یادگیری ژرف

تمرین ششم، بخش تئوری

جواد راضی (۴۰۱۲۰۴۳۵۴)

سوال اول: Q-Learning

(آ): به‌روزرسانی‌های الگوریتم یادگیری Q

قاعده به‌روزرسانی برای Q-Learning به شکل زیر است:

$$Q(s,a) \leftarrow Q(s,a) + \alpha \left[R(s,a) + \gamma \max_{a'} Q\,(s',a') - Q(s,a)\right]$$

اجرای هر اپیزود:

اپیزود 1: (0,0) → (0,1) → (0,2) → (1,2) → (1,1)

نتیجه: حرکات نامعتبر هستند (خارج از محدوده)، بنابراین به‌روزرسانی‌هایی صورت نمی‌گیرد.

اپیزود 2: (0,0) → (1,0) → (2,0) → (2,1) → (2,2)

از (2,2) (چاه) به (2,1):

$$Q(2,2) = 0 + 0.5 \times [-1 + 1 \times 0 - 0] = -0.5$$

حرکات دیگر: بدون پاداش یا چاه. در نتیجه، تغییری در مقادیر Q ایجاد نمی‌شود.

اپیزود 3: (0,0) → (0,1) → (0,2) → (1,2) → (2,2)

نتیجه: مشابه اپیزود 2، تنها حرکت اول از چاه مقدار Q را به‌روز می‌کند.

(ب) به‌روزرسانی‌های الگوریتم تقریبی Q-Learning

Approximate Q-Learning از ویژگی‌ها برای تقریب تابع Q استفاده می‌کند.

- اگر $\mathrm{F_i(s,a)}$ مقدار ویژگی i برای جفت action-state باشد، تابع Q تقریبی، $Q(s,a) = \sum_i w_i \cdot$ است $F_i(s,a)$.

- قاعده به‌روزرسانی وزن‌ها به شکل زیر است:

$$w_i \leftarrow w_i + \alpha \left[R(s,a) + \gamma \max_{a'} Q\,(s',a') - Q(s,a)\right] \cdot F_i(s,a)$$

اجرای دو اپیزود اول:

اپیزود 1: تغییری ایجاد نمی‌شود، زیرا حرکات نامعتبر هستند.

اپیزود 2: (0,0) → (1,0) → (2,0) → (2,1) → (2,2)

از (2,2) (چاه) به (2,1):

$$w_i = w_i + 0.5 \times [-1 + 1 \times 0 - 0] \times F_i(2,2) \;=\; 0.25$$

سوال دوم: مقایسه روش‌های Value-Based

بخش (آ):

هر دو روش Temporal Difference (TD) و Monte Carlo (MC) می‌توانند برای مسائل گسسته و پیوسته استفاده شوند. اما در عمل، TD معمولاً برای مسائل با فضای حالت گسسته و MC برای مسائل با فضای حالت پیوسته مناسب‌تر است. دلیل این امر این است که درMC ، ما باید تا پایان یک اپیزود صبر کنیم تا بتوانیم ارزش حالات را به‌روز کنیم، که در محیط‌های پیوسته ممکن است زمان‌بر باشد. در مقابل، TD می‌تواند پس از هر گام، ارزش حالات را به‌روز کند.

بخش (ب):

در روشTD ، ارزش حالات پس از هر گام به‌روز می‌شود، در حالی که در روشMC ، این به‌روزرسانی‌ها فقط در پایان هر اپیزود انجام می‌شود. بنابراین، در کل، تعداد به‌روزرسانی‌ها در روش TD بیشتر از روش MC است.

بخش (ج):

برای محاسبه ارزش حالات با استفاده از روش‌های Temporal Difference و Monte Carlo، می‌توانیم از فرمول‌های زیر استفاده کنیم:

برای TD

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha \left[r_{t+1} + \gamma \max_a Q\,(s_{t+1}, a) - Q(s_t, a_t) \right]$$

برای MC

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha [G_t - Q(s_t, a_t)]$$

که در آن G_t بازده کل اپیزود پس از زمان t است.

با استفاده از این فرمول‌ها و داده‌های ارائه شده، می‌توانیم ارزش حالات را برای هر دو روش محاسبه کنیم.

سوال سوم: Deep Q-Learning

(آ) مزیت اصلی استفاده از تابع $(\hat{q}(s, a; w))$در یادگیری تقویتی عمیق (DRL) نسبت به استفاده از یک جدول Q(s, a) ساده، توانایی مدل‌سازی و تخمین Q-valueها در فضاهای حالت و اکشن بزرگ است. در موقعیت‌هایی که فضای حالت و فضای اکشن خیلی بزرگ هستند، ذخیره‌سازی و به‌روزرسانی یک جدول Q برای هر جفت (s, a) ناکارآمد و غیرعملی می‌شود. در عوض، استفاده از یک تابع تقریبی مانند $(\hat{q}(s, a; w))$این امکان را فراهم می‌کند که Q-valueها را با استفاده از پارامترهای قابل یادگیری (مانند وزن‌ها و بایاس‌های یک شبکه عصبی) مدل‌سازی کنیم. این رویکرد به ما اجازه می‌دهد تا در فضاهای پیچیده و بزرگ، تعمیم بهتری داشته باشیم و بهینه‌سازی‌های موثرتری انجام دهیم.

(ب) برای این سوال، لازم است رابطه (۲) و (۳) را در نظر بگیریم. رابطه (۲) به‌روزرسانی وزن‌ها را با استفاده از تابع تقریبی $(\hat{q}(s, a; w))$نشان می‌دهد. رابطه (۳)، تابع هدف $(L(w))$ را تعریف می‌کند که انتظار دارد خطای بین Q-value تخمین زده شده و Q-value واقعی را کمینه کند. بروزرسانی وزن در رابطه (۲) به صورت گرادیان کاهشی تصادفی بر اساس تابع هدف $(L(w))$صورت می‌گیرد. به این معنی

که پارامترهای w در جهت کاهش خطای پیش‌بینی Q-value تغییر می‌کنند. این به‌روزرسانی به‌طور مستقیم با هدف کاهش خطای تابع هدف ($L(w)$)مرتبط است و می‌توان گفت که نمونه‌ای از گرادیان کاهشی تصادفی است.

(ج) در مورد استفاده از target network که در طول آموزش بروزرسانی نمی‌شود (رابطه (۴) و (۵))، بروزرسانی وزن‌ها بر اساس تابع هدف ($L^-(w)$)انجام می‌شود. در این حالت، (w^-)به عنوان یک ثابت در نظر گرفته می‌شود و در نتیجه، بروزرسانی وزن‌ها با استفاده از گرادیان کاهشی تصادفی روی ($L^-(w)$)انجام می‌شود. این رویکرد به عامل کمک می‌کند تا به یک پیش‌بینی ثابت‌تر و دقیق‌تر از Q-valueها برسد، حتی در شرایطی که محیط یادگیری ممکن است متغیر باشد. بنابراین، این روش نیز نمونه‌ای از گرادیان کاهشی تصادفی است که روی تابع هدف تعریف شده عمل می‌کند.

(د) موضوع اصلی در تعیین مقدار مناسب برای فرکانس به‌روزرسانی وزن‌های target network (C) در یادگیری تقویتی عمیق، تعادل بین پایداری و سرعت یادگیری است. این تعادل به‌عنوان یک موضوع trade-off در طراحی و پیاده‌سازی الگوریتم‌های یادگیری تقویتی عمیق مطرح می‌شود.

اگر وزن‌های target network بسیار سریع به‌روز شوند (یعنی C خیلی کوچک باشد)، ممکن است شبکه‌ی target به‌سرعت دچار تغییر شود و این تغییرات سریع می‌تواند منجر به نوسانات زیاد و عدم ثبات در فرآیند یادگیری شود. به‌عبارت دیگر، target network نمی‌تواند یک هدف ثابت و معتبر را برای یادگیری Q-network فراهم کند. این امر می‌تواند به کاهش کارایی و سرعت همگرایی الگوریتم بیانجامد.

از طرف دیگر، اگر وزن‌های target network بسیار کم به‌روز شوند (یعنی C بسیار بزرگ باشد)، این امر می‌تواند منجر به یک الگوریتم بسیار کند شود. در این حالت، target network برای مدت طولانی ثابت می‌ماند و این امر می‌تواند مانع از یادگیری بهینه و سریع Q-network شود. علاوه بر این، اگر تغییرات محیطی یا دینامیک‌های بازی در طول فرآیند یادگیری رخ دهد، target network ممکن است نتواند به‌سرعت این تغییرات را منعکس کند.

بنابراین، انتخاب یک مقدار مناسب برای C نیازمند یافتن تعادلی بین پایداری و سرعت یادگیری است. این تعادل به گونه‌ای است که هم اجازه دهد target network به‌اندازه کافی ثابت بماند تا یک هدف معتبر برای Q-network فراهم کند، و هم اجازه دهد که به‌روزرسانی‌های لازم صورت گیرد تا الگوریتم

بتواند به‌طور موثر به یادگیری و همگرایی ادامه دهد. این امر می‌تواند شـــامل آزمایش‌های گوناگون و تنظیم دقیق پارامترها در طول فرآیند توسعه و آزمایش الگوریتم باشد.

(ه) در یادگیری بانظارت، هدف کاهش خطای پیش‌بینی یک مدل بر اسـاس داده‌های واقعی اسـت. مثلاً در مسئله رگرسیون با استفاده از میانگین مربع خطا (MSE)، سعی می‌شود فاصله بین پیش‌بینی‌های مدل و مقادیر واقعی خروجی را کمینه کنیم. این روش بر پایه داده‌هایی اسـت که نمونه‌گیری شـده‌اند و توزیع آن‌ها ناشناخته است.

در مقابل، در یادگیری تقویتی، ما با مفهومی به نام "replay buffer" مواجه هســـتیم که نقش متفاوتی دارد. در یـادگیری تقویتی، عـامـل (agent) از تعـامـل بـا محیط و جمع‌آوری تجربـه‌هـا (مـانـنـد حـالـت‌هـا، اکشـن‌ها، پاداش‌ها) اسـتفاده می‌کند. Replay buffer یک مجموعه از این تجربیات اسـت که به‌منظور به‌روزرسانی پالیسی عامل استفاده می‌شود. این تجربیات به طور تصادفی از buffer انتخاب شده و برای آموزش و به‌روزرسانی مدل استفاده می‌شوند.

اصلی‌ترین تفاوت بین این دو سناریو در نوع داده‌ها و نحوه استفاده از آن‌ها است. در یادگیری بانظارت، ما به دنبال پیش‌بینی دقیق از داده‌های واقعی هســـتیم و معمولاً از داده‌های نمونه‌گیری شـــده از یک توزیع ناشـناخته اسـتفاده می‌کنیم. در حالی که در یادگیری تقویتی، از تجربیات گذشـته (که در replay buffer ذخیره شـده‌اند) برای یادگیری و بهبود اسـتراتژی‌های عامل اسـتفاده می‌شـود. هدف در یادگیری تقویتی، بهینه‌سازی پاداش‌های کلی در طول زمان است، نه فقط کاهش خطای پیش‌بینی برای داده‌های خاص.