CHE 477 October 20, 2016

Markov Chain Monte Carlo

Project 2

1 Problem Statement

Your goal is to use Markov Chain Monte Carlo to estimate the graphs that arise in a distribution network. Note this is not an optimization problem. Your Markov Chain will be a sequence of graphs. We do not know the transition probability matrix, but we can compute the relative probability of two graphs.

To be more specific, the state space is the set of *connected* undirected weighted graphs on a set of M vertices. Each graph in the sequence is defined by an adjacency matrix X_i, where i is the index of the sequence. We have a function, $\pi_j/\pi_i = f(X_i, X_j)$ which returns the relative probability of these graphs in the stationary distribution. That function is:

$$f(X_i, X_j) = e^{-(\theta(X_j)-\theta(X_i))/T} \tag{1}$$

where

$$\theta(X_i) = r\sum_e w_e + \sum_k^M \sum_{e\in p_{0k}} w_e \tag{2}$$

where $\sum_e$ is over all edges, w_e is weight of an edge, and $e \in p_{0k}$ is the set of edges in the path from node 0 to k. Remember that the graph *must* be connected, so θ must be non-zero. T and r are adjustable parameters. T is strictly positive.

The weights are calculated based on a 2D grid given as a parameter. Each vertex is a point on a 2D grid and the weight is the distance between them. Specifically, a set of M tuples, $m(\cdot)$, is passed as input and the edge weight between vertices i and j is $|m(j) - m(i)|$ (Euclidian distance).

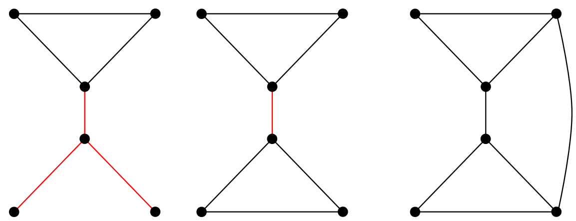
From these givens, construct a Markov Chain Monte Carlo algorithm to obtain the following results:

- Give examples of the top 1% most probable graphs under the stationary distribution
- What is the expected number of edges connected to vertex 0.
- What is the expected number of edges in the entire graph.
- What is the expected maximum distance of the shortest path in a graph that connects vertex 0 to another vertex.

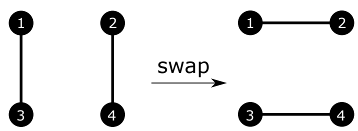
2 Proposal Distribution

There are multiple proposal distributions you can use, but part of the power of MCMC simulations is their use of the current state to propose a new state. A classic example is the random walk, which is achieved by adding a random increment the current position. Obviously then the next position will depend on the previous position.

You may consider adding or removing an edge on the current graph as a proposal distribution. i is the current graph and j is the proposed. To arrive at j choose one of the $M(M-1)/2$ edges in i and delete it if it exists or add it if it doesn't exist in i. This requires explicit calculation of the $q(i\,|\,j)/q(j\,|\,i)$ term. The graphs i and j will then be separated by a single edge mutation. The probability of $q(i\,|\,j)$ will be $1/W$, where W is the number of edge mutations that are possible at state j. $q(j \mid i)$ is defined similarly. The number of edge mutations is $M(M-1)/2 - B$, where B is the number of edges that, when cut, would make the graph disconnected. To identify the set of B edges, you can create a spanning tree over the graph and if an edge is in the spanning tree, it is in B. You can see three graphs below that are each separated by an edge mutation and their corresponding number of edges that cannot be deleted.



You may use other rules, such as completely generating a new adjacency matrix or a constrained switching. A constrained switching is where TWO edges' end vertices are swapped. For example, $[(1,5),(3,7)] \rightarrow [(1,7),(3,5)]$. Note that all four must be distinct. The constrained switching proposal distribution is nice because it guarantees that the graph is connected after the swap if it's already connected. Also, for a cosntrained swap $q(i \mid j)/q(j \mid i) = 1$. The constrained switching proposal results in a reducible Markov chain however, since it cannot result in a net gain or loss of an edge. So use it in conjunction with other proposal distributions. The rule is shown visually below



3 Intuition

This is an example of a centralized distribution network. For example, this could be a city with a single power plant and we need to determine the optimal way to distribute the electricity to each house. The term with paths is how much it will cost from losses in transmission to send the power to each house and the other term is the expense in constructing the network itself. The guarantee of a connected graph means each house will receive power. This might also be a model for laying out highways connecting downtown to various parts of the surrounding areas. It might also be a social tree, where one person can notify all others of important information by following the graph.

4 Requirements

Due dates and requirements:

10/21

1. Have a github repo in any language with 100% unit coverage
2. Be able to generate graphs from your proposal distribution

10/26: Complete the project objectives