Report for Problem 5

Yiran Xu

04/28/2019

1 Problem Formulation

We are asked to play rock-paper-scissors(RPS) and win our opponent, i.e. the times of our victories should be larger than our opponent. Our opponent plays with a preferred move 50% of the time and each of the other two 25%. However, we do not know exactly which move the opponent prefers. Our goal is to win our opponent in 100 times of game. If we denote the score for "win", "draw" and "lose" as {1, 0, -1}, then our goal is to outperform than our opponent's score, i.e. $s - s_o > 0$, where s represents the score of ours while s_o is the score of the opponent.
Specifically, we can formulate this problem as a Partially Observable Markov Decision Process (POMDP) problem.

- State space $\mathcal{X}$: $x_t \in \mathcal{X} = \{R_{ot}, P_{ot}, S_{ot}\} \times y$, which is *what your opponent plays* $\times$ *score differential between the opponent and me* at time t for $t = 1, ..., T-1$, where $\times$ is a Cartesian Product operator, score s is defined as -1 as our score is less than our opponent's, 0 as draw and 1 as our score is more than our opponent's.
- Control space $\mathcal{U}$: In this case, the control input u_t is what we decide to play at time t. Then the control space $\mathcal{U}_t = \{R_t, P_t, S_t\}$.
- Observation space $\mathcal{Z}$: This is based on how our opponent plays at time t, i.e. $\mathcal{Z}_t = \{R_{ot}, P_{ot}, S_{ot}\}$.
- Motion Model: Given x_t and u_t, x_{t+1} is what our opponent will play at next time. However, in this case, it seems that we can assume that x_{t+1} is independent of u_t since what we played at time t have nothing to do with our opponent's play because he has his own pattern. Thus $p_f(x_{t+1}|x_t, u_t) = p_f(x_{t+1}|x_t)$.
- Planning horizon: $T = 100$
- The stage and the terminal costs: Therefore we can set stage cost

$$l(x) = \begin{cases} 1 & \text{if win} \\ 0 & \text{if draw} \\ \text{-1} & \text{if lose} \end{cases} \quad (1)$$

.

Then we can convert this POMDP into MDP for DP algorithm implementation.

- State space $\mathcal{B} = \{\mathbf{b} \in [0,1]^3 \times y | \mathbf{1}^T\mathbf{b} = 1\}$, where $y \in \mathbb{R}$ is the cumulative score difference, $\mathbf{b}$ is the preference distribution of rock, paper and scissors of the opponent, i.e.

$$\mathbf{b}_t[0] = Pr(X_t = R|X_{1:t-1})$$
$$\mathbf{b}_t[1] = Pr(X_t = P|X_{1:t-1})$$
$$\mathbf{b}_t[2] = Pr(X_t = S|X_{1:t-1}).$$

  Actually,

$$\begin{aligned} Pr(X_t = R|X_{1:t-1}) = &Pr(X_t = R|prefer = R, X_{1:t-1})Pr(prefer = R|X_{1:t-1}) \\ &+ Pr(X_t = R|prefer = P, X_{1:t-1})Pr(prefer = P|X_{1:t-1}) \\ &+ Pr(X_t = R|prefer = S, X_{1:t-1})Pr(prefer = S|X_{1:t-1}) \end{aligned} \tag{2}$$

  The other two hold the same structure. For initialization $\mathbf{b_0} = [\frac{1}{3}, \frac{1}{3}, \frac{1}{3}]^T$.

- Control space $\mathcal{U}_t = \{R_t, P_t, S_t\}$. It is the same as the POMDP setting.

- Motion model: By using Bayes Rule,

$$Pr(prefer = R|X_{1:t-1}) = \frac{Pr(X_{1:t-1}|prefer = R)Pr(prefer = R)}{Pr(X_{1:t-1})}$$
$$Pr(prefer = P|X_{1:t-1}) = \frac{Pr(X_{1:t-1}|prefer = P)Pr(prefer = P)}{Pr(X_{1:t-1})}$$
$$Pr(prefer = S|X_{1:t-1}) = \frac{Pr(X_{1:t-1}|prefer = S)Pr(prefer = S)}{Pr(X_{1:t-1})}$$

  where

$$Pr(prefer = R) = Pr(prefer = P) = Pr(prefer = S) = \frac{1}{3},$$
$$Pr(X_{1:t-1}) = \sum_{prefer'} Pr(prefer'|X_{1:t-1})Pr(X_{1:t-1}|prefer')$$

$$Pr(X_{1:t-1}|prefer = R) = \prod_{X_i=R} Pr(X_i = R) \prod_{X_j=S} Pr(X_j = S) \prod_{X_k=P} Pr(X_k = P)$$

  with

$$Pr(X_i = R) = 0.5, Pr(X_j = S) = Pr(X_k = P) = 0.25.$$

  $Pr(X_{1:t-1}|prefer = P)$ and $Pr(X_{1:t-1}|prefer = S)$ are similar to this.

  Then we can plug those equations into Eq. 2 to update $\mathbf{b}_t$.

- Horizon $T = 100$.
- Costs: we can still use the same costs in POMDP as Eq.1.

Therefore, this problem is actually a simultaneous estimation and playing problem. Each round, we update our information about our opponent's preference and then use DP to decide the optimal policy. The state transition diagram is shown as Fig.1.

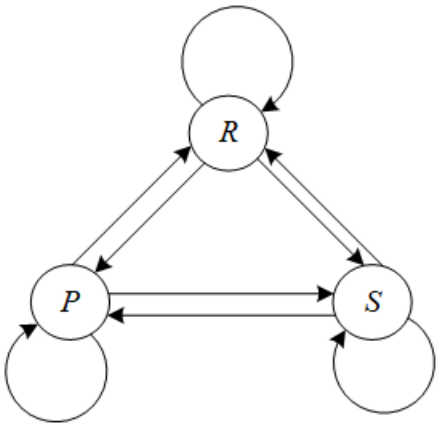


Figure 1: Transition Diagram for "Rock-Paper-Scissors"

2 Technical Approach

For each round, based on estimation of b_t, we can use Forward Dynamic Programming (DP) to determine the optimal policy. The algorithm for Forward DP is shown as Algorithm 1.

Algorithm 1: Forward Dynamic Programming Algorithm

Result: Optimal policy $\pi_t^*(x_t)$
T = 100, play set x_0, cost matrix $\mathbf{C}$
$V_0(x_0) = 0$
for $t = 1, 2, ..., T$ **do**
  $\mathrm{V}_t(\mathbf{b}_t) = \max_{u \in \mathcal{U}}(l(x_t, u_t) + \mathbb{E}_{x_{t+1} \sim p_f(*|x,u)}[V_{t+1}(x_{t+1})])$
  $\pi_t^*(\mathbf{b}_t) = argmax_{u \in \mathcal{U}}(l(x_t, u_t) + \mathbb{E}_{x_{t+1} \sim p_f(*|x,u)}[V_{t+1}(x_{t+1})])$
end

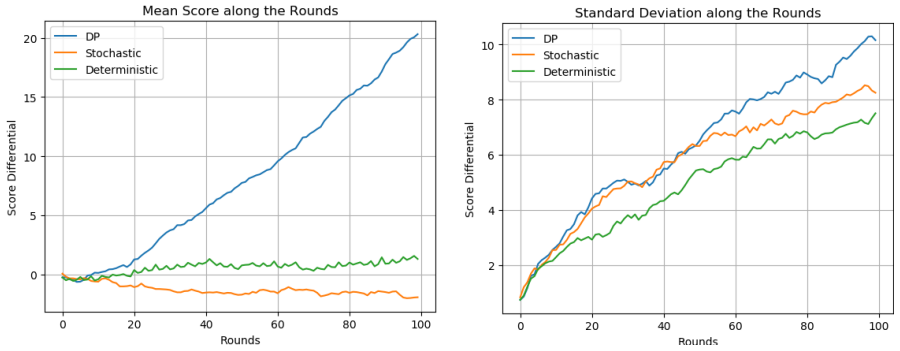
3 Results

3.1 Experiment Description

Since our opponent has a bias, we can assume he likes to play Paper. In order to explore the advantage using DP, three strategies - *deterministic, stochastic, and optimal (DP)*, are used. Here, *deterministic* strategy iterates *rock, paper, scissors, rock, paper, scissors, ...* Without losing generality, 50 times 100-game matches are played. Score y_t is the score differential at time t.

3.2 Results and Discussion

The results of 50 100-game matches are shown in Fig. 2. These figures show the mean values and standard deviation (std) as the number of rounds increases using three strategies. It is clear that DP outperforms the other two strategies as number of rounds increases from mean score. "Stochastic" strategy and "deterministic" nearly performs the same. For "optimal" strategy, it increases the score because as the game is going on we obtain more more information to estimate $\mathbf{b}_t$. Plus, std. of DP is the greatest in general, this is because the worst result will have big difference with the good results. This big deviation increases the std. Nevertheless, the other two strategies result in nearly same result: low score differential, so the stds are smaller.



(a) Mean (b) Standard Deviation

Figure 2: 50-Time Play: Mean and Std.

There are two more interesting things in this problem.
First of all, it's that the belief $\mathbf{b_t}$ in Eq. 2 is actually the same as following equations according to the results from my program:

$$
\begin{aligned}
\mathbf{b}_t[0] &= \frac{\#opponent\ played\ rock}{\#rounds} \\
\mathbf{b}_t[1] &= \frac{\#opponent\ played\ paper}{\#rounds} \\
\mathbf{b}_t[2] &= \frac{\#opponent\ played\ scissors}{\#rounds}
\end{aligned} \tag{3}
$$

This is the result followed by Maximum Likelihood Estimation (MLE), which indicates our optimal policy is actually the same as MLE.
Another one is the expectation in Algorithm 1. In this case, we know the belief $\mathbf{b}_t$, our optimal policy is always against our opponent's preference. This sounds intuitive but it is based on DP's choice.