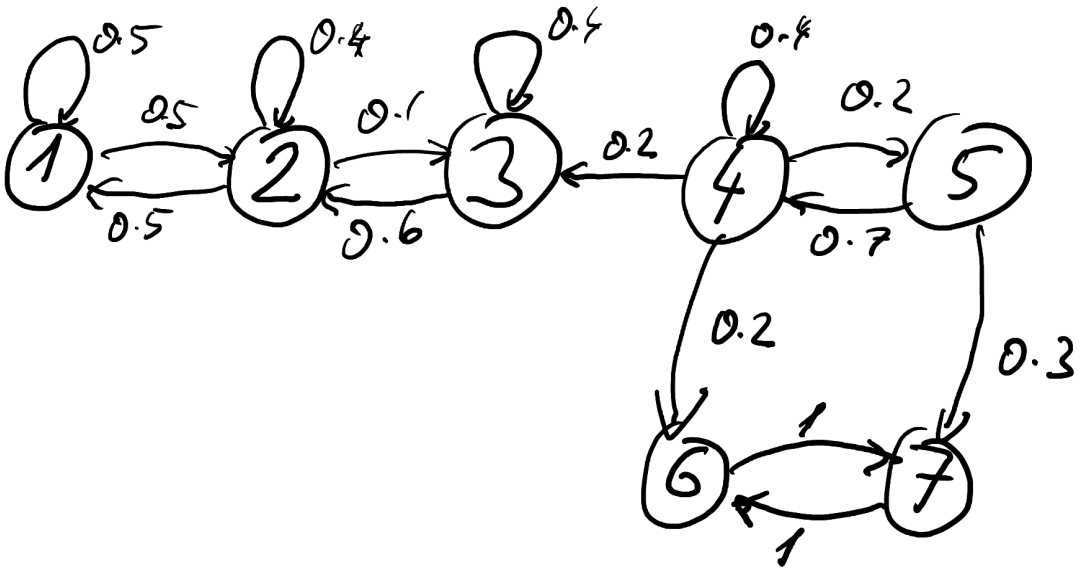
Exercise session 3 – Prob. & Stat. 2 — 27.10.2022

Misc

Problems 1–5 relate to the Markov chain on the Figure.



1. Identify the transient and recurrent states. Determine the equivalence classes of $\leftrightarrow$ and explain which of them are transient/recurrent, and which of the recurrent ones are periodic.

2. Does the distribution converge to a stationary distribution, if we start in state 1? If so, what is the stationary distribution?

3. Does the distribution converge to a stationary distribution, if we start in state 7? If so, what is the stationary distribution?

4. Assume the process starts in state 1, but we observe it after it (approximately) reaches stationary distribution.

(a) Find the probability that the state increases by 1 in the next step.

(b) Find the conditional probability that the process is in state 2 given that the first step is an increase by 1.

(c) Find the probability that the state increases by 1 in the next change of state. (I.e. changes by 0 do not count.)

5. Assume the process starts in state 4.

(a) For each recurrent class find the probability, that we reach that class at some time.

(b) What is the expected time till we reach some recurrent state?

---

6. Recall the Markov chain we used in the 2-SAT algorithm. Modify it by making $p_{0,0} = 1/2$ and $p_{0,1} = 1/2$. How does the mean time to reach n change?

7. In a gambler's ruin problem the player is repeatedly betting \$1 in a fair bet, until she loses ℓ_1 dollars or gains ℓ_2 dollars.

(a) What is the probability of the first and the second way how the game can end?

(b) What is the expected number of rounds before this happens?

8. A modification of the previous problem: the probability of losing a dollar is 2/3, probability of gaining it 1/3. Suppose that you start with i dollars and finish when you have n dollars or you have lost everything. Let W_t be the amount you have after t rounds of play.

(a) Show that $\mathbb{E}(2^{W_{t+1}}) = \mathbb{E}(2^{W_t})$.

(b) Use part (a) to determine the probability of finishing with 0 dollars and the probability of getting to n dollars, when starting with i dollars at the beginning.

(c) Verify that the probabilities satisfy the equations for the absorption probabilities we covered in the lecture.

(d) Generalize to any probability of losing $p > 1/2$.

(Hint: consider $\mathbb{E}(c^{W_t})$ for a suitable c.)

9. * There are $n+1$ sheep standing in a circle, let us label them $0, 1, \ldots, n$. A wolf eats sheep 0. Next, the wolf starts to random walk around the circle. At each position, he eats the sheep (if it is still there). Then he decides at random, which of the two directions to take next. (He does decide even on empty spots, where the sheep has been eaten already.)

(a) What is the probability, that the wolf eats sheep n as the last one?
(Hint: use Markov chains on a path.)

(b) What is the probability, that the wolf eats sheep i as the last one? (For any $i = 1, 2, \ldots, n$.)

10. * Let us define

$F_1(x,x) = P(\exists n > 0 : X_n = x \mid X_0 = x)$

$F_\infty(x,x) = P(\exists \text{ infinitely many } n > 0 : X_n = x \mid X_0 = x)$

$G(x,x) = \sum_{n \geq 0} P(X_n = x \mid X_0 = x)$

Show that for every state $x \in S$:

(a) If $F_1(x,x) = 1$ then $F_\infty(x,x) = 1$ and $G(x,x) = \infty$.

(b) If $F_1(x,x) < 1$ then $F_\infty(x,x) = 0$ and $G(x,x) = \frac{1}{1-F_1(x,x)} < \infty$

Random Walks

Given an undirected graph $G = (V, E)$ we consider a Markov chain with states V and transition probabilities $p_{u,v} = \frac{1}{\deg(u)}$ for every edge $uv \in E$. (That is, the person at vertex u is equally likely to use any edge to leave u.)

11. Show that $\pi_v = \frac{\deg(v)}{2|E|}$ is stationary distribution (assuming the graph is finite).

12. The Markov chain is aperiodic iff the graph is not bipartite.

13. * A maximum over all vertices $v \in V$ of the expected time to visit all other vertices, when we start at v is called the *cover time* of G.

(a) What do we know about cover time of a path?

(b) Compute (or estimate) a cover time of a clique.

(c) * Find a graph on n vertices with as large cover time as possible.