CMPSCI 687 Homework 1 - Fall 2022

Due **September 30, 2022**, 11:55pm Eastern Time

Note: we have added comments, below, to clarify some of the problems/questions in this homework. All parts of this document that were changed with respect to the original one are shown in blue.

1 Instructions

This homework assignment consists of a written portion and a programming portion. While you may discuss problems with your peers (e.g., to discuss high-level approaches), you must answer the questions on your own. In your submission, do explicitly list all students with whom you discussed this assignment. Submissions must be typed (handwritten and scanned submissions will not be accepted). You must use LaTeX. The assignment should be submitted on Gradescope as PDF with marked answers via the Gradescope interface. The source code should be submitted via the Gradescope programming assignment as a .zip file. Include with your source code instructions for how to run your code. You **must** use Python 3 for your homework code. You may not use any reinforcement learning or machine learning specific libraries in your code, e.g., TensorFlow, PyTorch, or scikit-learn. You *may* use libraries like numpy and matplotlib, though. The automated system will not accept assignments after 11:55pm on September 30. The tex file for this homework can be found here.

Discussion with Arundhati Gorkhe, Sharanya Kamath and TAs.

2 Hints and Probability Review

- **Write Probabilities of Events:** In some of the probability hints below that are not specific to RL, we use expressions like $\Pr(a|b)$, where a and b are events. Remember that in the RL notation used for this class, the values of $\Pr(s_0)$, $\Pr(a_0)$, $\Pr(A_0)$, or $\Pr(A_0|S_0)$ are all undefined, since those are simply states, actions, or random variables (not events). Instead, we ***must*** write about the probabilities of events. For example: $\Pr(A_0 = a_0)$ or $\Pr(A_0 = a_0|S_0 = s_0)$.

- **Bayes' Theorem:** $\Pr(a|b) = \frac{\Pr(b|a)\Pr(a)}{\Pr(b)}$. This is useful for dealing with conditional probabilities $\Pr(a|b)$ if the event a occurs *before* event b. For example, it is often difficult to work with an expression like $\Pr(S_0 = s_0|A_0 = a_0)$, because the agent *first* observes the current state, S_0, and only afterwards selects an action, A_0; in this case, it is much easier to deal with the 3 terms in $\frac{\Pr(A_0=a_0|S_0=s_0)\Pr(S_0=s_0)}{\Pr(A_0=a_0)}$.

- **The law of total probability:** For event a, and a set of events $\mathcal{B}$,

$$\Pr(a) = \sum_{b\in\mathcal{B}} \Pr(b)\Pr(a|b)$$

  See the example below for several useful applications of this property.

- **"Extra" given terms:** Remember that when applying laws of probability, any "extra" given terms stay in the result. For example, applying the law of total probability:

$$\Pr(a|c,d) = \sum_{b\in\mathcal{B}} \Pr(b|c,d)\Pr(a|c,d,b)$$

- **Conditional Probabilities - Useful property #1:** If you need to move terms from the "right-hand side" of a conditional probability to the "left-hand side", you can use the following identity: $\Pr(a|b,c) = \frac{\Pr(a,b|c)}{\Pr(b|c)}$

- **Conditional Probabilities - Useful property #2:** If you need to move terms from the "left-hand side" of a conditional probability to the "right-hand side", you can use the following identity: $\Pr(a,b|c) = \Pr(a|b,c)P(b|c)$

- **Expected Values:** The expected value of a random variable X with possible outcomes in $\mathcal{X}$ is

$$\mathbb{E}[X] = \sum_{x\in\mathcal{X}} x\Pr(X = x)$$

- **Conditional Expected Values:** The expected value of a random variable X with possible outcomes in $\mathcal{X}$, conditioned on an event $A = a$, is

$$\mathbb{E}[X \mid A = a] = \sum_{x \in \mathcal{X}} x \Pr(X = x \mid A = a)$$

- **Example problem:** The probability that the state at time $t = 1$ is $s \in \mathcal{S}$.

$$\begin{aligned}
\Pr(S_1 = s) &= \sum_{s_0 \in \mathcal{S}} \Pr(S_0 = s_0) \Pr(S_1 = s | S_0 = s_0) && (1)\\
&= \sum_{s_0 \in \mathcal{S}} d_0(s_0) \Pr(S_1 = s | S_0 = s_0) && (2)\\
&= \sum_{s_0 \in \mathcal{S}} d_0(s_0) \sum_{a_0 \in \mathcal{A}} \Pr(A_0 = a_0 | S_0 = s_0) && (3)\\
&\quad \times \Pr(S_1 = s | S_0 = s_0, A_0 = a_0) && (4)\\
&= \sum_{s_0 \in \mathcal{S}} d_0(s_0) \sum_{a_0 \in \mathcal{A}} \pi(s_0, a_0)\, p(s_0, a_0, s). && (5)
\end{aligned}$$

Part One: Written (55 Points Total)

1. ***(Your grade will be a zero on this assignment if this question is not answered correctly)*** *Read the class syllabus carefully, including the academic honesty policy. To affirm that you have read the syllabus, type your name as the answer to this problem.*

   Solution 1: Saurabh Bajaj. Discussion with Arundhati Gorkhe, Sharanya Kamath and TAs.

2. (**18 Points**) Given an MDP $M = (\mathcal{S}, \mathcal{A}, p, R, d_0, \gamma)$ and a fixed policy, π, the probability that the action at time $t = 0$ is $a \in \mathcal{A}$ is:

$$\Pr(A_0 = a) = \sum_{s \in \mathcal{S}} d_0(s) \pi(s, a). \qquad (6)$$

   Write similar expressions (using only $\mathcal{S}, \mathcal{A}, p, R, d_0, \gamma$, and π) for the following problems.

   Important:

   - *Assume, below, that the reward function will be in the form $R : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$. That is, the reward at time t depends only on the state at time t and action at time t.*
   - *All solutions below need to be derived from "first principles": you should repeatedly apply definitions and properties of probability distributions such as the ones discussed in Section 2, as well as the Markov Property (when appropriate), and then replace the relevant quantities with their corresponding definitions in RL (e.g., you can substitute $\Pr(A_0 = a | S_0 = s)$ with $\pi(s, a)$).*
   - *Remember that the Markov Property allows you to ignore history information, prior to time t, if you know S_t (that is, if the probability term is conditioned on S_t). It does not allow you to ignore variables associated with time t or any future times ($t + 1$, $t + 2$, etc). For instance:*

$$\Pr(S_1 = s_1 | A_1 = a_1, S_0 = s_0) \neq \Pr(S_1 = s_1 | S_0 = s_0)$$

   and

$$\Pr(S_2 = s_2 | S_4 = s_4, S_1 = s_1) \neq \Pr(S_2 = s_2 | S_1 = s_1).$$

- *When writing the final answers to the problems below (2a-2d), please reorganize your terms and summations in "temporal" order. For instance, instead of presenting your final answer as*

  $$\sum_{s_1} p(s_1, a_1, s_2)\pi(s_1, a_1) \sum_{a_0} p(s_0, a_0, s_1)\pi(s_0, a_0)$$

  rewrite it as follows:

  $$\sum_{a_0} \pi(s_0, a_0) \sum_{s_1} p(s_0, a_0, a_1)\pi(s_1, a_1)p(s_1, a_1, s_2).$$

Problems:

- **(Question 2a. 4 Points)** What is the expected reward at time $t = 8$ given that the state at time $t = 7$ is s_7 and the action at time $t = 6$ is a_6?

  (Solution 2a.)

  $$\mathbb{E}[R_8|S_7 = s_7, A_6 = a_6] \quad = \quad \sum_{r\in\mathcal{R}} rPr(R_8 = r|S_7 = s_7, A_6 = a_6)$$

  Evaluating RHS

  $$\begin{aligned} RHS &= \sum_{r\in\mathcal{R}} rPr(R_8 = r|S_7 = s_7, A_6 = a_6) \\ &= \sum_{r\in\mathcal{R}} r \sum_{s_8\in\mathcal{S}} Pr(S_8 = s_8|S_7 = s_7, A_6 = a_6)Pr(R_8 = r|S_8 = s_8, S_7 = s_7, A_6 = a_6) \\ &= \sum_{r\in\mathcal{R}} r \sum_{s_8\in\mathcal{S}} \sum_{a_7\in\mathcal{A}} Pr(A_7 = a_7|S_7 = s_7, A_6 = a_6)Pr(S_8 = s_8|S_7 = s_7, A_6 = a_6, A_7 = a_7)Pr(R_8 = r|S_8 = s_8) \end{aligned}$$

  Probability of being in state 8 depends on the state at t=7 and action at t=7, and probability of taking action at t=7 depends on state 7.

  $$\begin{aligned} RHS &= \sum_{r\in\mathcal{R}} r \sum_{s_8\in\mathcal{S}} \sum_{a_7\in\mathcal{A}} Pr(A_7 = a_7|S_7 = s_7)Pr(S_8 = s_8|S_7 = s_7, A_7 = a_7)Pr(R_8 = r|S_8 = s_8) \\ &= \sum_{r\in\mathcal{R}} r \sum_{s_8\in\mathcal{S}} \sum_{a_7\in\mathcal{A}} Pr(A_7 = a_7|S_7 = s_7)Pr(S_8 = s_8|S_7 = s_7, A_7 = a_7) \times \\ &\quad \sum_{a_8\in\mathcal{A}} Pr(A_8 = a_8|S_8 = s_8)Pr(R_8 = r|S_8 = s_8, A_8 = a_8) \\ &= \sum_{r\in\mathcal{R}} r \sum_{s_8\in\mathcal{S}} \sum_{a_7\in\mathcal{A}} \pi(s_7, a_7)p(s_7, a_7, s_8) \sum_{a_8\in\mathcal{A}} \pi(s_8, a_8)Pr(R_8 = r|S_8 = s_8, A_8 = a_8) \\ &= \sum_{a_7\in\mathcal{A}} \pi(s_7, a_7) \sum_{s_8\in\mathcal{S}} p(s_7, a_7, s_8) \sum_{a_8\in\mathcal{A}} \pi(s_8, a_8) \sum_{r\in\mathcal{R}} rPr(R_8 = r|S_8 = s_8, A_8 = a_8) \\ &= \sum_{a_7\in\mathcal{A}} \pi(s_7, a_7) \sum_{s_8\in\mathcal{S}} p(s_7, a_7, s_8) \sum_{a_8\in\mathcal{A}} \pi(s_8, a_8)R(s_8, a_8) \end{aligned}$$

- **(Question 2b. 4 Points)** What is the probability that the state at time $t = 64$ is s_{64} given that the state at time $t = 62$ is s_{62}?

  (Solution 2b.)

  $$Pr(S_{64} = s_{64}) \quad = \quad \sum_{s_{63}\in\mathcal{S}} Pr(S_{63} = s_{63})Pr(S_{64} = s_{64}|S_{63} = s_{63})$$

using the derivation in the example problem

$$Pr(S_1 = s|S_0 = s_0) \quad = \quad \sum_{a_0 \in \mathcal{A}} \pi(s_0, a_0)\, p(s_0, a_0, s).$$

We get,

$$\begin{aligned}
Pr(S_{64} = s_{64}) \quad &= \quad \sum_{s_{63} \in \mathcal{S}} Pr(S_{63} = s_{63}) \sum_{a_{63} \in \mathcal{S}} \pi(s_{63}, a_{63}) p(s_{63}, a_{63}, s_{64}) \\
&= \quad \sum_{s_{63} \in \mathcal{S}} \sum_{s_{62} \in \mathcal{S}} Pr(S_{62} = s_{62}) Pr(S_{63} = s_{63}|S_{62} = s_{62}) \sum_{a_{63} \in \mathcal{S}} \pi(s_{63}, a_{63}) p(s_{63}, a_{63}, s_{64})
\end{aligned}$$

We know that S_{62} was s_{62}, therefore

$$\begin{aligned}
Pr(S_{64} = s_{64}) \quad &= \quad \sum_{s_{63} \in \mathcal{S}} 1 * Pr(S_{63} = s_{63}|S_{62} = s_{62}) \sum_{a_{63} \in \mathcal{S}} \pi(s_{63}, a_{63}) p(s_{63}, a_{63}, s_{64}) \\
&= \quad \sum_{s_{63} \in \mathcal{S}} \sum_{a_{62} \in \mathcal{A}} \pi(s_{62}, a_{62}) p(s_{62}, a_{62}, s_{63}) \sum_{a_{63} \in \mathcal{S}} \pi(s_{63}, a_{63}) p(s_{63}, a_{63}, s_{64}) \\
&= \quad \sum_{a_{62} \in \mathcal{A}} \pi(s_{62}, a_{62}) \sum_{s_{63} \in \mathcal{S}} p(s_{62}, a_{62}, s_{63}) \sum_{a_{63} \in \mathcal{S}} \pi(s_{63}, a_{63}) p(s_{63}, a_{63}, s_{64})
\end{aligned}$$

- **(Question 2c. 4 Points)** What is the probability that the state at time $t = 2$ is s_2?

  (Solution 2c.)

  $$Pr(S_2 = s_2) \quad = \quad \sum_{s_1 \in \mathcal{S}} Pr(S_1 = s_1) Pr(S_2 = s_2|S_1 = s_1)$$

  From the example problem, we have

  $$Pr(S_1 = s_1) \quad = \quad \sum_{s_0 \in \mathcal{S}} d_0(s_0) \sum_{a_0 \in \mathcal{A}} \pi(s_0, a_0)\, p(s_0, a_0, s_1)$$

  Similarly,

  $$Pr(S_2 = s_2|S_1 = s_1) \quad = \quad \sum_{a_1 \in \mathcal{A}} \pi(s_1, a_1)\, p(s_1, a_1, s_2)$$

  Substituting them in the above equation,

  $$Pr(S_2 = s_2) \quad = \quad \sum_{s_0 \in \mathcal{S}} d_0(s_0) \sum_{a_0 \in \mathcal{A}} \pi(s_0, a_0) \sum_{s_1 \in \mathcal{S}} p(s_0, a_0, s_1) \sum_{a_1 \in \mathcal{A}} \pi(s_1, a_1)\, p(s_1, a_1, s_2)$$

- **(Question 2d. 6 Points)** What is the probability that the state at time $t = 5$ *was* s_5 given that $A_6 = a_6$ and $S_4 = s_4$?

  (Solution 2d.) We will be using the following properties of probabilities,

  $$Pr(a|b, c) = \frac{\Pr(a, b|c)}{\Pr(b|c)} \tag{7}$$

  $$Pr(a, b|c) = \Pr(a|b, c) P(b|c) \tag{8}$$

  $$Pr(a|c, d) = \sum_{b \in \mathcal{B}} \Pr(b|c, d) \Pr(a|c, d, b) \tag{9}$$

$$\begin{aligned}
Pr(S_5 = s_5|S_4 = s_4, A_6 = a_6) &= \frac{Pr(S_5 = s_5, A_6 = a_6|S_4 = s_4)}{Pr(A_6 = a_6|S_4 = s_4)} \\
Pr(S_5 = s_5, A_6 = a_6|S_4 = s_4) &= Pr(A_6 = a_6|S_4 = s_4, S_5 = s_5)Pr(S_5 = s_5|S_4 = s_4) \\
Pr(S_5 = s_5|S_4 = s_4, A_6 = a_6) &= \frac{Pr(A_6 = a_6|S_4 = s_4, S_5 = s_5)Pr(S_5 = s_5|S_4 = s_4)}{Pr(A_6 = a_6|S_4 = s_4)}
\end{aligned}$$

We have 3 terms to evaluate
Term 1,

$$\begin{aligned}
Pr(A_6 = a_6|S_4 = s_4) &= \sum_{s_6 \in \mathcal{S}} Pr(S_6 = s_6|S_4 = s_4)Pr(A_6 = a_6|S_6 = s_6, S_4 = s_4) \\
Pr(A_6 = a_6|S_6 = s_6, S_4 = s_4) &= Pr(A_6 = a_6|S_6 = s_6) = \pi(s_6, a_6) \\
Pr(S_6 = s_6|S_4 = s_4) &= \sum_{s_5 \in \mathcal{S}} Pr(S_5 = s_5|S_4 = s_4)Pr(S_6 = s_6|S_5 = s_5, S_4 = s_4) \\
Pr(S_6 = s_6|S_5 = s_5, S_4 = s_4) &= \sum_{a_5 \in \mathcal{A}} Pr(A_5 = a_5|S_5 = s_5, S_4 = s_4)Pr(S_6 = s_6|S_5 = s_5, A_5 = a_5, S_4 = s_4) \\
&= \sum_{a_5 \in \mathcal{A}} Pr(A_5 = a_5|S_5 = s_5)Pr(S_6 = s_6|S_5 = s_5, A_5 = a_5) \\
&= \sum_{a_5 \in \mathcal{A}} \pi(s_5, a_5)p(s_5, a_5, s_6)
\end{aligned}$$

Using Example problem, Term2

$$Pr(S_5 = s_5|S_4 = s_4) = \sum_{a_4 \in \mathcal{A}} \pi(s_4, a_4)p(s_4, a_4, s_5)$$

Therefore Term 1 becomes,

$$Pr(A_6 = a_6|S_4 = s_4) = \sum_{s_6 \in \mathcal{S}} \sum_{s_5 \in \mathcal{S}} \sum_{a_4 \in \mathcal{A}} \pi(s_4, a_4)p(s_4, a_4, s_5) \sum_{a_5 \in \mathcal{A}} \pi(s_5, a_5)p(s_5, a_5, s_6)\pi(s_6, a_6)$$

Term 3,

$$\begin{aligned}
Pr(A_6 = a_6|S_4 = s_4, S_5 = s_5) &= \sum_{s_6 \in \mathcal{S}} Pr(S_6 = s_6|S_4 = s_4, S_5 = s_5)Pr(A_6 = a_6|S_4 = s_4, S_5 = s_5, S_6 = s6) \\
&= \sum_{s_6 \in \mathcal{S}} Pr(S_6 = s_6|S_4 = s_4, S_5 = s_5)Pr(A_6 = a_6|S_6 = s6) \\
&= \sum_{s_6 \in \mathcal{S}} \sum_{a_5 \in \mathcal{A}} \pi(s_5, a_5)p(s_5, a_5, s_6)\pi(s_6, a_6)
\end{aligned}$$

Putting them all together gives

$$\begin{aligned}
Pr(S_5 = s_5|S_4 = s_4, A_6 = a_6) &= \frac{(\sum_{s_6 \in \mathcal{S}} \sum_{a_5 \in \mathcal{A}} \pi(s_5, a_5)p(s_5, a_5, s_6)\pi(s_6, a_6))(\sum_{a_4 \in \mathcal{A}} \pi(s_4, a_4)p(s_4, a_4, s_5))}{\sum_{s_6 \in \mathcal{S}} \sum_{s'_5 \in \mathcal{S}} \sum_{a_4 \in \mathcal{A}} \pi(s_4, a_4)p(s_4, a_4, s'_5) \sum_{a_5 \in \mathcal{A}} \pi(s'_5, a_5)p(s'_5, a_5, s_6)\pi(s_6, a_6)} \\
&= \frac{\sum_{a_4 \in \mathcal{A}} \pi(s_4, a_4)p(s_4, a_4, s_5) \sum_{a_5 \in \mathcal{A}} \pi(s_5, a_5) \sum_{s_6 \in \mathcal{S}} p(s_5, a_5, s_6)\pi(s_6, a_6)}{\sum_{a_4 \in \mathcal{A}} \pi(s_4, a_4) \sum_{s'_5 \in \mathcal{S}} p(s_4, a_4, s'_5) \sum_{a_5 \in \mathcal{A}} \pi(s'_5, a_5) \sum_{s_6 \in \mathcal{S}} p(s'_5, a_5, s_6)\pi(s_6, a_6)}
\end{aligned}$$

3. (**7 Points**) In class we discussed how reward functions can be used to specify what is the “goal” (or objective) of the agent. We presented three ways in which the reward function can be specified: some are extremely general, but not necessarily easy to define in practice; and some are less general but can be defined more intuitively in real-world problems:

- The most general formulation of the reward function is given by d_R, which specifies an arbitrary distribution over rewards given that the agent is in some state s, executes an action a, and transitions to some state s'.
- Alternatively, in some problems the reward function can be defined in a way that it (deterministically) returns a scalar number based on s, a, and s'. That is, R can be defined as a function of the form $R(s, a, s')$.
- Finally, an even simpler formulation of reward functions can be constructed that depends only on the state of the agent (s) and the action that it executed (a). That is, R can be defined as a function of the form $R(s, a)$.

Which form of the reward function should be used in practice depends on the particular learning problem or application at hand. In certain problems, for instance, rewards may depend on the state to which the agent transitioned (s') after executing an action (a); in this case, using $R(s, a, s')$ may be more convenient. Importantly, all of these definition are closely related, in the sense that we can, for example, write $R(s, a)$ in terms of d_R, and $R(s, a)$ in terms of $R(s, a, s')$.

- **(Question 3a. 4 Points)** First, show from "first principles", *step by step*, how to derive an equation for $R(s, a)$ in terms of d_R. Recall that, by definition, $R(s, a) = \mathbb{E}[R_t|S_t = s, A_t = a]$.

  (Solution 3a.) By definition, $R(s, a) = \mathbb{E}[R_t|S_t = s, A_t = a]$ and by first principles we know

  $$\mathbb{E}[X \mid A = a] = \sum_{x \in \mathcal{X}} x \operatorname{Pr}(X = x \mid A = a)$$

  and,

  $$Pr(A|B) = \sum_{x} Pr(X = x|B)Pr(A|B, X = x)$$

  This gives

  $$\begin{aligned} \mathbb{E}[R_t|S_t = s, A_t = a] &= \sum_{r \in \mathcal{R}} rPr(R_t = r|S_t = s, A_t = a) \\ &= \sum_{r \in \mathcal{R}} r \sum_{s' \in \mathcal{S}} Pr(S_{t+1} = s'|S_t = s, A_t = a)Pr(R_t = r|S_t = s, A_t = a, S_{t+1} = s') \\ &= \sum_{s' \in \mathcal{S}} Pr(S_{t+1} = s'|S_t = s, A_t = a) \sum_{r \in \mathcal{R}} rPr(R_t = r|S_t = s, A_t = a, S_{t+1} = s') \end{aligned}$$

  Since, $d_R = Pr(R_t = r|S_t = s, A_t = a, S_{t+1} = s')$ and $p(s, a, s') = Pr(S_{t+1} = s'|S_t = s, A_t = a)$,

  $$R(s, a) = \mathbb{E}[R_t|S_t = s, A_t = a] = \sum_{s' \in \mathcal{S}} p(s, a, s') \sum_{r \in \mathcal{R}} r d_R$$

- **(Question 3b. 3 Points)** Next, show (again, step by step) how to construct an equation for $R(s, a)$ in terms of $R(s, a, s')$.

  (Solution 3a.) By definition $R(s, a, s') = \mathbb{E}[R_t|S_t = s, A_t = a, S_{t+1} = s']$, by first principles we know

  $$\mathbb{E}[X \mid A = a] = \sum_{x \in \mathcal{X}} x \operatorname{Pr}(X = x \mid A = a)$$

  Therefore,

  $$R(s, a, s') = \sum_{r \in \mathcal{R}} rPr(R_t = r|S_t = s, A_t = a, S_{t+1} = s')$$

  And from previous section we have,

  $$\begin{aligned} R(s, a) &= \sum_{s' \in \mathcal{S}} Pr(S_{t+1} = s'|S_t = s, A_t = a) \sum_{r \in \mathcal{R}} rPr(R_t = r|S_t = s, A_t = a, S_{t+1} = s') \\ &= \sum_{s' \in \mathcal{S}} p(s, a, s')R(s, a, s') \end{aligned}$$

4. (**4 Points**) Suppose you wish to identify an optimal *deterministic* policy for a given MDP with 3 actions and $|\mathcal{S}|$ states. One way to identify the optimal policy is by performing brute force search; that is, by evaluating $J(\pi)$ for *all* possible deterministic policies π and returning the best one; i.e., the one with the highest expected return/performance. Assume you are given a black-box software that can evaluate *any* policy defined over this MDP in 5 seconds. Assume you have a total time of two hours to identify the optimal policy. Under this time constraint, and assuming you will be using brute force search to identify the optimal policy, what is the maximum number of states, $|\mathcal{S}|$, that the MDP may have? Show, formally and step by step, how you arrived at your solution.

(Solution 4.) Number of policies that can be evaluated in 2 hours,

$$Num = \frac{2 * 60 * 60}{5}$$

And if there are S states and A actions, the number of policies that can be present is

$$Num = |A|^{|S|}$$

Which gives,

$$\begin{aligned}
|A|^{|S|} &<= \frac{2 * 60 * 60}{5} \\
3^{|S|} &<= \frac{2 * 60 * 60}{5} \\
|S| &<= \frac{log(\frac{2*60*60}{5})}{log(3)} \\
|S| &<= 6.6196 \\
|S|_{max} &= 6
\end{aligned}$$

5. (**4 Points**) In class, we presented one particular MDP with finite state space and continuous actions, and showed that it *did not* have an optimal policy. Consider an MDP with $|\mathcal{S}| = \infty$ and $|\mathcal{A}| < \infty$. Assume $\gamma < 1$. *Formally* define an MDP with these properties (i.e., formally specify all elements of the tuple $(\mathcal{S}, \mathcal{A}, p, R, d_0, \gamma)$, where you can assume that $R : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$) for which an optimal policy, π^*, *does* exist. Describe such an optimal policy in English and present/describe its performance, $J(\pi^*)$.

(Solution 5.) Consider the diagram in figure 1, where the agent can be spawned anywhere on the horizontal line, at the objective is to reach the flag. Lets define the MDP with the following elements: **State:** $\mathcal{S}-$ Position on the ground. $|S| = \infty$. **Action:** $\mathcal{A} \in$ [Move right 1 unit, Move left 1 unit, Do nothing]. Since there are only 3 actions that can be performed, $|A| < \infty$. **transition function:** $p(s, a, s') \in [0, 1]$. **Reward:** $R_t = -d$ where $d, d \in \mathcal{Z}$ is the distance from the pole. **Initial state:** $d_0 = Pr(S_0 = -d) = 1$ for every episode. $\gamma < 1$. S_∞ is the pole, after which the episode ends and the agent dies.

Since the reward the the agent is getting is negative of the distance from the pole, so for an optimal policy it is going to reduce the distance between itself and the pole by taking an action that is towards the direction of the pole. For Example, if we take the current position of the agent as shown in figure 1, it should take the action "Move right 1 unit". Here $S_0 = -d$, therefore if it takes the action a_0 = "Move right 1 unit", it will go to the state $S_1 = -d + 1$ with a probability of $p(s_0, a_0, s_1)$, and the reward it will get will be $R_0 = -d$. The total reward for the optimal policy would be

$$\begin{aligned}
J(\pi^*) &= \sum_t \gamma^t R_t \\
&= R_0 + \gamma R_1 + \gamma^2 R_2 .... \\
&= (-d) + \gamma(-d + 1) + \gamma^2(-d + 2) ...
\end{aligned}$$

6. (**5 Points**) The objective of standard RL algorithms is to find policies that maximize expected return. However, this objective does not take into account the possible *risks* of deploying such policies. Consider, for example, an optimal policy (π_1) that, when executed by the agent, results in a return of +20 with 50% probability, and a

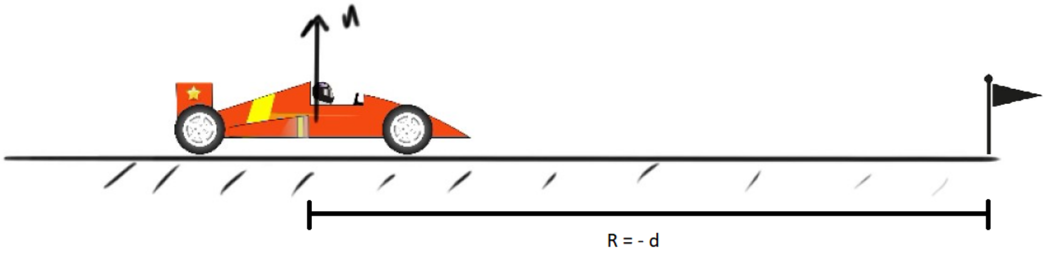

Figure 1: Move towards the flag

return of -12 with 50% probability. Consider an alternative optimal policy (π_2) that, when executed, produces a return of $+4$ deterministically. The expected return of both policies is the same. One could argue, however, that even though both policies have the same average performance, an agent could prefer policy π_2 since deploying it would never result in a (possibly catastrophic) low performance—in this particular case, a return of -12. RL algorithms have been proposed that identify policies that perform well (i.e., policies with high expected return) but whose *return variance* is small. Intuitively, these algorithms identify policies that, when deployed, produce returns that are both high *and* predictable/reliable. Recall that we defined return as the discounted sum of rewards: $\sum_{t=0}^{\infty} \gamma^t R_t$. Let an MDP exist such that it has two optimal policies. Can the expected value of their returns differ? If not, explain why; if so, give an example. Furthermore, can the variance of their returns differ? **If so, give an example by presenting an *infinite horizon* MDP with two optimal policies whose expected returns are equal, but such that the variance of their corresponding returns differs.**

Solution 6. A policy is considered to be optimal if it gives the maximum expected return. If there are two optimal policies, it would mean that they would be giving maximum expected return, and if both of them are giving the maximum expected return, then both the return are the same. Therefore the Expected value of the returns would be the same. Yes the variance of two optimal policies can be different. To explain this lets consider two policies with the same expected return. Let's consider an advertisement recommendation system, this is an infinite horizon policy since the recommender will have to recommend ads forever to the user. There are 3 categories from which it has to recommend, so it needs to decide which category to show, Action = [Cat1, Cat2, Cat3]. Policy1 - returns 0.25 with $\frac{1}{3}$ probability, 0 with $\frac{1}{3}$ probability and -0.25 with $\frac{1}{3}$ probability. It has an expected return of 0 and a variance of 0.0625. Policy2 - returns 15 with $\frac{1}{3}$ probability, 5 with $\frac{1}{3}$ probability and -20 with $\frac{1}{3}$ probability. This has an expected return of 0 and a variance of 325. Therefore we can see that although both these policies return the same expected return value, but Policy2 has a higher variance.

7. (**3 Points**) To fully specify an MDP we have to define $\mathcal{S}$, $\mathcal{A}$, p, R, d_0, and γ. In many real-world applications, however, it may not be possible (or it might be incredibly challenging) to *specify an explicit/analytic* equation for R. Give an example of a real-world problem that can be modeled as an MDP but where R is not known *a priori* by the agent (or cannot be easily specified analytically) and explain why that is the case. Also, describe one possible way by which the agent could incrementally learn/estimate R based on its interactions with the environment.

   Solution 7. Let's consider an example where a sprinkler system is installed in a garden, and it needs to sprinkle water in its surroundings based on the surrounding weather conditions. The task is to keep the grass alive. State is the current weather conditions and the span of land in which the water has to be sprinkled. Action is to decide the amount of water that will be sprinkled. This is an infinite horizon MDP, since the sprinkler will have to do this everyday. It's difficult to design a reward function in this case since the life of grass will not be defined exactly on the basis of one day's amount of water sprinkled. It will be a cumulative result of many days of sprinkling. It is possible that after sprinkling very little amount of water one day doesn't kill the grass instantly, but if the sprinkler continues to spread less water, the grass will die immediately. The grass could die even if the sprinkler decides to pour large amounts of water. Therefore deciding a reward function which is considered optimum for keeping the grass alive forever is difficult.

8. (**3 Points**) Similarly to the question above, describe a real-world problem that can be reasonably modeled as an MDP, but such that its transition function, p, is *not* be known *a priori* by the agent or cannot be easily

specified analytically. Explain why that is the case and propose one possible way by which the agent could estimate it based on its interactions with the environment.

Solution 8. Consider an agent that is sitting in a boat and it is paddling. The boat is floating in a turbulent water body. And the task is to go from 1 end to the other end of the water body. The actions that the agent can perform is [steer right, steer left, steer front, steer back, etc]. Although the actions are know, it is difficult to model the effect of turbulence on the boat and the agent. And it is quite possible that external factors and environmental conditions like wind, rain, stones, tree logs, etc might come in the path of the agent or could affect the direction and distance the agent is travelling to. And these external factors could change with time. Therefore it would be difficult to model these factors as states analytically in the MDP. Although the agent can learn how to steer when encountered with any of these external factors by taking an action that nullifies the effect of these factors. We can also consider a case where an agent is trained to run on a synthetic track and then the agent is transferred to a concrete track, the agent wouldn't know what the conditions of friction would be in the second case. Therefore the transition function wouldn't be known here.

9. (**6 Points**) Consider an MDP $(\mathcal{S}, \mathcal{A}, p, R, d_0, \gamma)$, where we know that the rewards produced by R are bounded between $-R_{\text{min}}$ and R_{max}. Assume that someone gives you a policy, π, whose return is $J(\pi) = \dfrac{R_{\text{max}}}{1-\gamma}$. What can you say about the suboptimality of this policy? Is there a way of bounding how much worse $J(\pi)$ may be when compared to the expected return of an optimal policy, $J(\pi^*)$? Can an alternative policy, π_{new}, be constructed that has higher return than π? If so, explain intuitively how; if not, *prove* why this cannot be done.

(Solution 9.) Total reward for a policy can be given as

$$\begin{aligned}
J(\pi) &= \sum_t \gamma^t R_t \\
&= R_0 + \gamma R_1 + \gamma^2 R_2 + \gamma^3 R_3 ... \\
J_{max}(\pi) &= R_{max} + \gamma R_{max} + \gamma^2 R_{max} + \gamma^3 R_{max} ... \\
&= R_{max}(1 + \gamma + \gamma^2 + \gamma^3 ...) \\
&= R_{max}\frac{1}{1-\gamma}
\end{aligned}$$

We can see here that

$$J(\pi) = J_{max}(\pi)$$

which is equal to the maximum reward that can be obtained while executing a policy. Which means that the current given policy is the **optimal policy**.

Since $J(\pi)$ is the optimal policy the **lower bound** from the optimal policy $\pi*$ would be **0** and the **upper bound** would be **0**.

But if we want to calculate the bounds of any general policy, then we need to calculate the lowest reward that can be obtained using the worst policy. The worst performing policy is the one which gets the minimum reward every time. From the above derivation it follows,

$$\begin{aligned}
J_{min}(\pi) &= -R_{min}\frac{1}{1-\gamma} \\
&= \frac{-R_{min}}{1-\gamma}
\end{aligned}$$

Therefore the upper bound of the worst performance of a policy with respect to the optimal policy is

$$\begin{aligned}
|J(\pi^*) - J(\pi)| &<= \frac{R_{max}}{1-\gamma} - J_{min}(\pi) \\
&<= \frac{R_{max}}{1-\gamma} - \frac{-R_{min}}{1-\gamma} \\
&<= \frac{R_{max} + R_{min}}{1-\gamma}
\end{aligned}$$

Therefore the worst performance of any generic π with respect to the optimal policy π^* is bounded in $\left[0, \frac{R_{max}+R_{min}}{1-\gamma}\right]$.

Since we saw above that the maximum return for any policy in this MDP is $J_{max} = \frac{R_{max}}{1-\gamma}$ which is equal to the return that we get from the given π, therefore there is no other π_{new} that can give a reward more than $\frac{R_{max}}{1-\gamma}$. If there is a π_{new} such that it gives

$$\begin{aligned} J(\pi_{new}) &> \frac{R_{max}}{1-\gamma} \\ J(\pi_{new}) &> J_{max} \end{aligned}$$

Which means,

$$\begin{aligned} J_{max} &= J(\pi_{new}) \\ J(\pi_{new}) &= \frac{R_{max}}{1-\gamma} \end{aligned}$$

Therefore, there is no π_{new} that gives a reqrd more than π

10. (**5 Points**) Consider the Dinosaur Game—a browser game developed by Google and built into the Google Chrome web browser. The player controls a pixelated Tyrannosaurus rex across a side-scrolling landscape, avoiding obstacles to achieve high scores. For a demo of this game, please see this video. Assume we wish to model this game as an MDP so that an RL agent can be trained to optimally control the dinosaur. Assume that the state includes information about the current velocity of the dinosaur and its distance to the next obstacle. Assume there are only two actions: Jump or Do_Nothing. Suppose that $\gamma = 1$ and that the MDP is finite-horizon; in particular, the game either terminates after 100,000 steps or if the agent collides with an obstacle (in which case the episode ends). The reward function returns +1 after each action that keeps the dinosaur alive, and 0 otherwise. To make the game more challenging, its developers implemented a rule that changes the speed with which the dinosaur moves: its speed increases automatically after every 100 steps. This means that, e.g., at time $t = 23$, the dinosaur might be moving 5 meters ahead after every action; at time $t = 140$, however, the dinosaur might be moving 7.5 meters ahead after every action. Using the formal definition of stationary and non-stationary MDPs, introduced in class, show formally that this is a non-stationary MDP. Furthermore, describe precisely how you would change the definition of this MDP (i.e., the definition of one or more components of $(\mathcal{S}, \mathcal{A}, p, R, d_0, \gamma)$) so that the resulting MDP does model the game as describe above, and so that it is stationary. Use the formal definitions of stationarity and non-stationarity to formally show that this new MDP is, indeed, stationary.

Solution 10. S = (v, d_{obst})
A = [Jump, DoNothing]
R = +1 every second it is alive
$d_0 = P(S_0 = s_0)$ is the same for all episodes
$\gamma = 1$

We know that the speed of the dinosaur changes after every 100 time steps, but this information is not included in the state. Which would mean that for t = 1 and i = 99,

$$P(S_{t+1} = s'|S_t = s, A_t = a) \neq P(S_{i+1} = s'|S_i = s, A_i = a)$$

$$P(v_{t+1}, d_{obst,t+1}|v_t, d_{obst,t}, a) \neq P(v_{i+1}, d_{obst,i+1}|v_i, d_{obst,i}, a)$$

since at time=100, the velocity changes suddenly. Since the velocity is different at t+1 and i+1, the probability of being in those states $P(v_{t+1}, d_{obst,t+1}|v_t, d_{obst,t}, a)$ and $P(v_{i+1}, d_{obst,i+1}|v_i, d_{obst,i}, a)$ will be different. If we include the timestamp/time-step in the state and re-write velocity in terms of the time-step (as a step function of the time V(t)), then the state would know the velocity at t+1 and i+1. And this would hold true for any value of t and i. Therefore the probability values of the transition from one t to t+1 and i to i+1 the same, because then we would know the probability distribution of the transition. Now the MDP is stationary.

$$P(v(t)_{t+1}, t+1, d_{obst,t+1}|v(t)_t, t, d_{obst,t}, a) = P(v(i)_{i+1}, i+1, d_{obst,i+1}|v(i)_i, i, d_{obst,i}, a)$$

Part Two: Programming (45 Points Total)

Consider the following MDP:

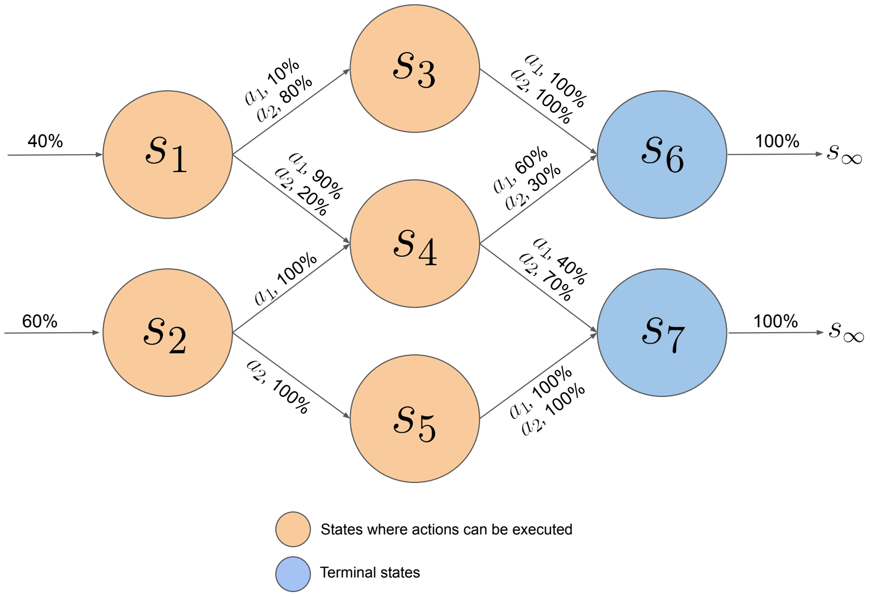

Figure 2: An MDP.

Assume the following initial state distribution, d_0, and transition function, p, where all transition probabilities not indicated in the table below are 0.

$d_0(s_1) = 0.4$	$d_0(s_2) = 0.6$
$p(s_1, a_1, s_3) = 0.1$	$p(s_1, a_1, s_4) = 0.9$
$p(s_1, a_2, s_3) = 0.8$	$p(s_1, a_2, s_4) = 0.2$
$p(s_2, a_1, s_4) = 1.0$	$p(s_2, a_2, s_5) = 1.0$
$p(s_3, a_1, s_6) = 1.0$	$p(s_3, a_2, s_6) = 1.0$
$p(s_4, a_1, s_6) = 0.6$	$p(s_4, a_1, s_7) = 0.4$
$p(s_4, a_2, s_6) = 0.3$	$p(s_4, a_2, s_7) = 0.7$
$p(s_5, a_1, s_7) = 1.0$	$p(s_5, a_2, s_7) = 1.0$

Assume the following reward function:

$R(s_1, a_1) = 5$	$R(s_1, a_2) = 2$
$R(s_2, a_1) = -3$	$R(s_2, a_2) = 7$
$R(s_3, a_1) = 3$	$R(s_3, a_2) = -5$
$R(s_4, a_1) = -6$	$R(s_4, a_2) = 8$
$R(s_5, a_1) = 4$	$R(s_5, a_2) = 10$

Finally, consider the following stochastic policy, π:

$\pi(s_1, a_1) = 0.4$	$\pi(s_1, a_2) = 0.6$
$\pi(s_2, a_1) = 0.35$	$\pi(s_2, a_2) = 0.65$
$\pi(s_3, a_1) = 0.9$	$\pi(s_3, a_2) = 0.1$
$\pi(s_4, a_1) = 0.5$	$\pi(s_4, a_2) = 0.5$
$\pi(s_5, a_1) = 0.1$	$\pi(s_5, a_2) = 0.9$

(Question 1. 15 Points) Find an analytic, closed-form expression for

$$J(\pi) \quad = \quad \mathbb{E}\left[\sum_{t=0}^{1} \gamma^t R_t\right]$$

as a function of γ. To do this, you *may* choose do it from "first principles", by repeatedly using the properties of probability distributions and expected values introduced in Section 2. If you do not wish to derive a closed-form expression for $J(\pi)$ this way, you are also allowed to write it directly as a function of d_0, p, π, and R, similarly to the final equation described in the "**Example problem**" introduced in Section 2.

Your final answer should be in the form of $J(\pi) = c_1 + \gamma c_2$, where each c_i is a real-valued constant. Hint: start by applying the property of linearity of expectation to the definition of $J(\pi)$ and then derive separate equations for each of the resulting terms.

(Solution 1.)

$$\begin{aligned} J(\pi) &= \mathbb{E}\left[\sum_{t=0}^{1} \gamma^t R_t\right] \\ &= \mathbb{E}\left[R_0 + \gamma R_1\right] \\ &= \mathbb{E}\left[R_0\right] + \gamma \mathbb{E}\left[R_1\right] \end{aligned}$$

We need to get the expected value of R_0 and R_1. For this we would require the probabilities of being in states $S_1, S_2, S_3, S_4 \& S_5$. And we will use that in the following equations.

E[Reward] = Pr(being in state s) (Reward that you get when you take action a) Pr(of taking action a when in state s)

$$\begin{aligned} \mathbb{E}\left[R_0\right] &= \sum_{x=0}^{1}\sum_{y=0}^{1} Pr(S_0 = s_x) R_0(s_x, a_y) \pi(s_x, a_y) \\ \mathbb{E}\left[R_1\right] &= \sum_{x=3}^{5}\sum_{y=0}^{1} Pr(S_1 = s_x) R_1(s_x, a_y) \pi(s_x, a_y) \end{aligned}$$

$$\begin{aligned} Pr(S_0 = s_1) &= d_0(s_1) = 0.4 \\ Pr(S_0 = s_2) &= d_0(s_2) = 0.6 \end{aligned}$$

From the example problem

$$\begin{aligned}
Pr(S_1 = s) &= \sum_{s_0} Pr(S_0 = s_0) Pr(S_1 = s | S_0 = s_0) \\
&= \sum_{s_0 \in \mathcal{S}} d_0(s_0) \sum_{a_0 \in \mathcal{A}} \pi(s_0, a_0)\, p(s_0, a_0, s). \\
Pr(S_1 = s_3) &= Pr(S_0 = s_1) Pr(S_1 = s_3 | S_0 = s_1) \\
&= d_0(s_1) Pr(S_1 = s_3 | S_0 = s_1) \\
&= d_0(s_1) \sum_{a} Pr(A_0 = a | S_0 = s_1) Pr(S_1 = s_3 | S_0 = s_1, A_0 = a) \\
&= d_0(s_1)[Pr(A_0 = a_1 | S_0 = s_1) Pr(S_1 = s_3 | S_0 = s_1, A_0 = a_1) + \\
&\quad Pr(A_0 = a_2 | S_0 = s_1) Pr(S_1 = s_3 | S_0 = s_1, A_0 = a_2)] \\
&= d_0(s_1)[\pi(s_1, a_1) p(s_1, a_1, s_3) + \pi(s_1, a_2) p(s_1, a_2, s_3)]
\end{aligned}$$

Similarly for s_4, and s_5

$$\begin{aligned}
Pr(S_1 = s_4) &= \sum_{s_0 \in \mathcal{S}} d_0(s_0) \sum_{a_0 \in \mathcal{A}} \pi(s_0, a_0)\, p(s_0, a_0, s_4) \\
&= d_0(s_1)[\pi(s_1, a_1) p(s_1, a_1, s_4) + \pi(s_1, a_2) p(s_1, a_2, s_4)] + \\
&\quad d_0(s_2)[\pi(s_2, a_1) p(s_2, a_1, s_4) + \pi(s_2, a_2) p(s_2, a_2, s_4)] \\
Pr(S_1 = s_5) &= \sum_{s_0 \in \mathcal{S}} d_0(s_0) \sum_{a_0 \in \mathcal{A}} \pi(s_0, a_0)\, p(s_0, a_0, s_5) \\
&= d_0(s_1)[\pi(s_1, a_1) p(s_1, a_1, s_5) + \pi(s_1, a_2) p(s_1, a_2, s_5)] + \\
&\quad d_0(s_2)[\pi(s_2, a_1) p(s_2, a_1, s_5) + \pi(s_2, a_2) p(s_2, a_2, s_5)]
\end{aligned}$$

Substituting values from the tables above

$$\begin{aligned}
Pr(S_1 = s_3) &= 0.4 * [0.4 * 0.1 + 0.6 * 0.8] = 0.208 \\
Pr(S_1 = s_4) &= 0.4 * [0.4 * 0.9 + 0.6 * 0.2] + 0.6 * [0.35 * 1.0 + 0.65 * 0] = 0.402 \\
Pr(S_1 = s_5) &= 0.4 * [0.4 * 0 + 0.2 * 0] + 0.6 * [0.35 * 0 + 0.65 * 1.0] = 0.39
\end{aligned}$$

Now substituting these value to get the expected value for R_0 and R_1

$$\begin{aligned}
\mathbb{E}[R_0] &= \sum_{x=0}^{1} \sum_{y=0}^{1} Pr(S_0 = s_x) R_0(s_x, a_y) \pi(s_x, a_y) \\
&= Pr(S_0 = s_1) R_0(s_1, a_1) \pi(s_1, a_1) + Pr(S_0 = s_1) R_0(s_1, a_2) \pi(s_1, a_2) + \\
&\quad Pr(S_0 = s_2) R_0(s_2, a_1) \pi(s_2, a_1) + Pr(S_0 = s_2) R_0(s_2, a_2) \pi(s_2, a_2) \\
&= 0.4 * 5 * 0.4 + 0.4 * 2 * 0.6 + 0.6 * -3 * 0.35 + 0.6 * 7 * 0.65 = 3.38 \\
\mathbb{E}[R_1] &= \sum_{x=3}^{5} \sum_{y=0}^{1} Pr(S_1 = s_x) R_1(s_x, a_y) \pi(s_x, a_y) \\
&= Pr(S_1 = s_3) R_1(s_3, a_1) \pi(s_3, a_1) + Pr(S_1 = s_3) R_1(s_3, a_2) \pi(s_3, a_2) + \\
&\quad Pr(S_1 = s_4) R_1(s_4, a_1) \pi(s_4, a_1) + Pr(S_1 = s_4) R_1(s_4, a_2) \pi(s_4, a_2) + \\
&\quad Pr(S_1 = s_5) R_1(s_5, a_1) \pi(s_5, a_1) + Pr(S_1 = s_5) R_1(s_5, a_2) \pi(s_5, a_2) \\
&= 0.208 * 3 * 0.9 + 0.208 * -5 * 0.1 + 0.402 * -6 * 0.5 + 0.402 * 8 * 0.5 + 0.39 * 4 * 0.1 + 0.39 * 10 * 0.9 = 4.5256
\end{aligned}$$

Therefore Now $J(\pi)$,

$$\begin{aligned}
J(\pi) &= \mathbb{E}[R_0] + \gamma \mathbb{E}[R_1] \\
&= 3.38 + \gamma 4.5256
\end{aligned}$$

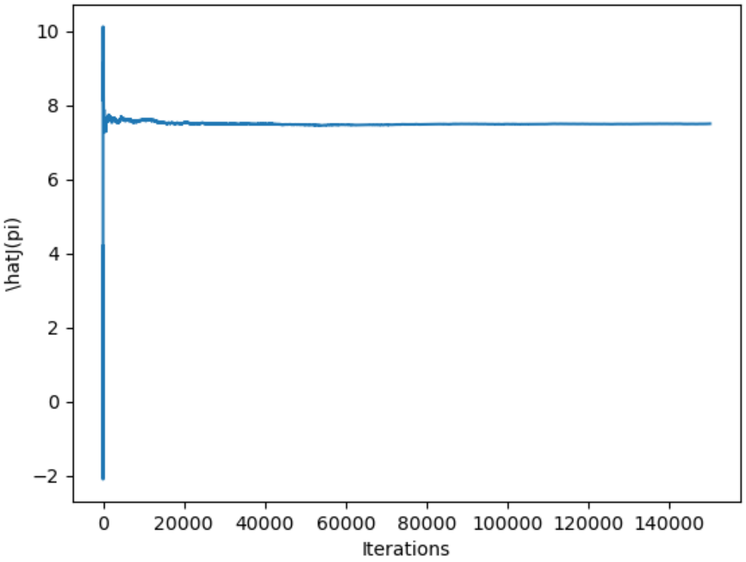

Figure 3: $\hat{J}(\pi)$

(Question 2 - 30 Points) ***Programming Question***

In this question, you will write a program to estimate $J(\pi)$ by simulating many of the possible outcomes (returns) that might result from running π on the previously-defined MDP. Each simulation will produce a particular sequence of states, actions, and rewards, and, thus, a particular discounted return. Since $J(\pi)$ is defined as the *expected* discounted return, you can construct an estimate of $J(\pi)$, $\hat{J}(\pi)$, by averaging the discounted returns observed across N simulations.

In particular:

- To run one simulation (or episode, or trial), you should follow the "Agent-Environment Interaction" procedure introduced in Class #3.
- Start by creating a function called *runEpisode* that takes as input a policy and a value of γ, and that returns the empirical discounted return resulting from that episode.
- Let G^i be the discounted return of the i^{th} episode. You will estimate $J(\pi)$ by computing $\hat{J}(\pi) := \frac{1}{N}\sum_{i=1}^{N} G^i$.

(Question 2a. 8 Points). Construct $\hat{J}(\pi)$ by running 150,000 simulations/episodes. You should then create a graph where the x axis shows the number of episodes, and the y axis presents the estimate $\hat{J}(\pi)$ constructed based on all episodes executed up to that point. That is, the point $x = 100$ in this graph should have as its corresponding y coordinate the estimate $\hat{J}(\pi)$ built using the discount returns from the first 100 simulations. **You should use $\gamma = 0.9$. Using a value $\gamma < 1$ will make it easier for you to debug whether your implementation of $\hat{J}(\pi)$ matches the expected results given by the analytic solution you constructed in part 1 of this question. In case you have already solved this question using a different value of γ, that is ok: you can still submit your results as they are, but please do report which value of γ you chose to use.**

Solution 2a. Using $\gamma = 0.9$ we get the variation of $\hat{J}(\pi)$ as shown in figure 3. Running the simulation 150,000 times we get $\hat{J}(\pi) = 7.45$

(Question 2b. 5 Points). Report the average discounted return, as well as its variance, at the end of this process; that is, report $\hat{J}(\pi)$ after executing 150,000 episodes.

γ	$\hat{J}(\pi)$	$J(\pi)$
0.25	4.5022	4.5114
0.5	5.6324	5.6428
0.75	6.7765	6.7742
0.99	7.867	7.8603

Table 1: Caption

Solution2b.

- Average discounted return Avg(G) = 7.45
- Variance of the discounted return Var(G) = 65.5505
- Variance of the average discounted return Var(J(pi)) = 0.006076

(Question 2c. 5 Points). Estimate $\hat{J}(\pi)$ using different discount rates: $\gamma \in \{0.25, 0.5, 0.75, 0.99\}$. Compare these estimates with their true values, computed according to the closed-form solution for $J(\pi)$ found in the first part of this question. These values should approximately match (i.e., $J(\pi) \approx \hat{J}(\pi)$).

Solution 2c. Table 1 reports the $\hat{J}(\pi)$ and shows the comparison between $J(\pi) \approx \hat{J}(\pi)$

(Question 2d. 12 Points). Next, we will use your *runEpisode* function to estimate the performance of *different* policies (other than the one we introduced/proposed) in order to search for the policy with highest performance. You may use any optimization method you want to implement this step (even, e.g., brute force search). To simplify this process, you *should* restrict your search to deterministic policies. You should use $\gamma = 0.75$. Describe how the policy search method you used works. Report the best policy, $\hat{\pi}^*$, identified by the process above, and present its estimated performance, $\hat{J}(\hat{\pi}^*)$, computed using $N = 350,000$ simulations.

Solution 2d. Since there are 5 states and 2 actions the total number of deterministic policies that can be are $|A|^{|S|} = 2^5 = 32$. I am using a brute force search for finding the best policy. First we create a list of all the possible policies. The we iterate through all of those policies one-by-one and compute the discounted return. The best policy is the one which gives the maximum discounted return. The pseudo code for the search is shown in Algorithm 1.

Algorithm 1 Find best policy

```
policies = $getAllPolicies()$
G = 0
bestJ = $-\infty$
for policy in policies do
    for iterations do
        G += runEpisode($policy, \gamma$)
    end for
    J = G/iterations
    if bestJ < J then
        bestJ = J
        bestPolicy = policy
    end if
end for
```

Best policy $\hat{\pi}^* =$ {'s1': 'a1', 's2': 'a2', 's3': 'a1', 's4': 'a2', 's5': 'a2'}. Here we show the action that needs to be taken at every state as {'state':'action'}
Estimated performance $\hat{J}(\hat{\pi}^*) = 12.94$