Probabilistic Artificial Intelligence

Problem Set 5

Nov 23, 2018

1. Particle filter

Suppose that you have a robot, which is moving randomly through an 1-dimensional environment. You want to track the robot's position, x, which is discretized to integer values, $x \in \mathbb{Z}$. The robot's movement is modeled as a random walk,

$$x_{t+1} = x_t + \epsilon_t, \tag{1}$$

where ϵ_t is uniformly distributed and can take integer values in $[-3, 3]$. To track the robot, a sensor that measures the distance to the robot has been placed at the origin. The measurement model is

$$y_t = (x_t + \eta_t)^2, \tag{2}$$

where η_t is distributed according to

$$P(\eta_t) = \begin{cases} 0.6 & \text{if } \eta_t = 0 \\ 0.2 & \text{if } |\eta_t| = 1 \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

You want to use a particle filter with six particles to track the robot's position. At initial time, the robot is at the origin, $x_0 = 0$. Hence, the particles are initialized to $x_i = 0$, $i \in \{0, 1, 2, 3, 4, 5\}$.

(i) You draw samples from the distribution of ϵ_0 and obtain $(-1, -1, 0, 1, 2, 3)$. What is the position of the particles after the prediction update?

(ii) You obtain a measurement, $y_1 = 1$. What are the weights of the individual particles?

(iii) Are five particles enough to accurately estimate the state? Why/Why not?

(iv) Why would a Kalman filter not work reliably in this case?

2. Hero in the maze

Consider the following problem related to probabilistic planning. You are a hero H who is being chased by a ghost G in a maze.

(i) Suppose the maze is a simple (infinite) chain of nodes, each node labeled with a number (from $-\infty$ to ∞): H starts at 0, G starts at -2. H always tries to move away from G, but only succeeds with probability p, and with probability $1-p$ gets stuck (i.e., with probability p, H moves 1 step to the right, from node i to $i+1$, and gets 1 unit of reward; with probability $1-p$, H doesn't change its location and gets 0 units of reward). G always chases after H and never gets stuck. If G catches H, H incurs -10 reward in the timestep in which it got caught (and 0 reward in all subsequent time steps). Both G and H move simultaneously. Write down the state space with the transition probabilities. For a discount factor γ, what is the expected long term future reward as a function p and γ? Calculate its value for $p = .9$ and $\gamma = .95$. *Hint: You may want to consider the relative positions of H and G instead of their absolute positions when choosing your state representation.*

(ii) Now, suppose the maze is a "T", i.e. an (infinitely large) tree, where only one node, the starting node of H, has degree 3, all other nodes have degree 2. In the first round, H has the choice of either moving "right" and being chased (the same as above); or moving "down" and not being chased. If H moves "down", it will also get stuck with probability $1-p$ like above, but only incur reward 1/2 for each step moved (which happens with probability p. In all subsequent actions, H continues to (attempt to) move in the same direction as in the first round (i.e., once it decides to move right, it has to continue to move right etc.) What is the expected long term future reward in this case, as a function of p and γ? Calculate its value for $p = .9$ and $\gamma = .95$. For these values of p and γ, which initial action should H take? For a value of $\gamma = .95$, give an explicit rule on how H should choose its initial action as a function of p. Compute the critical (decision-relevant) values of p (you may have to do this numerically).

3. Policy iteration

Consider an undiscounted MDP having three states, (1, 2, 3), with rewards -1, -2, 0, respectively. State 3 is a terminal state. In states 1 and 2 there are two possible actions: a and b. The transition model is as follows:

- In state 1, action a moves the agent to state 2 with probability 0.8 and makes the agent stay put with probability 0.2.
- In state 2, action a moves the agent to state 1 with probability 0.8 and makes the agent stay put with probability 0.2.
- In either state 1 or state 2, action b moves the agent to state 3 with probability 0.1 and makes the agent stay put with probability 0.9.

Answer the following questions:

(i) Draw the MDP described above. What can be determined *qualitatively* about the optimal policy in states 1 and 2?

(ii) Apply policy iteration, showing each step in full, to determine the optimal policy and the values of states 1 and 2. Assume that the initial policy has action b in both states.

(iii) What happens to policy iteration if the initial policy has action a in both states? Does discounting help? Does the optimal policy depend on the discount factor?

4. Value iteration

In finite MDPs, the value function can be expressed as a vector that has as many entries as states in the state space, X. Given a value vector V, we defined the Bellman update operator, $\mathcal{B}(\cdot)$, for every element of V as follows:

$$\mathcal{B}(V(x)) = \max_a \big(r(x,a) + \gamma \sum_{x'} P(x'|x,a)V(x')\big). \tag{4}$$

Show that the Bellman operator is a contraction with respect to $\|\cdot\|_\infty$, that is to say that, for any V, V', holds that:

$$\max_{x \in X} |\mathcal{B}(V(x)) - \mathcal{B}(V'(x))| = \|\mathcal{B}V - \mathcal{B}V'\|_\infty \leq \gamma\|V - V'\|_\infty. \tag{5}$$