دانشگاه صنعتی شریف
دانشکده‌ی مهندسی کامپیوتر
طراحان: امیدرضا معصومی، مبین باقری، محمد شفیع‌زاده، دنیا جعفری، سپهر ذوالفقاری، علیرضا ملک حسینی

هوش مصنوعی

بهار۱۴۰۴
استاد: احسان تن قطاری

تمرین پنجم | **یادگیری تقویتی** | مهلت ارسال: ۳۱مرداد

- مهلت ارسال پاسخ تا ساعت ۲۳:۵۹ روز مشخص‌شده است.
- در طول ترم امکان ارسال با تاخیر پاسخ همه‌ی تمارین سقف ۴ روز و در مجموع ۱۰ روز، وجود دارد. پس از گذشت این مدت، پاسخ‌های ارسال‌شده پذیرفته نخواهند بود. همچنین، به ازای هر ساعت تأخیر غیر مجاز نیم درصد از نمره ی تمرین کم خواهد شد.
- هم‌کاری و هم‌فکری شما در انجام تمرین مانعی ندارد اما پاسخ ارسالی هر کس حتما باید توسط خود او نوشته شده باشد.
- در صورت هم‌فکری و یا استفاده از هر منابع خارج درسی، نام هم‌فکران و آدرس منابع مورد استفاده برای حل سوال مورد نظر را ذکر کنید.
- لطفا تصویری واضح از پاسخ سوالات نظری بارگذاری کنید. در غیر این صورت پاسخ شما تصحیح نخواهد شد.

- مهلت ارسال پاسخ تا ساعت ۲۳:۵۹ روز مشخص‌شده است.
- در طول ترم امکان ارسال با تاخیر پاسخ همه‌ی تمارین سقف ۴ روز و در مجموع ۱۰ روز، وجود دارد. پس از گذشت این مدت، پاسخ‌های ارسال‌شده پذیرفته نخواهند بود. همچنین، به ازای هر ساعت تأخیر غیر مجاز نیم درصد از نمره ی تمرین کم خواهد شد.
- هم‌کاری و هم‌فکری شما در انجام تمرین مانعی ندارد اما پاسخ ارسالی هر کس حتما باید توسط خود او نوشته شده باشد.
- در صورت هم‌فکری و یا استفاده از هر منابع خارج درسی، نام هم‌فکران و آدرس منابع مورد استفاده برای حل سوال مورد نظر را ذکر کنید.
- لطفا تصویری واضح از پاسخ سوالات نظری بارگذاری کنید. در غیر این صورت پاسخ شما تصحیح نخواهد شد.

سوالات (۱۰۰ نمره)

۱. (نمره) درستی یا نادرستی عبارت‌های زیر را با ذکر دلیل مشخص کنید.

(آ) اگر ضریب تخفیف γ شرط $۱ > \gamma > ۰$ را ارضا کند، می‌توان تضمین کرد که روش value iteration همگرا می‌شود

(ب) اگر برتر بودن یک سیاست را به معنی این در نظر بگیریم که موجب به دست آوردن reward بیشتری می‌شود، سیاست‌هایی که توسط روش value iteration به‌دست می‌آیند، از سیاست‌هایی که توسط روش policy iteration به‌دست می‌آیند، برتر هستند.

(ج) Q-learning می‌تواند تابع بهینه Q^* را بدون اینکه حتی یک بار سیاست بهینه را اجرا کند بیاموزد.

(د) اگر یک MDP مدل انتقال T ای داشته باشد که برای همهٔ سه‌تایی‌های $T(s, a, s')$ احتمالِ غیرصفر اختصاص دهد، آنگاهQ-learning شکست خواهد خورد.

(ه) فرض کنید عامل ۱ تابع مطلوبیت $U_۱$ و عامل ۲ تابع مطلوبیت $U_۲$ را دارد. اگر

$$U_۱ = k_۱ U_۲ + k_۲$$

که در آن $k_۱ > ۰$ و $k_۲ > ۰$ باشد، آنگاه عامل ۱ و عامل ۲ ترجیحات یکسانی دارند.

(و) به این سوال پاسخ کوتاه دهید. یک تغییر متداول در یادگیری Q، گنجاندن پاداش‌ها از گام‌های زمانی بیشتر در عبارت X است. بنابراین، عبارت معمول ما

$$r_t + \gamma \cdot \max_{a_{t+1}} q(s_{t+1}, a_{t+1}; w)$$

به این شکل تبدیل می‌شود:

$$r_t + \gamma \cdot r_{t+1} + \gamma^2 \cdot \max_{a_{t+2}} q(s_{t+2}, a_{t+2}; w)$$

به نظر شما مزایای استفاده از پاداش‌های بیشتر در این تخمین چیست؟

حل.

۱) **درست** ـ برای $0 < \gamma < 1$، عملگرِ بلمنِ بهینگی یک انقباض با ضریب γ در نورم ℓ_∞ است؛ لذا طبق قضیهٔ باناخ، Value Iteration به V همگرا می‌شود.

۲) **نادرست** ـ اجرای کامل هر دو روش به π منجر می‌شود؛ بنابراین سیاست‌های خروجی یکی ذاتاً «برتر» از دیگری نیستند. Policy Iteration به‌صورت یکنواخت سیاست را بهبود می‌دهد، ولی Value Iteration چنین تضمینی برای سیاست‌های میانی ندارد.

۳) **درست** ـ Q-learning یک روش off-policy است و بدون اجرای سیاست بهینه، با قاعدهٔ

$$Q(s,a) \leftarrow Q(s,a) + \alpha\big[r + \gamma \max_{a'} Q(s',a') - Q(s,a)\big]$$

و با کاوش کافی و نرخ‌های یادگیری مناسب، به Q همگرا می‌شود.

۴) **نادرست** ـ نامنفی و غیرصفر بودن همهٔ $T(s' \mid s,a)$ باعث شکست نمی‌شود؛ Q-learning به T نیاز ندارد و در MDPهای کاملاً تصادفی نیز (تحت شرایط استاندارد) همگرا است.

۵) **درست** ـ دو تابع مطلوبیت زمانی ترجیحات یکسانی را نمایش می‌دهند که یکی از آن‌ها تبدیل مثبتی از دیگری باشد، یعنی:

$$U_1 = k_1 U_2 + k_2 \quad \text{که در آن } k_1 > 0$$

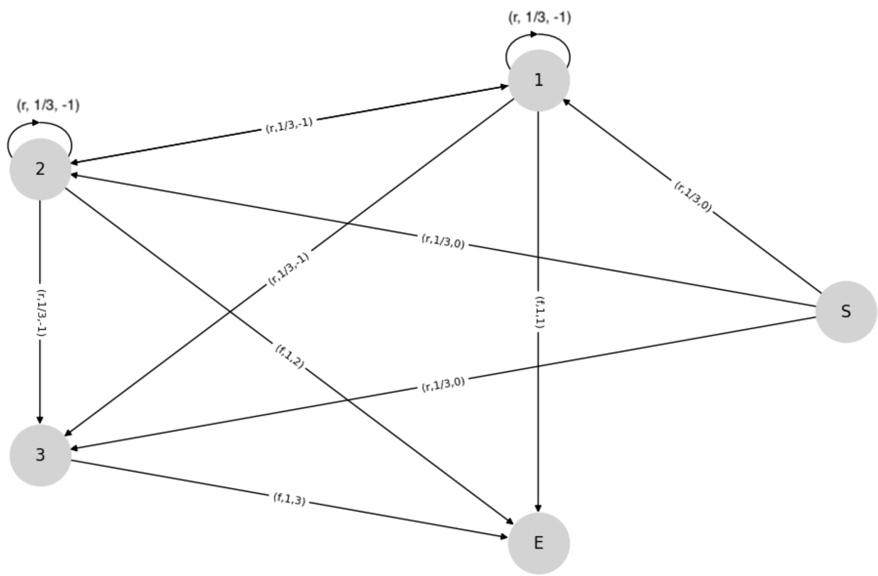
از آنجا که در صورت سوال $k_1 > 0$ و $k_2 \in \mathbb{R}$ داده شده‌اند، تابع U_1 تبدیل آفین مثبتی از U_2 است. بنابراین، عامل ۱ و عامل ۲ ترجیحات یکسانی دارند.

۶) گنجاندن پاداش‌ها از چندین گام زمانی، تخمین «واقع‌بینانه‌تری» از پاداش کل واقعی ارائه می‌دهد، زیرا درصد بیشتری از آن از تجربه واقعی ناشی می‌شود. این روش می‌تواند به پایدارسازی فرآیند آموزش کمک کند، در حالی که همچنان امکان آموزش در هر گام زمانی (بوت‌استرپینگ) را فراهم می‌کند. این نوع روش، «یادگیری تفاضل زمانی N-گامی» نامیده می‌شود.

۲. (نمره) یک بازی جدید در نظر بگیرید که هربار یک عدد رندوم با توزیع احتمال یکنواخت بین ۱ تا ۳ تولید می‌شود. شما هربار می‌توانید به مقدار عدد تولید شده امتیاز بگیرید و بازی تمام شود، یا امتیاز -1 بگیرید و ادامه دهید. $\gamma = 0.9$ در نظر بگیرید.

(آ) برای این بازی یک MDP ارائه دهید و اجزای مختلف آن را مشخص کنید.

(ب) سیاست بهینه‌ای که پس از ۳ مرحله اجرای value iteration بدست می‌آید را حساب کنید.



Iteration	State	Action		Updated Value(max)
		r	f	
۱	۱	-۱	۱	۱
	۲	-۱	۲	۲
	۳	-۱	۳	۳
۲	۱	۰.۸	۱	۱
	۲	۰.۸	۲	۲
	۳	۰.۸	۳	۳
۳	۱	۰.۸	۱	۱
	۲	۰.۸	۲	۲
	۳	۰.۸	۳	۳

(ج) با داشتن سیاست اولیه پایان بازی برای اعداد ۱ و ۲ و ادامه برای عدد ۳، با اجرای ۳ مرحله از policy iteration سیاست بهینه را پیدا کنید.

حل.

(آ) محیط از S شروع می‌شود و با احتمال یکسان به هر سه حالت عدد می‌رود. در هر عدد، می‌توان اکشن f را انجام داد که بازی با همان مقدار پاداش تمام می‌شود. یا می‌توان اکشن r را انجام داد که با احتمال یکسان می‌تواند به هر حالت عدد برود.

(ب) با توجه به رابطه، ارزش هر اکشن را در جدول یادداشت می‌کنیم.

$$v_{k+۱}(s) \leftarrow \max_{a\in\mathcal{A}(s)} \sum_{s'} \mathcal{P}^a_{ss'}\left[\mathcal{R}^a_{ss'} + \gamma v_k(s')\right]$$

(ج) ابتدا ارزش هر حالت را با سیاست فعلی حساب می‌کنیم.

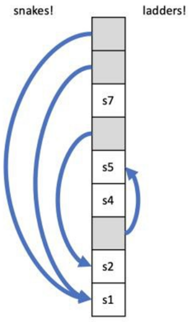
$$v_{k+۱}(s) \leftarrow \sum_{a} \pi(a|s) \sum_{s'} \mathcal{P}^a_{ss'}\left[\mathcal{R}^a_{ss'} + \gamma v_k(s')\right]$$

Iteration	State	Value	Policy Action
۱	۱	۱	f
	۲	۲	f
	۳	-۰.۱۴۳	f
۲	۱	۱	f
	۲	۲	f
	۳	۳	f
۳	۱	۱	f
	۲	۲	f
	۳	۳	f

سپس سیاست را با ارزش های بدست آمده بروزرسانی می‌کنیم.

$$\pi'(s) \leftarrow \arg\max_{a \in \mathcal{A}(s)} \sum_{s'} \mathcal{P}^a_{ss'} \left[\mathcal{R}^a_{ss'} + \gamma v_\pi(s')\right]$$

۳. (نمره)



شما در حال بازی نسخه‌ای ساده‌شده از یک بازی کلاسیک هستید که «مار و پله» نامیده می‌شود.

- شما در امتداد یک مسیر یک‌بعدی حرکت می‌کنید که خانه‌های $s_۱, s_۲, s_۴, s_۵, s_۷$ را شامل می‌شود.
- شما دو عمل دارید: climb و quit.
- اگر از حالت s_i عمل climb را انتخاب کنید، با احتمال ۰/۵ یک خانه بالا می‌روید و با احتمال ۰/۵ دو خانه بالا می‌روید.
- اما! اگر روی خانه‌ای فرود بیایید که پله‌ای از آن بالا می‌رود، بلافاصله و با احتمال ۱ به خانهٔ مقصد پله منتقل می‌شوید.
- و اگر روی خانه‌ای فرود بیایید که ماری از آن پایین می‌آید، بلافاصله و با احتمال ۱ به خانهٔ مقصد مار سقوط می‌کنید.
- بنابراین، برای مثال، اگر در $s_۵$ باشید و climb کنید:
  - با احتمال ۰/۵ یک خانه بالا می‌روید و به خانهٔ خاکستری $s_۶$ می‌رسید؛ چون $s_۶$ مار دارد، بلافاصله به $s_۲$ سقوط می‌کنید.
  - با احتمال ۰/۵ دو خانه بالا می‌روید و به $s_۷$ می‌رسید.
  - اگر در $s_۷$ باشید و climb کنید، ابتدا به خانهٔ خاکستری $s_۸$ می‌روید و چون $s_۸$ مار دارد، به $s_۱$ منتقل می‌شوید.
- اگر عمل quit را انتخاب کنید، بازی تمام می‌شود و دیگر نمی‌توانید حرکتی انجام دهید.
- هر اپیزود جدید از $s_۱$ شروع می‌شود.
- پاداشِ انتخاب climb در هر حالت برابر با ۰ است.
- پاداشِ انتخاب quit در حالت s_i برابر با i است.

(آ) سیاستِ بهینه در افق زمانی یک چیست؟ برای هر یک از حالات $s_۱, s_۲, s_۴, s_۵, s_۷$ بنویسید که عملِ بهینه چیست.

(ب) اگر مقادیر Q تمام حالات را با صفر مقداردهی اولیه کنید و یک دور value iteration بدون تنزیل ($\gamma = ۱$) انجام دهید، تابع Q حاصل چه خواهد بود؟ مقدار $Q(s, \text{quit})$ و $Q(s, \text{climb})$ را برای هر یک از حالات $s_۱, s_۲, s_۴, s_۵, s_۷$ مشخص کنید.

(ج) اگر $\gamma = ۱$ (یعنی هیچ تنزیلی نداریم)، سیاستِ بهینه در افق نامتناهی چیست؟ برای هر یک از حالات $s_۱, s_۲, s_۴, s_۵, s_۷$ بنویسید که عملِ بهینه چیست.

(د) اکنون تنزیل ($\gamma < ۱$) را در نظر بگیرید. نابرابری‌ای بنویسید که شامل مقادیر عددی، γ، $Q(s_۲, \text{climb})$ و $Q(s_۷, \text{climb})$ باشد و شرط را مشخص کند که تحت آن عملِ بهینه در حالت $s_۵$، انتخابِ quit باشد.

حل.

(آ) سیاست بهینه در افق ۱.

	s_1	s_2	s_4	s_5	s_7
عملِ بهینه	quit	quit	quit	quit	quit

(ب) یک دور Value Iteration با $V^{(0)} \equiv 0$ و $\gamma = 1$.

	s_1	s_2	s_4	s_5	s_7
$Q(s, \texttt{quit})$	۱	۲	۴	۵	۷
$Q(s, \texttt{climb})$	۰	۰	۰	۰	۰

(ج) افقِ نامتناهی بدون تنزیل ($\gamma = 1$).

	s_1	s_2	s_4	s_5	s_7
عملِ بهینه	climb	climb	climb	climb	quit

(د) شرط بهینگیِ quit در s_5 با تنزیل. می‌خواهیم $Q(s_5, \texttt{quit}) > Q(s_5, \texttt{climb})$.

$$
\begin{aligned}
Q(s_5, \texttt{quit}) &= 5, \\
Q(s_5, \texttt{climb}) &= R(s_5, \texttt{climb}) + \tfrac{1}{2}\gamma \max_{a'} Q(s_2, a') + \tfrac{1}{2}\gamma \max_{a'} Q(s_7, a') \\
&= 0 + \tfrac{1}{2}\gamma \max\big(Q(s_2, \texttt{quit}), Q(s_2, \texttt{climb})\big) + \tfrac{1}{2}\gamma\, Q(s_7, \texttt{quit}) \\
&= \tfrac{\gamma}{2}\Big(\max\big(Q(s_2, \texttt{quit}), Q(s_2, \texttt{climb})\big) + 7 \Big).
\end{aligned}
$$

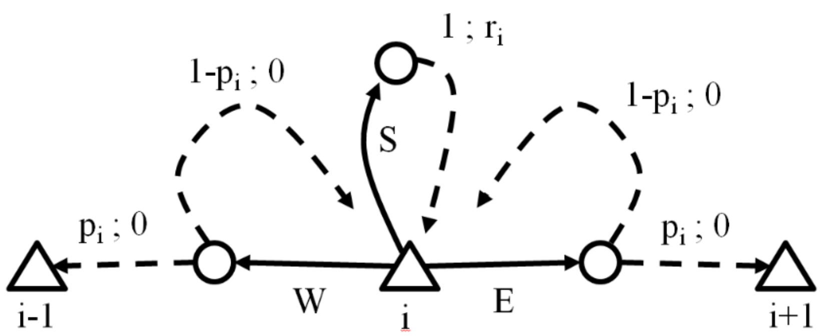
پس شرط بهینگیِ quit در s_5 برابر است با

$$\boxed{5 > \tfrac{\gamma}{2}\big(\max\{2, Q(s_2, \texttt{climb})\} + 7 \big)}.$$

۴. (نمره)

در امتداد یک بزرگراه اصلی، N شهر وجود دارد که با شماره‌های ۱ تا N شماره‌گذاری شده‌اند. شما تاجری از شهر ۱ هستید (شروع شما از آنجاست). هر روز، می‌توانید یکی از کارهای زیر را انجام دهید: به شهر مجاور بروید (حرکت به سمت شرق یا غرب)، یا در شهر فعلی بمانید و به تجارت بپردازید (عمل Stay). اگر تصمیم بگیرید از شهر i سفر کنید، با احتمال p_i با موفقیت به شهر بعدی می‌رسید، اما با احتمال $1 - p_i$ گرفتار طوفان می‌شوید و در این صورت روزتان هدر می‌رود و هیچ‌جا نمی‌روید. اگر تصمیم بگیرید در شهر i بمانید و تجارت کنید، پاداشی برابر $r_i > 0$ دریافت می‌کنید؛ روزهای سفر پاداشی برابر با صفر دارند، چه موفق به رفتن به شهر دیگر شوید و چه نشوید.

نمودار زیر، اقدامات و انتقال‌ها از شهر i را نشان می‌دهد. پیکان‌های پررنگ نشان‌دهنده اقدامات هستند؛ پیکان‌های خطچین انتقال‌های حاصل را با برچسبی شامل احتمال و پاداش (به همین ترتیب) نمایش می‌دهند.



(آ) اگر برای همه i، داشته باشیم $r_i = ۱$، $p_i = ۱$ و $\gamma = ۰/۵$ باشد، مقدار $V_{\text{stay}}(۱)$ در حالتی که همیشه stay انتخاب می‌شود چقدر است؟

(ب) اگر برای همه i، داشته باشیم $r_i = ۱$، $p_i = ۱$ و $\gamma = ۰/۵$ باشد، مقدار بهینه $V^*(۱)$ بودن در شهر ۱ چقدر است؟

(ج) اگر مقادیر r_i و p_i اعداد مثبت معلوم باشند و discount factor تقریباً برابر با یک باشد ($\gamma \approx ۱$)، سیاست بهینه را توصیف کنید.

(د) فرض کنید از الگوریتم Value Iteration استفاده می‌کنیم. $V_k(s)$ مقدار حالت s پس از k مرحله مقدارگذاری است و همه مقادیر اولیه صفر هستند.
اگر مقدار بهینه برای بودن در شهر ۱ مثبت باشد یعنی $V^*(۱) > ۰$، بیشترین مقدار k که در آن ممکن است $V_k(۱) = ۰$ باشد چیست؟

(ه) اگر همه‌ی r_i و p_i مثبت باشند، بیشترین مقدار k که در آن ممکن است $V_k(s) = ۰$ برای برخی حالت‌ها باشد چیست؟

(و) فرض کنید r_iها و p_iها را نمی‌دانیم، بنابراین تصمیم می‌گیریم از Q-Learning استفاده کنیم.
فرض کنید دنباله زیر از حالت‌ها، کنش‌ها و پاداش‌ها تجربه شده است:
(s=1, a=stay, r=4)،
(s=1, a=east, r=0)،
(s=2, a=stay, r=6)،
(s=2, a=west, r=0)،
(s=1, a=stay, r=4)

اگر نرخ یادگیری برابر ۰/۵، discount factor برابر ۱ و همه مقادیر اولیه $Q(s,a) = ۰$ باشند، مقدارهای نهایی $Q(s,a)$ را در جدول زیر وارد کنید. هر سطر باید مقادیر q-value پس از انتقال مشخص‌شده در ستون اول را نشان دهد. مقادیر بدون تغییر را می‌توان خالی گذاشت.

(s,a,r,s’)	Q(1,S)	Q(1,E)	Q(2,W)	Q(2,S)
initial	0	0	0	0
(1,S,4,1)				
(1,E,0,2)				
(2,S,6,2)				
(2,W,0,1)				
(1,S,4,1)				

حل.

(آ)

$$\forall i \in \{۱, \dots, N\}, \quad V^{\text{stay}}(i) = r_i + \gamma V^{\text{stay}}(i)$$
$$V^{\text{stay}}(i) = ۱ + ۰/۵ V^{\text{stay}}(i)$$
$$V^{\text{stay}}(i) = ۲$$
$$V^{\text{stay}}(۱) = ۲$$

(ب) برای تمام شهرها (حالت‌ها) $i = ۱, \dots, N$، معادلات بلمن به صورت زیر نوشته می‌شوند:

$$V^*(i) = \max\left\{ \underbrace{r_i + \gamma V^*(i)}_{\text{stay}}, \underbrace{p_i\gamma V^*(i-۱) + (۱-p_i)\gamma V^*(i)}_{\text{left}}, \underbrace{p_i\gamma V^*(i+۱) + (۱-p_i)\gamma V^*(i)}_{\text{right}} \right\}$$

با توجه به اینکه $p_i = ۱$، معادله به شکل زیر ساده می‌شود:

$$V^*(i) = \max\left\{ \underbrace{۱ + \gamma V^*(i)}_{\text{stay}}, \underbrace{\gamma V^*(i-۱)}_{\text{left}}, \underbrace{\gamma V^*(i+۱)}_{\text{right}} \right\}$$

از آنجا که همه شهرها پاداش یکسان دارند ($r_i = ۱$) و مقدار $V^*(i)$ در تمام حالت‌ها یکسان است، ماکزیمم مقدار همیشه با عمل stay به دست می‌آید. در نتیجه:

$$V^*(i) = ۱ + \gamma V^*(i) \implies V^*(i) = \frac{۱}{۱-\gamma} = ۲ \quad (\text{برای}\ \gamma = ۰/۵)$$

(ج) سیاست بهینه این است که همیشه به سمت شهری با بیشترین پاداش حرکت کنید. پس از رسیدن به آن شهر، برای همیشه در آنجا بمانید و به تجارت ادامه دهید.

(د) با فرض $r_i > ۰$، بزرگترین مقدار k برابر با ۰ خواهد بود، زیرا:

$$V_۱(s) = \max\{r_i + ۰, \cdots\} > ۰$$

اگر فرض $r_i > ۰$ را نداشته باشیم، آنگاه بزرگترین k ممکن برابر با $N - ۱$ خواهد بود.

۱. از آنجا که $V^*(۱) > ۰$، حداقل یکی از r_iها باید strictly positive باشد.

۲. پس از یک تکرار:

$$V_۱(i) > ۰$$

۳. پس از دو تکرار:

$$V_۲(i-۱) > ۰$$

۴. نهایتاً پس از i تکرار:

$$V_i(۱) > ۰$$

۵. اگر برای همه‌ی $j < i$ داشته باشیم $r_j = ۰$، آنگاه:

$$V_j(۱) = ۰ \quad \forall j < i$$

۶. در بدترین حالت وقتی $i = N$:

$$V_{N-۱}(۱) = ۰ \quad \text{امکان پذیر است}$$

اما:

$$V_N(۱) > ۰$$

(ه) با فرض اینکه $r_i > ۰$ باشد، بزرگترین مقدار ممکن برای k برابر با ۰ خواهد بود، زیرا:

$$V_۱(s) = \max\{r_i + ۰, \cdots\} > ۰$$

(و) پس از مشاهده‌ی انتقال $(۱, S, ۴, ۱)$، مقدار $Q(۱, S)$ را update می‌کنیم:

$Q(۱, S) \leftarrow ۰٫۵[۴ + ۱ \cdot ۰] + ۰٫۵(۰) = ۲$

پس از مشاهده‌ی انتقال $(۱, E, ۰, ۲)$، مقدار $Q(۱, E)$ را update می‌کنیم:

$Q(۱, E) \leftarrow ۰٫۵[۰ + ۱ \cdot ۰] + ۰٫۵(۰) = ۰$

پس از مشاهده‌ی انتقال $(۲, S, ۶, ۲)$، مقدار $Q(۲, S)$ را update می‌کنیم:

$Q(۲, S) \leftarrow ۰٫۵[۶ + ۱ \cdot ۰] + ۰٫۵(۰) = ۳$

پس از مشاهده‌ی انتقال $(۲, W, ۰, ۱)$، مقدار $Q(۲, W)$ را update می‌کنیم:

$Q(۲, W) \leftarrow ۰٫۵[۰ + ۱ \cdot ۲] + ۰٫۵(۰) = ۱$

پس از مشاهده‌ی انتقال $(۱, S, ۴, ۱)$، مقدار $Q(۱, S)$ را update می‌کنیم:

$Q(۱, S) \leftarrow ۰٫۵[۴ + ۱ \cdot ۲] + ۰٫۵(۲) = ۴$

۵. (نمره) یک مدل MDP با افق نامتناهی و ضریب تخفیف γ با مشخصات زیر در نظر بگیرید:

$$T(s, a, s') = P(s'|f(s, a)), \quad R(s, a, s') = R(s, a)$$

که در آن $f : S \times A \to Y$ یک تابع قطعی به مجموعه حالت‌های پس‌ازتصمیم Y است. دنباله حالت‌ها s_t، اقدامات a_t، حالت‌های پس‌ازتصمیم y_t و پاداش‌ها r_t به صورت زیر است:

$$\begin{tikzcd}
(s_۰, a_۰) \arrow[r, "f"] \arrow[dr] & y_۰ \arrow[r, "P"] & (s_۱, a_۱) \arrow[r, "f"] \arrow[dr] & y_۱ \arrow[r, "P"] & (s_۲, a_۲) \arrow[dr] & \\
& r_۰ & & r_۱ & & r_۲
\end{tikzcd}$$

مقدار مورد انتظار با تخفیف پاداش‌ها تحت سیاست π به صورت زیر تعریف می‌شود:

$$V^{\pi}(s_۰) = \mathbb{E}\left[\sum_{t=۰}^{\infty} \gamma^t R(s_t, a_t) \;\middle|\; a_t = \pi(s_t)\right]$$

و تابع ارزش بهینه به صورت $V^*(s) = \max_{\pi} V^{\pi}(s)$. برای حالت‌های پس‌ازتصمیم:

$$W^{\pi}(y_۰) = \mathbb{E}\left[\sum_{t=۱}^{\infty} \gamma^{t-۱} R(s_t, a_t) \;\middle|\; y_۰\right], \quad W^*(y) = \max_{\pi} W^{\pi}(y)$$

(آ) W^* را بر حسب V^* به دست آورید.

(ب) V^* را بر حسب W^* بنویسید. راهنمایی: ابتدا V^* را بر حسب Q بنویسید و سپس آن را بر اساس W^* بنویسید.

(ج) می‌دانیم معادله بلمن فورد برای تابع ارزش به شکل زیر تعریف می‌شود.

$$V^*(s) = \max_a \left[R(s,a) + \gamma \sum_{s'} T(s,a,s') V^*(s') \right]$$

معادل این معادله برای W^* را ارائه دهید.

(د) جاهای خالی را در الگوریتم زیر طوری پر کنید که یک الگوریتم policy iteration به دست آید. این الگوریتم باید تضمین کند که π^* را پیدا می‌کند. همچنین هر مورد را توضیح دهید:

- سیاست $\pi^{(1)}$ را به صورت دلخواه مقداردهی اولیه کنید
- برای $i = 1, 2, 3, \ldots$:
  - $W^{\pi^{(i)}}(y)$ را برای تمام $y \in Y$ محاسبه کن
  - سیاست جدید $\pi^{(i+1)}$ را طوری محاسبه کن که $\pi^{(i+1)}(s) = \arg\max_a$ ________________
  - اگر ________________________ برای تمام $s \in S$، $\pi^{(i)}$ را برگردان

(ه) با توجه به دنباله مشاهده شده $s_t, a_t, y_t, s_{t+1}, a_{t+1}, y_{t+1}$ و نرخ یادگیری $\alpha \in (0, 1)$، قاعده به‌روزرسانی برای تخمین W^* را مشابه قاعده به‌روزرسانی الگوریتم Q-learning کامل کنید.

$$W(y_t) \leftarrow (1-\alpha) W(y_t) + \alpha \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_$$

حل.

(آ) W^* بر حسب V^*

$$\boxed{W^*(y) = \sum_{s'} P(s' \mid y) V^*(s')}$$

$$\begin{aligned}
W^*(y_0) &= \mathbb{E}\left[R(s_1,a_1) + \gamma R(s_2,a_2) + \gamma^2 R(s_3,a_3) + \cdots \mid y_0\right] \\
&= \sum_{s_1} P(s_1 \mid y_0) \mathbb{E}\left[R(s_1,a_1) + \gamma R(s_2,a_2) + \cdots \mid s_1\right] \\
&= \sum_{s_1} P(s_1 \mid y_0) V^*(s_1)
\end{aligned}$$

با استفاده از مستقل‌بودن V^* از زمان، جایگزین می‌کنیم $y_0 \to y$ و $s_1 \to s'$.

(ب) V^* بر حسب W^*

$$\boxed{V^*(s) = \max_a \left[R(s,a) + \gamma W^*(f(s,a))\right]}$$

$$\begin{aligned}
V^*(s_0) &= \max_{a_0} Q(s_0, a_0) \\
&= \max_{a_0} \mathbb{E}\left[R(s_0,a_0) + \gamma R(s_1,a_1) + \gamma^2 R(s_2,a_2) + \cdots \mid s_0, a_0\right] \\
&= \max_{a_0} \left(R(s_0,a_0) + \mathbb{E}\left[\gamma R(s_1,a_1) + \gamma^2 R(s_2,a_2) + \cdots \mid f(s_0,a_0)\right]\right) \\
&= \max_{a_0} \left(R(s_0,a_0) + \gamma W^*(f(s_0,a_0))\right)
\end{aligned}$$

با تعمیم و تغییر نام متغیرها $s_0 \to s$، $a_0 \to a$.

(ج) معادله بلمن برای W^*

$$\boxed{W^*(y) = \sum_{s'} P(s'|y) \max_a \left(R(s', a) + \gamma W^*(f(s', a))\right)}$$

$$\begin{aligned} W^*(y) &= \sum_{s'} P(s'|y) V^*(s') \quad \text{(از قسمت الف)} \\ &= \sum_{s'} P(s'|y) \max_a \left[R(s', a) + \gamma W^*(f(s', a))\right] \quad \text{(از قسمت ب)} \end{aligned}$$

(د) Policy iteration

$$\boxed{\pi^{(i+۱)}(s) = \arg\max_a \left[R(s, a) + \gamma W^{\pi^{(i)}}(f(s, a))\right] \quad ; \quad \pi^{(i)}(s) = \pi^{(i+۱)}(s)}$$

$$\begin{aligned} Q^\pi(s_۰, a_۰) &= \mathbb{E}\left[R(s_۰, a_۰) + \gamma R(s_۱, a_۱) + \gamma^۲ R(s_۲, a_۲) + \cdots \mid s_۰, a_۰\right] \\ &= R(s_۰, a_۰) + \gamma \mathbb{E}\left[R(s_۱, a_۱) + \gamma R(s_۲, a_۲) + \cdots \mid f(s_۰, a_۰)\right] \\ &= R(s_۰, a_۰) + \gamma W^\pi(f(s_۰, a_۰)) \end{aligned}$$

بهبود سیاست از $\pi^{(i+۱)}(s) = \arg\max_a Q^{\pi^{(i)}}(s, a)$ استفاده می‌کند. هنگامی که سیاست همگرا شد، الگوریتم پایان می‌یابد.

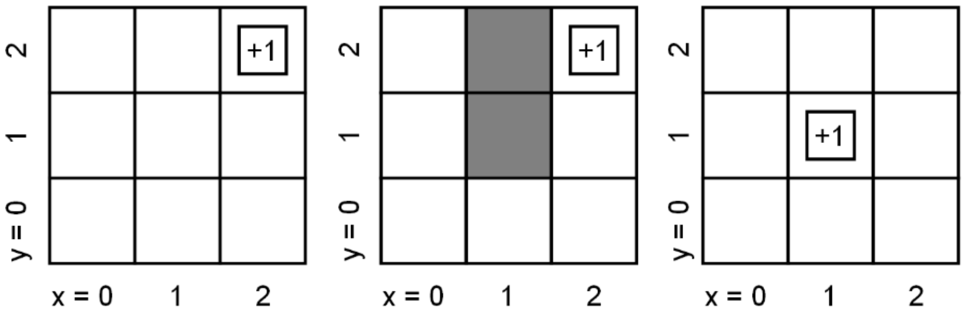
(ه) قاعده به‌روزرسانی برای W^*

$$\boxed{W(y_t) \leftarrow (۱ - \alpha) W(y_t) + \alpha \max_a \left(R(s_{t+۱}, a) + \gamma W(f(s_{t+۱}, a))\right)}$$

$$\begin{aligned} W^*(y) &= \mathbb{E}_{s'}\left[\max_a \left(R(s', a) + \gamma W^*(f(s', a))\right)\right] \\ &\approx \max_a \left(R(s_{t+۱}, a) + \gamma W(f(s_{t+۱}, a))\right) \quad \text{(تخمین با یک نمونه)} \end{aligned}$$

تخمین فعلی را با نرخ یادگیری α به سمت مقدار هدف به‌روزرسانی کنید.

۶. (نمره) در مسئله زیر مربوط به agent gridworld، می‌تواند اعمال N، S، E، W را انجام دهد که عامل را به ترتیب به سمت شمال، جنوب، شرق و غرب یک خانه حرکت می‌دهند. هیچ نویزی وجود ندارد، بنابراین این اعمال همیشه agent را در جهت مورد نظر حرکت می‌دهند، مگر آنکه آن جهت به بیرون از شبکه یا به خانه‌ای مسدود (خاکستری) منتهی شود، که در آن صورت هیچ حرکتی انجام نمی‌شود. خانه‌هایی که دارای +۱ هستند همچنین اجازهٔ اجرای عمل X را می‌دهند که عامل را از شبکه خارج کرده و وارد حالت نهایی (terminal) می‌کند. پاداش برای تمام انتقال‌ها صفر است، به جز انتقال خروج که پاداش آن +۱ است. discount factor برابر 0.5 فرض می‌شود.



(A) (B) (C)

(آ) مقدارهای بهینه برای شبکه (A) را کامل کنید.

(ب) سیاست بهینه برای شبکه (B) را مشخص کنید.

(ج) فرض کنید برای هر حالت غیرنهایی $s = (x, y)$ مجموعه‌ای از ویژگی‌های حقیقی $f_i(s)$ داریم، و می‌خواهیم مقدار بهینه $V^*(s)$ را با رابطهٔ خطی زیر تقریب بزنیم:

$$V(s) = \sum_i w_i \cdot f_i(s)$$

اگر ویژگی‌ها $f_۱(x, y) = x$ و $f_۲(x, y) = y$ باشند، مقادیر $w_۱$ و $w_۲$ را طوری بدهید که سیاست به‌دست‌آمده از (one-step look-ahead) در شبکه (A) بهینه باشد.

(د) آیا می‌توان مقادیر واقعی و بهینه V^* را برای شبکه (A) تنها با استفاده از این دو ویژگی نمایش داد؟ چرا یا چرا نه؟

(ه) برای هر یک از مجموعه ویژگی‌های زیر مشخص کنید کدام یک (در صورت وجود) از MDPهای شبکه‌ای بالا می‌توانند «حل شوند»؛ به این معنا که بتوان مقادیری (احتمالاً غیربهینه) برای $V(s)$ پیدا کرد که سیاست حاصل از نگاه یک‌مرحله‌ای به آینده، بهینه باشد.

i. $f_۱(x, y) = x$، $f_۲(x, y) = y$

ii. برای هر (i, j)، ویژگی $f_{i,j}(x, y) = ۱$ اگر $(x, y) = (i, j)$، و صفر در غیر این صورت.

iii. $f_۱(x, y) = (x - ۱)^۲$، $f_۲(x, y) = (y - ۱)^۲$، و $f_۳(x, y) = ۱$

حل.

(آ)

$$V(۲, ۲) = ۱ \quad \text{(با انجام عمل خروج)}$$
$$V(۱, ۲) = ۰/۵ \cdot V(۲, ۲) = ۰/۵$$
$$V(۲, ۱) = ۰/۵ \cdot V(۲, ۲) = ۰/۵$$
$$V(۱, ۱) = ۰/۵ \cdot V(۲, ۱) = ۰/۲۵$$
$$V(۰, ۲) = ۰/۵ \cdot V(۱, ۲) = ۰/۲۵$$
$$V(۲, ۰) = ۰/۵ \cdot V(۲, ۱) = ۰/۲۵$$
$$V(۱, ۰) = ۰/۵ \cdot V(۲, ۰) = ۰/۱۲۵$$
$$V(۰, ۱) = ۰/۵ \cdot V(۱, ۱) = ۰/۱۲۵$$
$$V(۰, ۰) = ۰/۵ \cdot V(۱, ۰) = ۰/۰۶۲۵$$

ماتریس نهایی مقادیر:

	$x=۰$	$x=۱$	$x=۲$
$y=۲$	۰٫۲۵	۰٫۵	۱
$y=۱$	۰٫۱۲۵	۰٫۲۵	۰٫۵
$y=۰$	۰٫۰۶۲۵	۰٫۱۲۵	۰٫۲۵

(ب)

	$x=۰$	$x=۱$	$x=۲$
$y=۲$	$\downarrow$		$\checkmark$
$y=۱$	$\downarrow$		$\uparrow$
$y=۰$	$\rightarrow$	$\rightarrow$	$\uparrow$

(ج)
- اگر $|w_۱| > |w_۲|$، عامل ترجیح می‌دهد در جهت افقی حرکت کند:
  - $w_۱ > ۰$: به شرق
  - $w_۱ < ۰$: به غرب
- اگر $|w_۱| < |w_۲|$، عامل ترجیح می‌دهد در جهت عمودی حرکت کند:
  - $w_۲ > ۰$: به شمال
  - $w_۲ < ۰$: به جنوب
- اگر $|w_۱| = |w_۲|$:
  - هر دو مثبت: حرکت به‌صورت قطری به شمال‌شرق (رسیدن به (۲،۲))
  - هر دو منفی: حرکت به جنوب‌غرب (دور شدن از هدف)

در نتیجه، اگر $w_۱ = w_۲ = c > ۰$ باشد و عامل از برخورد با دیوار اجتناب کند، مسیر بهینه را به سمت (۲،۲) طی کرده و با انجام عمل X پاداش می‌گیرد.

(د) فرض می‌کنیم تابع ارزش را با فرم زیر تقریب می‌زنیم:

$$V(x,y) = w_۱ x + w_۲ y$$

و مقادیر بهینه برخی خانه‌ها در گرید A به‌صورت زیر است:

$$V(۰,۲) = ۰٫۲۵ \Rightarrow ۰w_۱ + ۲w_۲ = ۰٫۲۵$$
$$V(۱,۲) = ۰٫۵ \Rightarrow ۱w_۱ + ۲w_۲ = ۰٫۵$$
$$V(۲,۲) = ۱ \Rightarrow ۲w_۱ + ۲w_۲ = ۱$$

از رابطهٔ اول:

$$۲w_۲ = ۰٫۲۵ \Rightarrow w_۲ = ۰٫۱۲۵$$

جای‌گذاری در رابطهٔ دوم:

$$w_۱ + ۲(۰٫۱۲۵) = ۰٫۵ \Rightarrow w_۱ = ۰٫۲۵$$

بررسی در معادلهٔ سوم:

$$۲(۰٫۲۵) + ۲(۰٫۱۲۵) = ۰٫۵ + ۰٫۲۵ = ۰٫۷۵ \neq ۱$$

دستگاه معادلات ناسازگار است؛ یعنی هیچ انتخابی از $w_۱, w_۲$ وجود ندارد که همهٔ مقادیر بهینه را دقیق بازسازی کند. تابع ارزش به‌طور غیرخطی به موقعیت و مسیر بهینه وابسته است. ویژگی‌های x, y اطلاعات کافی ندارند.

(ه) i. $f_1(x,y) = x$ ، $f_2(x,y) = y$

فقط شبکه‌ی (A) قابل حل است. زیرا ویژگی‌ها agent را به یک جهت ثابت هدایت می‌کنند، اما در (B) وجود مانع و در (C) موقعیت مرکزی هدف باعث ناهماهنگی با این جهت می‌شود.

ii. $f_{i,j}(x,y) = 1$ اگر $(x,y) = (i,j)$، وگرنه ۰

هر سه شبکه، (A)، (B) (C) قابل حل هستند؛ زیرا برای هر خانه ویژگی جداگانه وجود دارد و می‌توان با انتخاب مناسب وزن‌ها جهت بهینه را تعیین کرد. (one-hot encoded)

iii. $f_1(x,y) = (x-1)^2$ ، $f_2(x,y) = (y-1)^2$ فقط شبکه‌ی (C) قابل حل است؛ این تابع ارزش زمانی جهت درست ایجاد می‌کند که $w_1 = w_2 = c < 0$ و $w_3 = 0$ انتخاب شود. اما در شبکه‌های (A) و، (B) هدف در گوشه قرار دارد.