalue modulus. The spectral gap is

$$g = 1 - \mu.$$

2.1 Stationary distribution

First verify that P_0 is a valid transition matrix:

- all entries are nonnegative,
- each column sums to 1.

Compute the eigenvector corresponding to the eigenvalue 1 and normalize it so that its entries sum to 1. This vector is the stationary distribution π. Verify numerically that

$$P\pi \approx \pi.$$

Report the stationary distribution. Explain why a stationary distribution is an equilibrium for the Markov-chain dynamics, even though an individual realization of the chain may keep changing state.

2.2 Error dynamics

Let x_0 be an initial probability distribution and define

$$x_t = P^t x_0.$$

Set

$$e_t = x_t - \pi.$$

Show that

$$e_{t+1} = P e_t,$$

and therefore

$$e_t = P^t e_0.$$

Explain why this means that convergence to stationarity is controlled by the powers of P acting on the initial error.

2.3 Spectral decomposition

For this part, assume that P is diagonalizable. Write the initial error as a linear combination of eigenvectors:

$$e_0 = c_2 v_2 + c_3 v_3 + \cdots + c_N v_N.$$

There is no v_1 term because $v_1 = \pi$ is the stationary direction and $e_0 = x_0 - \pi$ has total mass zero.

Use this expansion to show that

$$e_t = c_2 \lambda_2^t v_2 + c_3 \lambda_3^t v_3 + \cdots + c_N \lambda_N^t v_N.$$

Conclude that, for a chosen vector norm and for some constant K depending on x_0 and on the eigenvectors,

$$\|x_t - \pi\| \leq K \mu^t.$$

In the numerical experiment below, you will use total variation distance, which is

$$\frac{1}{2}\|x_t - \pi\|_1.$$

This inequality is the main point: the closer μ is to 1, the slower the Markov chain converges.

2.4 Estimated time to reach a tolerance

Suppose that you want

$$\|x_t - \pi\| \leq \varepsilon.$$

Using the estimate

$$\|x_t - \pi\| \lesssim K \mu^t,$$

derive the approximate condition

$$t \gtrsim \frac{\log(K/\varepsilon)}{-\log(\mu)}.$$

When μ is close to 1, use the approximation

$$-\log(\mu) \approx 1 - \mu = g$$

to explain why the convergence time is approximately proportional to

$$\frac{1}{g}.$$

In words: a larger spectral gap means faster convergence.

2.5 Numerical verification

Write a Python program, **spectralConvergence.py**, that performs the following experiment.

For P_0 and for the matrices P_α:

- verify that P is column-stochastic,
- compute the stationary distribution π,
- compute the eigenvalues of P,
- compute $\mu = |\lambda_2|$ and $g = 1 - \mu$,
- choose one or more initial states and set x_0 to be concentrated at that state,
- iterate $x_{t+1} = Px_t$,
- compute the total variation distance

$$\delta(t) = \frac{1}{2} \sum_{i=1}^{N} |x_t(i) - \pi_i|,$$

- plot $\delta(t)$ on a logarithmic vertical scale.

On the same plot, compare the observed decay with a curve proportional to μ^t. You do not need to match the multiplicative constant exactly; the goal is to compare the slope on the logarithmic scale.

For each matrix, report a table containing:

- the matrix label, for example P_0, $P_{0.5}$, or $P_{0.8}$,
- the second-largest eigenvalue modulus μ,
- the spectral gap $g = 1 - \mu$,
- the observed number of steps needed for $\delta(t)$ to fall below 10^{-3}, if it does within your simulation time.

Discuss how the observed convergence changes when μ is closer to 1 or farther from 1.

3. Synthesis

Write a short final discussion answering the following questions.

- For which values of p do you see high clustering together with short average path length?
- How variable are the Watts-Strogatz statistics across different random realizations with the same p?
- In the Markov-chain experiment, how does the second-largest eigenvalue modulus affect the observed convergence speed?
- Why is the spectral gap $g = 1 - \mu$ a more direct predictor of convergence speed than visual inspection of the entries of P?

- What are the limitations of using only one graph realization for each value of p, or only one initial condition for the Markov chain?

Your answer should distinguish clearly between empirical observations from your plots and mathematical statements justified by the spectral argument. Do not claim that the Watts-Strogatz statistics determine the Markov-chain convergence in this lab, since the two exercises are separate.

3 What to submit

Submit:

- a short report in PDF format,
- the Python files used for the computations and figures,
- the generated figures, if they are not already embedded in the report,
- in case LLM tools were used, an appendix reporting the most important prompts used for the project.

Your report should include:

- definitions of all graph statistics used in the lab,
- plots of clustering and average path length as functions of p,
- the verification that the Markov-chain matrices are column-stochastic,
- the derivation of the eigenvalue-based convergence estimate,
- numerical plots of convergence to stationarity,
- a table comparing matrix labels, μ, spectral gap, and observed convergence time,
- a final interpretation of the Watts-Strogatz statistics and the Markov-chain convergence experiment.

You may use external sources, but they must be cited correctly. Use Python for all implementations.