CS 344 (Spring 2018): Class Test 2

Instructor: Shivaram Kalyanakrishnan

11.05 a.m. – 12.00 p.m., February 16, 2018, 103 New CSE Building

Total marks: 15

Note. Provide brief justifications and/or calculations along with each answer to illustrate how you arrived at the answer.

Question 1. Let $M = (S, A, T, R, \gamma)$ be an arbitrary MDP. It is a well-known result that if $\pi : S \to A$ is not an optimal policy for M, then there exists a policy $\pi' : S \to A$ such that

1. π and π' differ in exactly one state; that is,

$$\exists \bar{s} \in S : ((\pi'(\bar{s}) \neq \pi(\bar{s})) \wedge (\forall s \in S \setminus \{\bar{s}\} : \pi'(s) = \pi(s))), \text{ and}$$

2. π' strictly dominates π; that is,

$$(\forall s \in S : V^{\pi'}(s) \geq V^{\pi}(s)) \wedge (\exists s \in S : V^{\pi'}(s) > V^{\pi}(s)).$$

Your task is to use this result to design a hill climbing algorithm that searches over policies for M, and returns an optimal policy. Your answers to 1a, 1b, and 1c must work together to realise a successful algorithm.

1a. Provide pseudocode for the $f()$ function, which guides the hill climbing step. The function must take a policy as input, and compute a scalar (a real number) as its output. If $f()$ requires you to solve a set of equations E, you can assume access to a function called $solve()$ that returns the solution of E. You do not have to implement $solve()$ yourself—but make sure you fully specify E. [2 marks]

1b. Provide pseudocode for the $getNeighbours()$ function, which must take a policy as input, and compute a set of policies as output. What is the cardinality of the set of policies it returns? [2 marks]

1c. Provide pseudocode for hill climbing, using $getNeighbours()$ and $f()$ as subroutines. [1 mark]

1d. Prove that your code returns an optimal policy. [1 mark]

1e. What is the computational complexity of $f()$? What is the best upper bound you can provide on the computational complexity of the entire hill climbing procedure to find an optimal policy? [1 mark]

Assume the MDP M is available to all your functions; you do not have to pass it as an argument.

Question 2. Consider an MDP (S, A, T, R, γ) in which all the rewards are non-negative: that is, for $s, s' \in S, a \in A : R(s, a, s') \geq 0$. Let $V^\star : S \to \mathbb{R}$ be the optimal value function of the MDP.

Now suppose that we run the Value Iteration algorithm on this MDP, and we begin with an iterate $V_0 : S \to \mathbb{R}$ that is uniformly zero: that is, $\forall s \in S : V_0(s) = 0$. For $t \geq 0$, V_{t+1} is obtained by applying the Bellman Optimality Operator to V_t.

Show that all iterates remain upper-bounded by the optimal value function. In other words, show that for $t \geq 0, s \in S : V_t(s) \leq V^\star(s)$. [3 marks]

Question 3. Let X, Y, and Z be Boolean random variables. X takes values x and $\neg x$; Y takes values y and $\neg y$; and Z takes values z and $\neg z$.

3a. If X and Y are independent, and Y and Z are independent, does it follow that X and Z are independent. In other words, is true that $(X \perp Y) \wedge (Y \perp Z) \implies (X \perp Z)$? Prove that your answer is correct. [2 marks]

3b. Take the eight elements of the form $\mathbb{P}\{X \wedge Y \wedge Z\}$ where $X \in \{x, \neg x\}$, $Y \in \{y, \neg y\}$, and $Z \in \{z, \neg z\}$, as *atoms.* Express each of the following probabilities in terms of the atoms, combined only using the operations of addition, subtraction, multiplication, and division.

3b(i). $\mathbb{P}\{(x \vee y)|z\}$. [1 mark]

3b(ii). $\mathbb{P}\{x|(y \vee z)\}$. [1 mark]

3b(iii). $\mathbb{P}\{x \vee (y \wedge z)\}$. [1 mark]

Solutions

1a. Here is one possible way to define $f()$.

```
f(π)

Obtain $V^\pi$ by solving Bellman's Equations: for $s \in S$:
$V^\pi(s) = \sum_{s' \in S} T(s, \pi(s), s')\{R(s, \pi(s), s') + \gamma V^\pi(s')\}$.
$f \leftarrow \sum_{s \in S} V^\pi(s)$.
Return $f$.
```

1b. In order to use the result provided, we define as a neighbour of π every policy that differs from π in exactly one position.

```
getNeighbours(π)

$N \leftarrow \emptyset$.
For $s \in S$:
    For $a \in A \setminus \{\pi(s)\}$:
        Set $\pi'$ such that $\pi'(s) = a$ and for $s' \in S \setminus \{s\}$, $\pi'(s') = \pi(s')$.
        $N \leftarrow N \cup \{\pi'\}$.
Return $N$.
```

We observe that $|N| = |S|(|A| - 1)$.

1c. The following is the high-level template for hill climbing.

```
hillClimbing

$\pi \leftarrow$ Arbitrary initial policy.
While True:
    $N \leftarrow getNeighbours(\pi)$.
    $\pi' \leftarrow \arg\max_{\pi'' \in N} f(\pi'')$.
    If $f(\pi') > f(\pi)$
        $\pi \leftarrow \pi'$.
    else
        Return $\pi$.
```

1d. From the result given, and from our definitions of $f()$ and $getNeighbours()$, it follows that every non-optimal policy will have at least one neighbour that has a strictly higher f value. The only policies that will not have improving neighbours are optimal policies—one of which is guaranteed to be reached since no policy is visited more than once, and the total number of policies is finite.

1e. $f()$ primarily involves the solution of linear equations, which need at most take $O(|S|^3 + |S|^2|A|)$ arithmetic operations. The number of policies visited is at most $|A|^{|S|}$, which gives an overall upper bound of $\text{poly}(|S|, |A|) \cdot |A|^{|S|}$ arithmetic operations.

2. The application of the Bellman Optimality Operator yields, for $t \geq 0, s \in S$:

$$V_{t+1}(s) = \max_{a \in A} \left(\sum_{s' \in S} T(s, a, s') \{ R(s, a, s') + \gamma V_t(s')) \} \right).$$

Since V_0 is the zero function, and all the rewards and transition probabilities are non-negative, it follows that for $s \in S$, $V_1(s) \geq V_0(s)$. This observation becomes the base case of our induction to show that for $t \geq 0, s \in S$, $V_{t+1}(s) \geq V_t(s)$. For $t \geq 1$, the induction hypothesis establishes that

$$\begin{aligned} V_{t+1}(s) &= \max_{a \in A} \left(\sum_{s' \in S} T(s, a, s') \{ R(s, a, s') + \gamma V_t(s')) \} \right) \\ &\geq \max_{a \in A} \left(\sum_{s' \in S} T(s, a, s') \{ R(s, a, s') + \gamma V_{t-1}(s')) \} \right) \\ &= V_t(s). \end{aligned}$$

We have shown that for $s \in S$, $V_0(s) \leq V_1(s) \leq V_2(s) \leq \ldots$. Suppose for some $t > 0$, it is true that $V_t(s) > V^\star(s)$, it would follow that $\lim_{t \to \infty} V^t(s) > V^\star(s)$, which contradicts the well-known result that $\lim_{t \to \infty} V^t(s) = V^\star(s)$.

An intuitive way to reason out this result is as follows. If V_0 is uniformly zero, it follows that for $t \geq 1, s \in S$, $V_t(s)$ is the maximal expected t-step discounted reward possible from s. One the other hand, $V^\star(s)$ is the maximal expected infinite discounted reward possible from s. The difference $V^\star(s) - V_t(s)$ is therefore an expected infinite discounted reward starting at the $(t+1)$-st transition—which is non-negative because the rewards are all non-negative.

3a. X and Z need not be independent. Perhaps the easiest way to verify the claim is to set $X = Z$. Consider the following joint probability distribution, in which X and Y are picked uniformly at random (each outcome with probability 1/2), and Z is a copy of X.

X	Y	Z	$\mathbb{P}\{X, Y, Z\}$
x	y	z	0.25
x	y	$\neg z$	0
x	$\neg y$	z	0.25
x	$\neg y$	$\neg z$	0
$\neg x$	y	z	0
$\neg x$	y	$\neg z$	0.25
$\neg x$	$\neg y$	z	0
$\neg x$	$\neg y$	$\neg z$	0.25

Observe that $0 = \mathbb{P}\{x, \neg z\} \neq \mathbb{P}\{x\}\mathbb{P}\{\neg z\} = 0.5 \times 0.5$, establishing X and Z are not independent.

3b(i).

$$\mathbb{P}\{(x \vee y)|z\} = \frac{\mathbb{P}\{(x \vee y) \wedge z\}}{\mathbb{P}\{z\}}$$
$$= \frac{\mathbb{P}\{x \wedge y \wedge z\} + \mathbb{P}\{x \wedge \neg y \wedge z\} + \mathbb{P}\{\neg x \wedge y \wedge z\}}{\mathbb{P}\{x \wedge y \wedge z\} + \mathbb{P}\{x \wedge \neg y \wedge z\} + \mathbb{P}\{\neg x \wedge y \wedge z\} + \mathbb{P}\{\neg x \wedge \neg y \wedge z\}}.$$

3b(ii).

$$\mathbb{P}\{x|(y \vee z)\} = \frac{\mathbb{P}\{x \wedge (y \vee z)\}}{\mathbb{P}\{y \vee z\}}$$
$$= \frac{\mathbb{P}\{x \wedge y \wedge z\} + \mathbb{P}\{x \wedge y \wedge \neg z\} + \mathbb{P}\{x \wedge \neg y \wedge z\}}{\mathbb{P}\{x \wedge y \wedge z\} + \mathbb{P}\{x \wedge y \wedge \neg z\} + \mathbb{P}\{x \wedge \neg y \wedge z\} + \mathbb{P}\{\neg x \wedge y \wedge z\} + \mathbb{P}\{\neg x \wedge y \wedge \neg z\} + \mathbb{P}\{\neg x \wedge \neg y \wedge z\}}.$$

3b(iii).

$$\mathbb{P}\{(x \vee (y \wedge z))\} = \mathbb{P}\{x \wedge y \wedge z\} + \mathbb{P}\{x \wedge y \wedge \neg z\} + \mathbb{P}\{x \wedge \neg y \wedge z\} + \mathbb{P}\{x \wedge \neg y \wedge \neg z\} + \mathbb{P}\{\neg x \wedge y \wedge z\}.$$