IE531: Algorithms for Data Analytics
Spring, 2018
Homework 4: Markov Chains
Due Date: April 13, 2018

Instructions

1. You can modify any of the C++ code on Compass to solve these problems, if you want. It might help you with honing your programming skills. If these attempts (at using C++ code) is turning out to to be intense, you can use MATLAB just this once.

2. You will submit a PDF-version of your answers on Compass on-or-before midnight of the due date.

Instructions

1. (*20 points*) (**Random Walks on Graphs**) Let $G = (V, E)$ be a graph with a vertex set V and a set of edges $E \subseteq V \times V$. $G = (V, E)$ is *undirected* if $\forall v_1, v_2 \in V, ((v_1, v_2) \in E) \Leftrightarrow ((v_2, v_1) \in E)$. A *random walk* in a connected, undirected graph G is defined as follows – assume you are currently at vertex $v_1 \in V$, you pick one its d_v-many neighboring vertices (i.e. one-hop-adjacent vertices) with probability $\frac{1}{d_v}$, and proceed as often as necessary. Show that in steady-state you will spend $\frac{d_v}{2card(E)}$ % of time in vertex v, where $card(\bullet)$ is the cardinality (i.e. size) of the set-argument.

   Construct the $n \times n$ probability matrix $\mathbf{P}$ for the random walk. Show that $\boldsymbol{\pi}_v = \frac{d_v}{2card(E)}$ satisfies $\boldsymbol{\pi}\mathbf{P} = \boldsymbol{\pi}$. You have to figure out what $\sum_v d_v$ sums up to.

2. (*80 points*) (**Random Knight's Tour**) In this problem we start the knight at one of the four corner squares in an otherwise empty chessboard. The knight selects one of the next positions at random independently of the past moves.

   (a) (*5 points*) Interpret the Knight's tour as a Markov Chain, where the discrete-states represent the 64 squares of the chessboard. (**PS**: I am just looking for an implicit definition of the 64×64 probability-matrix here; if you want, give me a piece of pseudo-code that constructs the probability-matrix).

   The Knight moves in an "L-shaped" pattern. You have to figure out how to represent this on a (8×8) chessboard. Some moves of the Knight will not be permitted; some will be. For example, there are only two legal moves from the (1,1) spot on the chessboard.

   (b) (*20 points*) As a follow-on to problem 2a, is the Markov Chain *irreducible*? Is it *aperiodic*?

   Read the text (and watch me on Echo360) to figure out what *irreducible* and *aperiodic* means.

(c) (*50 points*) Find the stationary distribution of this Markov Chain.

This will require some thought and work. Look at our solution to problem 1 – the Knight is essentially taking a random-walk based on the number of legal-moves from its current position. Fill the (8×8) chessboard with 64 entries that correspond to the number of legal moves the Knight can make. For example there are two legal moves from (1,1), (1,8), (8,1) and (8,8) – so these four-spots in the (8×8) chessboard will have the entry "2." Now, fill the remaining 60 entries (it is not hard; there is a symmetry to this activity that you should exploit). Following this use the results of problem 1.

(d) (*5 points*) Where is the Knight most likely to be spotted in steady-state? Where is the Knight least likely to be spotted in steady-state?

Straightforward.