Max Kramer

I affirm that I have adhered to the honor code on this assignment.

Hello again, scientist! I'll do all my writing in italics, and problems for you will be in **bold.** *Comment your code, and* explain your ideas in plaintext. *As a general rule, I expect you to do at least as much writing as I do. Code should be part of your solution, but I expect variables to be clear and explanation to involve complete sentences. Cite your sources; if you work with someone in the class on a problem, that's an extremely important source. Don't work alone.*

Problem 11.05.02.

Here's a simple game: there's four squares in a line, A-B-C-D. You start on square A. On each turn, you flip a (fair) coin, advancing forward one square if you get heads and going backwards one square (or staying where you are, if you're already at A) if you get tails. You win on the turn that you get to square D (after which you could keep flipping coins, but you'll always stay at D from then on).

Write the stochastic matrix for this game; call it S.

```
S = [0.5 0.5 0 0;0.5 0 0.5 0;0 0.5 0 0; 0 0 0.5 1]
```

```
S =

    0.5000    0.5000         0         0
    0.5000         0    0.5000         0
         0    0.5000         0         0
         0         0    0.5000    1.0000
```

The matrix S represents the state trainsition matrix for the game. If you are on A, you can only move to B or stay on A, both with probabilities of 0.5. On B, you can only move to C or back to A, again with probabilities of 0.5. This also holds true for moving from C to D or C to B. Once you have reached D, there is certainty you will stay on D.

The first turn on which you can win is turn 3. ***What is the probability of that?***

```
prob = 0.5 * 0.5 * 0.5
```

```
prob =

    0.1250
```

In order to win in 3 turns, you must advance each turn. Each turn has a 50% chance of advancement, so one has a (0.5)^3 = 12.5% chance of winning in three turns.

Compute S^3. Explain what the first column represents.

```
Scubed = S^3
```

```
Scubed =

    0.3750    0.3750    0.1250         0
    0.3750    0.1250    0.2500         0
    0.1250    0.2500         0         0
    0.1250    0.2500    0.6250    1.0000
```

S^3 represents the probability of transitioning from each state to another state after 3 turns. The first column of S^3 represents the probability of moving from A to each state after 3 turns.

Find the steady-state vector for S and explain why your answer makes sense. *This is an example of what's called an "absorbing state."*

```
step1 = S - eye(4);
step2 = horzcat(step1,[0;0;0;0]);
rref(step2)
```

```
ans =

     1     0     0     0     0
     0     1     0     0     0
     0     0     1     0     0
     0     0     0     0     0
```

The steady state vector for S is [0;0;0;1]. After a certain number of turns have passed, you are certain to reach state D, which is inescapable.

What is the probability that you haven't won by the end of turn 30? *The fact that this number is more than .0001 is evidence that this game is completely terrible and unfun and goes on way too long a lot of the time. Game design is hard!*

```
Sthirty = S^30;
sum(Sthirty([1 2 3],1))
```

```
ans =

    0.0534
```

There is a 5.34% chance you will not have won the game by turn 30. This is calculated by summing the probability of transitioning from state A to states A through C after 30 turns.

If you're interested, here's the Chutes and Ladders analysis. http://www.datagenetics.com/blog/november12011/ I very strongly recommend looking at the section titled "Transition Matrix," where he shows heat maps of positions on different turns. This is a superposition; you're looking at a superposition. In particular, quantum tunneling works because there's a small but non-zero chance that a game is still going on after a thousand turns.

Published with MATLAB® R2019b