Homework 9:
Markov Chain Monte Carlo
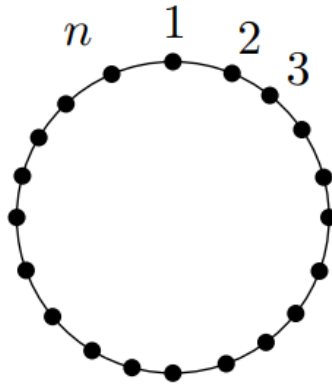
Problem 1: Markov chains

Figure 1: A cycle graph with n vertices.

For each of the following three graphs, consider a Markov chain defined by a uniformly random walk over the nodes of the graph:

1. The cycle graph with $n = 17$.

2. The cycle graph with $n = 18$.

3. The cycle graph with $n = 18$, and additional self-loops at nodes $2, 4, 6, \ldots, 18$.

(a) [2 points] For each Markov chain, explain whether it is aperiodic or irreducible. If it is aperiodic and irreducible, what is its stationary distribution?

(b) [4 points] Given a Markov chain and an initial state s, let $\mu_v(t)$ denote the distribution of the state after t transitions of the chain starting the random walk from v.

Take $v = 1$ and for each of the three chains, plot $\|\mu_v(t) - \pi\|_{TV}$ for $t \in \{0, 1, 2, \ldots, 100\}$, where the *total variation distance* between two distributions p_1 and p_2 on a domain Ω is defined as

$$\|p_1 - p_2\|_{TV} = \frac{1}{2} \sum_{v \in \Omega} |p_1(v) - p_2(v)|.$$

Graph the three lines on the same plot, clearly labeling them. For the Markov chains that are not both aperiodic and irreducible, set π to be equal to the uniform distribution in your total variation calculations.

Include both a plot and your code for this part.

(c) [2 points] Report the values of the second-largest and the smallest eigenvalue of the transition matrix for each of these three Markov chains.

(d) [4 points] Discuss the results of parts (b) and (c) and their connection. You should specifically mention the mixing time and the power iteration algorithm.

(e) [6 points] Using these insights, can you construct a graph on 18 vertices, where each node has degree 3, where the resulting random walk mixes significantly faster than on the cycle graph? Report how you constructed this graph, and the plot the total variation distance for the random walks on this graph you found as you did for part (b).

Problem 2: Visiting 30 national parks

Your goal is to design a short *tour* of 30 national parks. This is a route starting from one park, visiting all the parks in some order, and the returning to the starting point. In the file `parks.csv` linked on the course webpage, you'll find the longitude and latitude of the 30 national parks in alphabetical order (by the park name).

To simplify the problem, you may assume that the distance between two national parks is given by the distance

$$\text{distance} = \sqrt{(\text{longitude}_1 - \text{longitude}_2)^2 + (\text{latitude}_1 - \text{latitude}_2)^2}$$

To make sure your distance calculation is correct, the distance between Acadia and Arches (the first two listed in the file) is 41.75 units, and the total length of the tour that visits all the parks in alphabetical order and then returns to Acadia is 491.92.

You will now study an MCMC algorithm that is parameterized by a "temperature" parameter T that governs the tradeoff between local search and random search. The algorithm is as follows.

- Start with a uniformly random permutation of the $N = 30$ parks. This is your initial tour τ.

- For $i = 1, 2, \ldots,$ `NUMITER`:

  - Choose a uniformly random pair of consecutive parks in your current tour τ. Let τ' be the tour that results from switching the order of those two parks.

  - If τ' has lower total length than the current tour τ, set $\tau := \tau'$.

  - Otherwise, if $T > 0$, with probability $e^{-\Delta_d/T}$, update $\tau := \tau'$, where $\Delta_d = \text{length}(\tau') - \text{length}(\tau)$ is the difference in total length between τ' and τ.

(a) [2 points] Implement the MCMC algorithm. Have your algorithm also keep track of the *best* tour seen during all `NUMITER` iterations. Include your code.

How many states are in your Markov chain? If the number of iterations tends to infinity, would you eventually see all possible routes?

(b) [4 points] Set `NUMITER` to be 12,000. Run the algorithm in four different regimes, with $T \in \{0, 1, 10, 100\}$. For each value of T, run 10 trials. Produce one figure for each value of T. In each figure, plot the iteration number against the length of the current route τ of the route (**not** the current value of the best tour seen so far) during that iteration. Do this (in the same plot) for all 10 trials that share a value T. (Your solution will contain 4 figures, end each plot should have 10 lines.) Among the 4 value of T, which one seems to work the best?

(c) [2 points] Modify the above MCMC algorithm as follows: In each iteration, select two parks completely at random (as opposed to two consecutive parks in τ). Repeat the experiments for part (b). Among the 4 values of T, which one seems to work the best now?

(d) [4 points] Describe whatever differences you see relative to part (b), and propose an explanation of why this Markov chain performs differently than the previous one. If the best values of T are different in the two parts, try to provide an explanation. (You may wish to think about this problem relative to your investigation of the mixing times of different Markov chains in Problem 1.)

(e) [4 points] The algorithm we're running here is very similar to an algorithm widely used in practice called *simulated annealing*, which you can read about here:
`https://en.wikipedia.org/wiki/Simulated annealing`.
Explain what the main differences are, and briefly brainstorm (in 1-2 sentences) why it might make sense in practice to perform simulated annealing rather than what we do here.