Homework 1: Markov Processes and Dynamic Programming Solutions

Problems

In square brackets are the points assigned to each problem.

1. [25 pts] Consider a Markov Chain with five states, $\mathcal{X} = \{1, 2, 3, 4, 5\}$ and the following transition matrix:

$$P = \begin{bmatrix} 1/2 & 1/4 & 0 & 0 & 1/4 \\ 1/4 & 1/2 & 1/4 & 0 & 0 \\ 0 & 1/4 & 1/2 & 1/4 & 0 \\ 0 & 0 & 1/4 & 1/2 & 1/4 \\ 1/4 & 0 & 0 & 1/4 & 1/2 \end{bmatrix} \tag{1}$$

   (a) Draw the state transition diagram for this chain (like the ones shown in the lecture). Is this Markov Chain irreducible? Is it periodic? Explain your answers.

   (b) What is the long-term behavior of this Markov chain? In other words, if this chain were initialized from an initial mass function $p_0 = \left[\frac{25}{150}, \frac{20}{150}, \frac{35}{150}, \frac{24}{150}, \frac{46}{150}\right]^T$, how would p_t evolve over time? Compare a simulated value of p_{100} to the theoretical value of $p_\infty := \lim_{t\to\infty} p_t$.

   (c) **The average age problem**. Consider a group of 5 people sitting at round table, in a way that each person can only talk with his/her right and left neighbor. The five people are aged 25, 20, 35, 24, and 46, respectively. Each person knows only their own age but can talk with his/her neighbors to obtain information from them. Based on your observations in parts a) and b) above, describe an approach for determining the average age of the five people, still under the assumption that the people can only talk with their neighbors.

Solutions

   (a) The transition diagram is shown in Fig. 1. This Markov Chain is irreducible and aperiodic. The

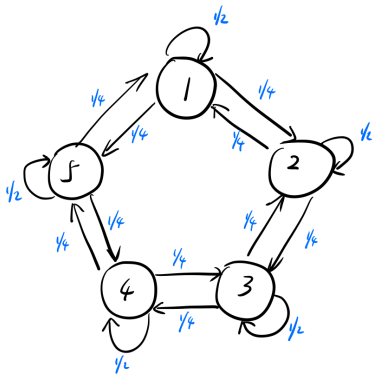


Figure 1: State transition diagram

   graph is strongly connected, we can go from every state to every state, so it is irreducible. Every state has a self loop so it is aperiodic.

   (b) The long-term behavior of this Markov chain is to average all states. p_t will asymptotically converge to $\left[\frac{30}{150}, \frac{30}{150}, \frac{30}{150}, \frac{30}{150}, \frac{30}{150}\right]^T$. From numerical simulation we obtain $p_{100} = \left[\frac{30}{150}, \frac{30}{150}, \frac{30}{150}, \frac{30}{150}, \frac{30}{150}\right]^T$. Using **Perron-Frobenius Theorem**, we know the 1 is a simple eigenvalue, associated eigenvector is $\mathbf{1}$ and the left eigenvector $w = \left[\frac{30}{150}, \frac{30}{150}, \frac{30}{150}, \frac{30}{150}, \frac{30}{150}\right]^T$ is the stationary distribution, i.e. $p_\infty = w$

   (c) Consider state $p0 = [25, 20, 35, 24, 46]^T$, using same transition matrix P, we can obtain the $p_\infty = [30, 30, 30, 30, 30]^T$.

2. [25 pts] Consider a system with the following motion model:

$$x_{t+1} = x_t u_t + u_t^2$$

The system state x_t can only take on values $\{-1, 0, 1, 2\}$, while the control input u_t is constrained to be -1 or 1. Let the planning horizon be $T = 2$, stage cost be $\ell(x, u) := xu$, and terminal cost be $\mathfrak{q}(x) = x^2$.

(a) Use dynamic programming to find an optimal policy.

(b) Find the optimal cost, control sequences, and state trajectory for $x_0 = 2$.

Solutions

(a) Define the transition matrices to be,

$$p(\cdot|x,-1) := \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \quad p(\cdot|x,1) := \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$V_T = q_T \Rightarrow V_2(-1) = 1, V_2(0) = 0, V_2(1) = 1, V_2(2) = 4$
Following the iteration:

$$Q_t(x,u) = l_t(x,u) + \mathbb{E}_{x'}[V_{t+1}(x')] \quad \forall x \in \mathcal{X}, u \in \mathcal{U}$$

$$V_t(x) = \min_{u \in \mathcal{U}} Q_t(x,u) \quad \forall x \in \mathcal{X}$$

$$\pi_t(x) = \arg\min_{u \in \mathcal{U}} Q_t(x,u) \quad \forall x \in \mathcal{X}$$

We get the following results:

$$Q_1(-1,1) = -1, Q_1(-1,-1) = 5, Q_1(0,1) = 1, Q_1(0,-1) = 1$$

$$Q_1(1,1) = 5, Q_1(1,-1) = -1, Q_1(2,1) = 6, Q_1(2,-1) = -1$$

$$\Rightarrow V_1(-1) = -1, V_1(0) = 1, V_1(1) = -1, V_1(2) = -1$$

$$\pi_1(-1) = 1, \pi_1(0) = \{1,-1\}, \pi_1(1) = -1, \pi_1(2) = -1$$

$$Q_0(-1,1) = 0, Q_0(-1,-1) = 0, Q_0(0,1) = -1, Q_0(0,-1) = -1$$

$$Q_0(1,1) = 0, Q_0(1,-1) = 0, Q_0(2,1) = -3, Q_0(2,-1) = 1$$

$$\Rightarrow V_0(-1) = 0, V_0(0) = -1, V_0(1) = 0, V_0(2) = -3$$

$$\pi_0(-1) = \{1,-1\}, \pi_0(0) = \{1,-1\}, \pi_0(1) = \{1,-1\}, \pi_0(2) = -1$$

(b) Given $x_0 = 2$, The control sequence applied by following π^* is $\{-1, 1\}$. The optimal cost $V^{\pi^*}(x_0) = -3$. The state trajectory is $\{2, -1, 0\}$.

3. [30 pts] You are controlling a linear system by selecting its modes of operation:

$$x_{t+1} = \begin{cases} Ax_t, & \text{if } u_t = 1, \\ Bx_t, & \text{if } u_t = 2 \end{cases}$$

where $x_t \in \mathbb{R}^n$ is the current state, and $A, B \in \mathbb{R}^{n\times n}$ are two given matrices. The stage and terminal costs of operating the system are the same and satisfy $\ell(x,u) \equiv \mathfrak{q}(x) := \frac{1}{2}x^T x$. This setup has applications in sensor scheduling and in embedded control systems with limited computation and communication resources.

(a) Use dynamic programming to show that the optimal cost-to-go/value function $V_t^\star(x)$ of the problem is the minimum of 2^{T-t} positive definite quadratic functions. Describe the optimal policy $\pi_t^\star$ using these functions.

(b) Consider the $T = 3$ horizon problem with matrices $A = \begin{bmatrix} 0.75 & -1 \\ 1 & 0.75 \end{bmatrix}$ and $B = \begin{bmatrix} 1 & 0.5 \\ 0.5 & 0.5 \end{bmatrix}$. Plot the regions of the state space where it is optimal to select action 1 and 2 respectively at time $t = 0, \dots, 3$, for example on a discretized $[-1, 1] \times [-1, 1]$ square around the origin. Also, plot the optimal value function $V_0^\star(x)$ on that space. Justify your plots through mathematical analysis.

Solutions

(a) *Proof.* Prove that the optimal cost-to-go/value function $V_t^*(x)$ of the problem is the minimum of 2^{T-t} positive definite quadratic functions. by induction.
Base case: When $t = T$, the optimal value function $V_T^*(x_T) = x_T^T x_T$ which is the minimum of $2^{T-T} = 1$ positive definite quadratic function.
Induction Hypothesis: Assume the statement holds when $t = k$, i.e. the optimal value function $V_k(x_k)$ is the minimum of 2^{T-k} positive definite quadratic functions.
Inductive Step: Prove the statement holds at $t = k-1$. i.e. the optimal value function $V_{k-1}(x_{k-1})$ is the minimum of $2^{T-\{k-1\}}$ positive definite quadratic functions. Since they are only two possible control inputs, from motion model and use dynamic programming we know

$$V_{k-1}^*(x_{k-1}) = \min\left(V_k^*(U_1 x_{k-1}), V_k^*(U_2 x_{k-1})\right) + x_{k-1}^T x_{k-1} \tag{2}$$

The first term of above equation, is the minimum of $2^{T-k} \cdot 2 = 2^{T-\{k-1\}}$ positive definite quadratic equations. The proof is completed by noticing that the sum of two positive definite matrices is still positive definite.

□

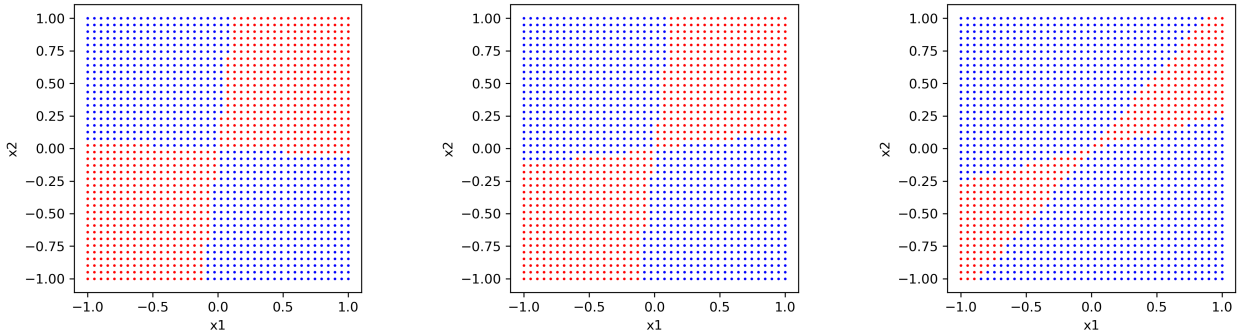
Therefore we can express the optimal policy as:

$$\pi_t^*(x) = \arg\min_{f\in\mathcal{F}} f(x) \tag{3}$$

where $\mathcal{F} = \{q_1, q_2, \dots, q_{2^{T-t}}\}$ is the set of 2^{T-t} quadratic functions.

(b) The optimal action at time $t = 0, 1, 2$ is shown in Fig. 2. The decision boundary are linear as expected, since finding optimal cost-to-go is equivalent to find minimum over a few quadratic functions. The optimal value function $V_0^*(x)$ at region $[-1, 1] \times [-1, 1]$ is shown in Fig. 3 The cost map has lower value around origin and higher value away from it, because the motion dynamic is trying to change the length and angle of vector x, and the state/terminal cost function is squared of those adjusted length. So state around at origin has small magnitude at beginning and therefore has lower optimal value.

4. Problem Solution is omitted.



(a) Optimal action t = 0 (b) Optimal action t = 1 (c) Optimal action t = 2

Figure 2: Optimal control selection. Red region:action 1; Blue region:action 2

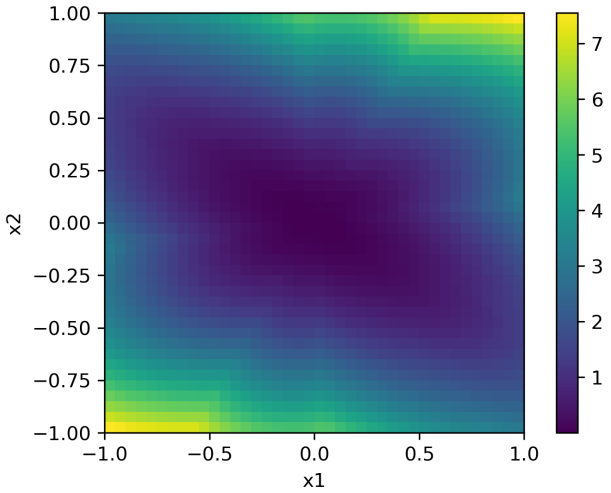


Figure 3: Optimal value function at region $[-1, 1] \times [-1, 1]$

5. [65 pts] Confident in your new found dynamic programming skills, you challenge your friend to a 100 game match of rock-paper-scissors. Your friend is a good player and knows to randomize his strategy but he is still biased towards one of the three options. You are not certain if your friend prefers rock, paper, or scissors but you know that he plays his preferred move 50% of the time and each of the other two options, 25% of the time. Your goal is to maximize your score, i.e., beat your friend with as large of a lead as possible by the end of the 100th game.

   (a) **Problem Formulation:** formulate this problem as a Markov Decision Process. Clearly define the state space $\mathcal{X}$, the control space $\mathcal{U}$, the motion model f or p_f , the initial state x_0 , the planning horizon T , the stage and terminal costs l, $\mathfrak{q}$, and any other elements necessary to make this a well defined problem. Note that your opponents bias towards one of the options is not directly observable.

   (b) **Technical Approach:** describe how you would go about finding an optimal policy for maximizing the expected lead with which you beat your opponent. Write down the explicit equations you would use to solve this problem. Implement your theoretical idea in python.

   (c) **Results:** use your python implementation to compare three different strategies *deterministic*, stochastic, and *optimal*. The deterministic strategy iterates rock, paper, scissors, rock, paper, scissors, rock, ... The stochastic strategy chooses among the three options uniformly at random for each game. The optimal strategy is the one you formulated and computed above. Provide a plot showing the mean and standard deviation over 50 100-game matches played by the three strategies with the number of games on the x axis, and the game score differential on the y axis. Assume here that your friend has a bias towards paper and generate 5000 plays from his strategy. Use the same data to compare the performance of the deterministic, stochastic, and optimal policies. Provide a discussion of your results.

Solutions

(a) **Problem Formulation**

- planning horizon $T = 100$
- state space $\mathcal{X} \coloneqq \{(\xi, b) \mid \xi \in X, b \in \mathcal{B}\} = X \times \mathcal{B}$ where ξ is the score differential of T games, $X = [-T, T] \in \mathbb{Z}$ i.e., Your #win - #lose of T games (in each game, win +1 score, draw 0 score, lose −1 score), and b is the inference of friend's preference distribution (pmf) on paper (P), scissor (S), rock (R). $\mathcal{B} = \left\{ b \in [0,1]^3 \mid \mathbf{1}^T b = 1 \right\}$.
- control space $\mathcal{U} = \{P, S, R\}$
- initial state $x_0 = [\xi_0, b_0]^T$, where $\xi_0 = 0$ and $b_0 = [1/3, 1/3, 1/3]^T$
- state cost $l(x, u) = 0$ and terminal cost $\mathfrak{q}(x_T) = -\xi$
- motion model
  Let $y \in \{P, S, R\}$ be the unknown preference of opponent, $z_t \in \{P, S, R\}$ be your opponent's play in t-th game. Then we known

$$p(z \mid y) = \begin{cases} 1/2, & \text{if z = y} \\ 1/4, & \text{otherwise} \end{cases} \tag{4}$$

The observation likelihood distribution is then

$$w(z) = \begin{bmatrix} p(z \mid y = R) \\ p(z \mid y = P) \\ p(z \mid y = S) \end{bmatrix} \tag{5}$$

So we can write the motion model as functions $x_{t+1} = \begin{bmatrix} b_{t+1} \\ \xi_{t+1} \end{bmatrix} = f(x_t, u_t, v_t)$, where the noise v_t is z_t

$$b_{t+1} = \frac{\mathbf{diag}\left(w(z_t)\right) b_t}{w(z_t)^T b_t} = \frac{w(z_t) \odot b_t}{w(z_t)^T b_t} \tag{6}$$

$$\xi_{t+1} = \xi_t + \mathbb{1}_{\{u_t \text{ beats } z_t\}} - \mathbb{1}_{\{z_t \text{ beats } u_t\}} \tag{7}$$

Note that, $\odot$ means the element-wise product. We can also formulate this motion model as $x_{t+1} \sim p_f(\cdot \mid x_t, u_t)$. More specifically,

$$b_{t+1} = \begin{cases} \frac{w(S)\odot b_t}{w(S)^T b_t} & \text{w.p.} \quad w(S)^T b_t \\ \frac{w(R)\odot b_t}{w(R)^T b_t} & \text{w.p.} \quad w(R)^T b_t \\ \frac{w(P)\odot b_t}{w(P)^T b_t} & \text{w.p.} \quad w(P)^T b_t \end{cases} \tag{8}$$

and we need to have probabilistic transition matrix for score differential as well, let P^R means the transition if $u_t = R$ Then we can obtain 3 different transition matrices:

$$P^R(\cdot \mid \xi_t) = \begin{bmatrix} \xi_t + 1 & \text{w.p. } w(S)^T b_t \\ \xi_t & \text{w.p. } w(R)^T b_t \\ \xi_t - 1 & \text{w.p. } w(P)^T b_t \end{bmatrix} \quad P^S(\cdot \mid \xi_t) = \begin{bmatrix} \xi_t & \text{w.p. } w(S)^T b_t \\ \xi_t - 1 & \text{w.p. } w(R)^T b_t \\ \xi_t + 1 & \text{w.p. } w(P)^T b_t \end{bmatrix} P^P(\cdot \mid \xi_t) = \begin{bmatrix} \xi_t - 1 & \text{w.p. } w(S)^T b_t \\ \xi_t + 1 & \text{w.p. } w(R)^T b_t \\ \xi_t & \text{w.p. } w(P)^T b_t \end{bmatrix} \tag{9}$$