Subject: شماره دانشجویی: 810101381

Date

تمرین ۶ کتبی هوش مصنوعی

امین آقاکثیری

MDP

سوال اول

Gridworld:

ستون ردیف

(2,3)

	1	2	3
2			+5
1	S		-5

سوال ①

$$V_{i+1}(S) = \max_a \left(\sum_{s'} T(s,a,s') \left(R(s,a,s') + \gamma V_i(s') \right) \right)$$

$\gamma \rightarrow 0.9$

S	(1,1)	(1,2)	(1,3)	(2,1)	(2,2)	(2,3)
V_0	0	0	-5	0	0	+5
V_1	0	-0	-5	0	7.6	+5
V_2	0	4.522	-5	5.472	8.284	+5

discount

$\gamma = 0.9$

مرحله اول:

$$V_1((1,1)) = \max_a \begin{cases} \text{up: } 0.8 \times (0 + 0.9 \times 0) + 0.1 \times (0 + 0.9 \times 0) + 0.1 \times (0 + 0.9 \times 0) = 0 \\ \text{down: } 0.8 \times (0+0) + 0.1(0+0) + 0.1(0+0) = 0 \\ \text{left: } 0 \\ \text{right: } 0 \end{cases} = 0$$

(action: up → (2,1); left → (1,1); right → (1,2)) — left: 0 → مانند بالا

$V((2,1)) = 0$ → زیرا مانند $V_1((1,1))$ تمام خانه‌های اطراف آن ((1,1) و (1,2) و (2,2)) در مرحله قبل برابر 0 بودند!

$$V((1,2)) = \max_a \begin{cases} \text{up: } 0.8(0+0) + 0.1(0+0) + 0.1(-5 + 0.9(-5)) = -0.95 \\ \text{down: } 0.8(0+0) + 0.1(0+0) + 0.1(-9.5) = -0.95 \\ \text{right: } 0.8(-9.5) + 0.1(0+0) + 0.1(0+0) = -7.6 \\ \text{left: } 0 \end{cases} = 0$$

(right; $-5 + 0.9(-5) \rightarrow -9.5$)

از اینجا به بعد دیگر مقادیری که 0 می‌شوند را نمی‌نویسم برای کوتاه نکردن [illegible]

$$V((2,2)) = \max_a \begin{cases} \text{up: } 0.1(5 + 0.9 \times 5) = 0.95 \\ \text{down: } 0.1(9.5) = 0.95 \\ \text{left: } 0 \\ \text{right: } 0.8(9.5) = 7.6 \end{cases} = +7.6$$

($5 + 0.9 \times 5 \rightarrow 9.5$)

Subject:
Date

مرحله دوم:

$$V_2((1,1)) = \max_a \begin{cases} \text{up: } a(0)=0 \\ \text{down: } 0 \\ \text{left: } 0 \\ \text{right: } 0.8(-0.855)=0 \end{cases} = 0$$

$$V_2((2,1)) = \max_a \begin{cases} \text{up: } 0.1(0.9\times 7.6) = 0.684 \\ \text{down: } 0.1(6.84) = 0.684 \\ \text{left: } 0 \\ \text{right: } 0.8(6.84) = 5.472 \end{cases} = 5.472$$

$$V_2((1,2)) = \max_a \begin{cases} \text{up: } \overset{\text{up}}{0.8}(6.84) + \overset{\text{left}}{0} + \overset{\text{right}}{0.1}(-9.5) = 4.522 \\ \text{down: } \overset{\text{down}}{0.8}(0) + 0 + 0.1(-9.5) = -1.634 \\ \text{left: } 0.1(0) + 0.1(6.84) = 0.684 \\ \text{right: } \overset{\text{right}}{0.8}(-5) + \overset{\text{down}}{0.1}(0) + \overset{\text{up}}{0.1}(4) = -3.6 \end{cases} = 4.522$$

$$V_2((2,2)) = \max_a \begin{cases} \text{up: } 0.8(6.84) + 0.1(9.5) = 6.422 \\ \text{down: } 0.8(0) + 0.1(9.5) = 0.95 \\ \text{left: } 0.1(6.84) + 0.1(0) = 0.684 \\ \text{right: } 0.8(9.5) + 0.1(6.84) + 0.1(0) = 8.284 \end{cases} = 8.284$$

سوال ②

$$\pi^*(s) = \arg\max_a Q^*(s,a) \quad ; \quad Q^*(s,a) = \sum_{s'} T(s,a,s')[R(s,a,s') + (\overset{0.9}{\gamma} V^*(s'))]$$

$$\pi^*((2,1)) = \arg\max_a \begin{cases} \text{up: } \overset{\text{up and left}}{0.9}(4.9248) + 0.1(7.45) \\ \text{down: } 0.8(0) + 0.1(4.924) + 0.1(7.45) \\ \text{left: } 0.9(4.924) + 0.1(0) \\ \text{right: } 0.8(7.45) + 0.1(4.924) + 0.1(0) \end{cases} = \text{right}$$

Subject: 0.1+0.8

Date

اشتباه در خانه (1و2) در صورت امتن به پایین پرتاب میشد

$$\pi((1,1)) = \underset{a}{\operatorname{argmax}} \begin{cases} up: 0.8(4.924) + 0.1(-0) + 0.1(4.069) \\ down: 0.9(-0) + 0.1(4.069) \\ left: 0.9(-0) + 0.1(4.924) \\ right: 0.8(4.069) + 0.1(-0) + 0.1(4.924) \end{cases} = up$$

$$\pi((2,2)) = \underset{a}{\operatorname{argmax}} \begin{cases} up: 0.8(7.45) + 0.1(4.924) + 0.1(9.5) \\ down: 0.8(4.069) + 0.1(4.924) + 0.1(9.5) \\ left: 0.8(4.924) + 0.1(7.45) + 0.1(4.069) \\ right: 0.8(9.5) + 0.1(7.45) + 0.1(4.069) \end{cases} = right$$

$$\pi((1,2)) = \underset{a}{\operatorname{argmax}} \begin{cases} up: 0.8(7.378) + 0.1(-9.5) + 0.1(-0) \\ down: 0.8(4.069) + 0.1(-0.075) + 0.1(-9.5) \\ left: 0.8(-0.075) + 0.1(4.969) + 0.1(7.378) \\ right: 0.8(-9.5) + 0.1(4.069) + 0.1(7.378) \end{cases} = up$$

جدول نهایی:

S	(1,1)	(1,2)	(1,3)	(2,1)	(2,2)	(2,3)
$\pi^*(S)$	up	up	–	right	right	–

سوال (3) کافی است میانگین مجموع reward ها برای هر خانه را بدست آورد که می‌شود Value برای آن خانه:

reward: ° ° -5

I) (1,1) ↑ (1,2) ↑ (1,3)

° ° +5

II) (1,1) ↑ (1,2) ↑ (2,2) ↑ (2,3)

° ° +5

III) (1,1) ↑ (2,1) ↑ (2,2) ↑ (2,3)

Discount Factor $\gamma = 0.9$

discounted reward فرمول: $1 \times r_0 + \gamma r_1 + \gamma^2 r_2 + \dots$

$$V^*((1,1)) = \frac{1}{3}\left[\overbrace{(1\times 0 + 0.9\times(-5))}^{(I)} + \overbrace{(1\times 0 + 0.9\times 0 + (0.9)^2\times 5)}^{(II)} + \overbrace{(1\times 0 + 0.9\times 0 + (0.9)^2\times 5)}^{(III)}\right] = \boxed{1.2}$$

تعداد حضور در آرایش‌ها

$$V^*((2,2)) = \frac{1}{2}\left[\overbrace{(1\times 5)}^{(II)} + \overbrace{(1\times 5)}^{(III)}\right] = \boxed{+5}$$

P4PCO

Subject:
Date

$$\begin{cases} sample = R(s, \pi(s), s') + \gamma V^{\pi}(s') \\ V^{\pi}(s) \leftarrow V^{\pi}(s) + \alpha(sample - V^{\pi}(s)) \end{cases}$$

value برای استیت بعدی

سوال ④ و تشابه

اولین ترنزیشن : $(1,1) \xrightarrow{right} (1,2)$; reward = 0

$sample = 0 + 0.9(0) = 0 \longrightarrow V^{\pi}((1,1)) = 0 + 0.1 = 0$

دومین ترنزیشن : $(1,2) \xrightarrow{right} (1,3)$; reward = −5

$sample = -5 + 0.9(-5) = -9.5 \longrightarrow V\pi((1,2)) = 0 + 0.1(-9.5 + 0) = -0.95$

نتیجه نهایی:

s	(1,1)	(1,2)	(1,3)	(2,1)	(2,2)	(2,3)
$V^{\pi}(s)$	0	−0.95	−5	0	0	+5

DQN

سوال اول

DQN یکی از الگوریتم های پیشرفته در مبحث RL است که Q-Learning را با الگوریتم های deep learning ترکیب می کند.

در واقع از یک deep neural network برای تخمین زدن $Q(s,a)$ ها (که روش محاسبه آنها را با استفاده از الگوریتم Q-Learning بلدیم) استفاده می کند. مزیت این الگوریتم نسبت به Q-learning این است که با استفاده از آن می توان به صاحبه مسائلی با ابعاد بسیار بالا استفاده کرد (تعداد حالت های زیاد). حال کمی به توضیح نحوه عملکرد DQN می پردازیم:

Subject :
Date

این الگوریتم سه جز اصلی دارد :

① Replay Buffer : یک بافر است که تجربیات قبلی ما درون آن به صورت (s, a, r, s') ذخیره می شود که در ادامه برای یافتن ارتباط های بین transition های متوالی و استیت ها استفاده می شود.

② Q-Network : یک deep neural network است که استیت کنونی را به عنوان ورودی می گیرد و مقادیر Q را برای تمام action های ممکن خروجی می دهد. وظیفه این شبکه این است که اختلاف بین Q های پیش بینی شده و Q های هدف را کمینه کند.

③ Target Network : یک شبکه جدا از شبکه بالا است (که توضیح دادم) که برای تعداد مشخصی از اجرای آن، داخلی پارامترهای ثابت است و وظیفه آن تولید Q های هدف برای update کردن Q-network است.

حال با توجه به توضیحات بالا مانند الگوریتم DQN را توضیح می دهم. ابتدا Q-network را با وزن های رندوم می سازیم و یک کپی از آن را هم به عنوان target network در نظر می گیریم (مقادیر را مانند replay buffer هم کمینه است). سپس شروع به انجام transition در محیط می کنیم و هر کدام را که انجام دادیم، به صورت (s, a, r, s') در بافر ذخیره می کنیم. پس از تجربه محیط به تعداد زیاد، به سراغ شبکه های رفیم. به صورت رندوم از بافرمان، شروع به انتخاب batch ی کنیم و برای هر کدام از ترنزیشن ها در این batch، مقدار زیر را محاسبه می کنیم :

$$y = R + \gamma \max_{a} Q_{target}(s', a)$$

PAPCO

Subject:
Date

که مقدار Qtarget از target network بدست می‌آید (با آپدیت کردن weight های target network در هر مرحله

برای match شدن با وزن های Q-network). سپس Q network را با مینیمم کردن مقدار خطا

که به صورت $\frac{1}{2}(y - Q(s,a))^2$ می توان تعریف کرد، آپدیت می کنیم و این فرایند را ادامه می دهیم

تا Q-network ما همگرا شود.