CS 520: INTRODUCTION TO ARTIFICIAL INTELLIGENCE SPRING 2020

Assignment 3

Temporal Models - Reinforcement Learning

Deadline: May 5th, 11:55pm.
Perfect score: 100.

Assignment Instructions:

Teams: Assignments should be completed by teams of up to two students. No additional credit will be given for students that complete an assignment individually. Please inform the TAs as soon as possible about the members of your team so they can update the scoring spreadsheet (find the TAs' contact info under the course's site on Sakai).

Submission Rules: Submit your reports electronically as a PDF document through Sakai (`sakai.rutgers.edu`). For programming questions, you need to also submit a compressed file via Sakai, which contains your code. Do not submit Word documents, raw text, or hardcopies etc. Make sure to generate and submit a PDF instead. Each team of students should submit only a single copy of their solutions and indicate all team members on their submission. Failure to follow these rules will result in lower grade in the assignment.

Late Submissions: No late submission is allowed. 0 points for late assignments.

Extra Credit for LaTeX: You will receive 10% extra credit points if you submit your answers as a typeset PDF (using LaTeX, in which case you should also submit electronically your source code). There will be a 5% bonus for electronically prepared answers (e.g., on MS Word, etc.) that are not typeset. If you want to submit a handwritten report, scan it and submit a PDF via Sakai. We will not accept hardcopies. If you choose to submit handwritten answers and we are not able to read them, you will not be awarded any points for the part of the solution that is unreadable.

Precision: Try to be precise. Have in mind that you are trying to convince a very skeptical reader (and computer scientists are the worst kind...) that your answers are correct.

Collusion, Plagiarism, etc.: Each team must prepare its solutions independently from other teams, i.e., without using common notes, code or worksheets with other students or trying to solve problems in collaboration with other teams. You must indicate any external sources you have used in the preparation of your solution. Do not plagiarize online sources and in general make sure you do not violate any of the academic standards of the department or the university. Failure to follow these rules may result in failure in the course.

Problem 1: Hidden Markov Models (50 points total)

You are up in your friend's apartment building, watching cars on the street below. They are far enough away that all you can see is their color. You want to catch a taxi home, so you are trying to reason about the probability that a given car is a taxi, given its color. You know that 75% of all taxis are yellow, and that only 10% of non-taxi cars are yellow. You also know that taxis are not likely to bunch up: The car following a taxi is another taxi only 25% of the time. However, non-taxi cars are followed by taxis 50% of the time. Assume 40% of all cars are taxis and 60% are non-taxis.

1. To formulate the above problem as a Hidden Markov Model (temporal model), give the transition model and the evidence model as conditional probabilities (use any correct notation you like).
2. You just saw a car but you could not tell whether it was yellow or not (no evidence at time $t = 0$), and now (at $t = 1$) you see a yellow car. What is the probability that the car you see (at $t = 1$) is a taxi?
3. What is the probability that the next car (at $t = 2$) will be a taxi?
4. You observe that the next car (at $t = 2$) is also yellow. Use this new information to update your belief that the previous car you saw (at $t = 1$) was a taxi.
5. Explain qualitatively why your new estimate in Question 4 makes sense, given your evidence and transition models (I am looking here for a short informal explanation in English).
6. Using the Viterbi algorithm, what is the most likely sequence of cars (taxi or non-taxi) at times 0, 1 and 2 given the observations at times 1 and 2?

Problem 2: Kalman Filters (10 points total)

Consider the problem of implementing an autopilot system for a boat. The boat's position is represented by two coordinates (X_t, Y_t). The boat is navigating with a constant velocity $v_x = 10\, meters/second$ in the X-direction and $v_y = 5\, meters/second$ in the Y-direction. Due to random ocean currents, the boat's position at time $t+1$ is distributed as $X_{t+1} \sim N\big(X_t + v_x, 1\big)$, $Y_{t+1} \sim N\big(Y_t + v_y, 1\big)$. You also have access to the Global Positioning System (GPS). When the boat is in a position (X_t, Y_t), the coordinates returned by the GPS are $\hat{X}_t \sim N\big(X_t, 3\big)$, $\hat{Y}_t \sim N\big(Y_t, 3\big)$. The initial position of the boat is not know precisely, it is given by $X_0 \sim N\big(0, 0.1\big)$, $Y_0 \sim N\big(0, 0.1\big)$.
The autopilot repeatedly estimates the position of the boat.

1. Calculate the distribution of the boat's position at time $t = 1$ (after sailing for one second), if the estimated position according to the GPS at time $t = 1$ is $(8, 6)$.
2. Calculate the distribution of the boat's position at time $t = 2$ (after sailing for two seconds), if the estimated position according to the GPS at time $t = 2$ is $(19, 9)$.

Problem 3: Markov Decision Processes (40 points total)
The autopilot system does not only estimate the position of the boat, it also controls it. Consider here a simplified version of this control problem.

- Assume the position of the boat is discret and takes values from the grid $\{(0,0), (0,1), (0,2), (1,0), (1,1), (1,2), (2,0), (2,1), (2,2)\}$.
- The boat's position is always known.
- The boat's actions are move-east, move-west, move-north , move-south.
- $(0,0)$ is the most west-north position, and $(2,2)$ is the most east-south position.
- The boat moves into the intended adjacent position with probability 0.9, and to a random position among the other adjacent positions with probability 0.1.
  Example 1: $P((0,1) \mid (1,1), north) = 0.9$, $P((1,0) \mid (1,1), north) = \frac{0.1}{3}$, $P((1,2) \mid (1,1), north) = \frac{0.1}{3}$, $P((2,1) \mid (1,1), north) = \frac{0.1}{3}$.
  Example 2: $P((0,1) \mid (0,0), east) = 0.9$, $P((1,0) \mid (0,0), east) = 0.1$.
- When the boat tries to move outside the grid, it remains in the same position with probability 1.

- The initial position is $(0, 0)$
- The reward of all the states is 0, except for state $(0, 2)$ (the goal) where the reward is 10, and for state $(0, 1)$ (iceberg) where the reward is -5.

Using a discount factor $\gamma = 1$:

1. Show the policy π and value function V^π at each iteration of the Value Iteration algorithm for 2 iterations. The initial values are set to 0.
2. Show the policy π and value function V^π at each iteration of the Policy Iteration algorithm for 2 iterations. Run the policy evaluation part of the Policy Iteration algorithm for 2 iterations only. The initial values are set to 0, and the initial policy for all the states is move east.