به نام خدا

دانشگاه صنعتی امیرکبیر

(پلی تکنیک تهران)

دانشکده مهندسی کامپیوتر

مبانی و کاربردهای هوش مصنوعی ترم بهار ۱۴۰۳-۱۴۰۲

پروژه سوم

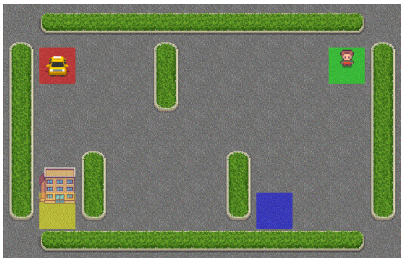
مهلت تحویل جمعه ۱۸ خرداد ساعت ۲۳:۵۵

مقدمه

در این پروژه، ما از الگوریتم‌های مختلف برای حل مسئله‌ی محیط Taxi-v3 استفاده می‌کنیم. محیط Taxi-v3 یک شبیه‌سازی از یک تاکسی است که مسافر را از یک نقطه به نقطه دیگر می‌رساند. هدف از این پروژه، پیاده‌سازی و بررسی عملکرد روش‌های مختلف برای حل یک مسئله مارکوف[۱] است که شامل الگوریتم‌های تکرار ارزش[۲]، تکرار سیاست[۳]، یادگیری Q[4] و ارزیابی مستقیم[۵](مونته کارلو) است.

محیط پروژه

محیط Taxi-v3 به شکل زیر است:



[1] Reinforcement Learning
[2] Value Iteration
[3] Policy Iteration
[4] Q Learning
[5] Direct Evaluation

این محیط شامل یک صفحه‌ی مشبک ۵×۵ می‌باشد که ۴ تا از خانه های آن با رنگ های قرمز، سبز، آبی و زرد مشخص شده. مسافر و مقصد هر کدام در یکی از این خانه های رنگی قرار دارند. مثلا در شکل بالا، مسافر در خانه‌ی سبز و مقصد در خانه‌ی زرد قرار دارد. مکان شروع تاکسی نیز، هر کدام از ۲۵ خانه‌ی جدول می‌تواند باشد. در هر حالت، مسافر می‌تواند یا در یکی از ۴ خانه‌ی رنگی (مبدا) و یا درون تاکسی باشد. تاکسی نیز در یکی از ۲۵ خانه و مقصد نیز در یکی از ۴ خانه رنگی قرار دارد پس تعداد حالت ها برابر با ۵×۴×۲۵ = ۵۰۰ می‌باشد. همچنین در هر مرحله ۶ action مختلف برای تاکسی داریم: ۰- پایین ۱- بالا ۲- راست ۳-چپ ۴- سوار کردن مسافر ۵- پیاده کردن مسافر. پس در کل این مساله دارای ۵۰۰ state(از ۰ تا ۴۹۹) و ۶ action(از ۰ تا ۵) می‌باشد. اگر تاکسی در مکان اشتباه مسافر را سوار یا پیاده کند، امتیاز -۱۰ و در صورت پیاده کردن مسافر در مقصد امتیاز +۲۰ می‌گیرد. در هر دو حالت ذکر شده بازی تمام می‌شود اما در بقیه‌ی حالات بازی ادامه پیدا می‌کند و امتیاز -۱ می‌گیرد(living reward = -1). محیط این بازی به طور پیشفرض قطعی می‌باشد منتها انتظار می‌رود غیرقطعی بودن را نیز در کدی که می‌نویسید در نظر بگیرید؛ به این معنی که برای هر transition، احتمال رفتن به حالت جدید را هم لحاظ کنید. برای اطلاعات بیشتر راجع به کلیت بازی و محیط آن می‌توانید به لینک زیر مراجعه کنید.

https://gymnasium.farama.org/environments/toy_text/taxi/

در انتهای همین فایل نحوه راه‌اندازی پروژه توضیح داده شده است.

تکرار ارزش

این روش بر اساس محاسبه برآورد بهینه‌ی تابع ارزش عمل می‌کند. در هر مرحله، برای هر حالت، ارزش بهینه‌ی تمام اقدامات ممکن محاسبه شده و تابع ارزش به‌روزرسانی می‌شود.
پیاده‌سازی تابع **value_iteration**: باید برای هر حالت، حداکثر ارزش عمل‌های ممکن محاسبه شود و تابع ارزش بر اساس این ارزش‌ها به‌روزرسانی گردد.

$$V_{k+1}(s) \leftarrow \max_a \sum_{s'} T(s,a,s') \left[R(s,a,s') + \gamma V_k(s') \right]$$

در تابع **optimal_policy_extraction**، با یک قدم نگاه رو به جلو، به وسیله‌ی جدول ارزش به دست آمده از **value_iteration**، سیاست بهینه را استخراج می‌کند.

$$\pi_i^*(s) = \arg\max_a Q_i^*(s,a) = \arg\max_a \sum_{s'} T(s,a,s')[R(s,a,s') + \gamma V_{i-1}^*(s')]$$

گزارش 1: آیا استفاده از الگوریتم تکرار ارزش تحت هر شرایطی به همگرایی می‌انجامد.

تکرار سیاست

این روش از یک سیاست اولیه شروع می‌کند و با محاسبه تابع ارزش برای سیاست فعلی و بهبود سیاست بر این اساس پیش می‌رود.

در تابع **evaluate**، شما به ارزیابی و بروزرسانی تابع ارزش (**vtable**) برای یک سیاست داده شده می‌پردازید. این فرآیند، که به عنوان Policy Evaluation شناخته می‌شود، نیازمند محاسبه ارزش انتظاری برای هر حالت بر اساس سیاست فعلی و تابع ارزش قدیمی است. در اینجا، ما باید تمام حالات را در نظر بگیریم و بر اساس سیاست که می‌تواند به صورت غیرمستقیم از محیط یا از یک پارامتر داده شده به تابع دریافت شود، ارزش هر حالت را به روزرسانی کنیم.

$$V_{k+1}^{\pi_i}(s) \leftarrow \sum_{s'} T(s, \pi_i(s), s') \left[R(s, \pi_i(s), s') + \gamma V_k^{\pi_i}(s') \right]$$

در تابع **improvement**، هدف ما این است که یک سیاست برای بهترین عمل و برای هر حالت بر اساس ارزیابی فعلی تابع ارزش (**vtable**) بدست آورید. این فرآیند به معنی محاسبه و برآورد بهترین عمل ممکن برای هر حالت بر اساس تابع ارزش است که از تابع **evaluate** حاصل می‌شود.

$$\pi_{i+1}(s) = \arg\max_a \sum_{s'} T(s, a, s') \left[R(s, a, s') + \gamma V^{\pi_i}(s') \right]$$

گزارش 2: الگوریتم تکرار سیاست را با الگوریتم تکرار ارزش مقایسه کنید.

یادگیری Q

یادگیری Q یک الگوریتم بدون مدل است که تابع ارزش عمل Q را برای هر جفت حالت-عمل بدون نیاز به مدل محیط به‌روزرسانی می‌کند. در تابع **q_learning_train** که برای آموزش با استفاده از الگوریتم Q-Learning طراحی شده است، از شما خواسته شده برای هر اپیزود و در هر گام، Q-values را به‌روزرسانی کنید.

$$sample = R(s, a, s') + \gamma \max_{a'} Q(s', a')$$

$$Q(s, a) \leftarrow (1 - \alpha)Q(s, a) + (\alpha)\,[sample]$$

گزارش 3: توضیح دهید که چرا مقدار epsilon در ابتدای آموزش بالاست و در طی یادگیری بهتر است کاهش یابد؟

گزارش 4: بیان کنید Q-Learning یک الگوریتم Off-Policy است یا On-Policy؟ توضیح دهید.

یادگیری مستقیم

در یادگیری مستقیم، الگوریتم مستقیماً از تجربیات به دست آمده برای یادگیری تابع ارزش یا سیاست بهینه استفاده می‌کند. در تابع **monte_carlo** شما باید با استفاده از روش Monte Carlo، تابع ارزش را برای یک سیاست داده شده بر اساس چندین اپیزود از تجربیات بازی به روز کنید. این فرآیند شامل دنبال کردن یک سیاست طی چندین اپیزود و آپدیت تابع ارزش بر اساس پاداش‌های دریافتی است.

- Idea: Average together observed sample values
  - Act according to π
  - Every time you visit a state, write down what the sum of discounted rewards turned out to be
  - Average those samples

گزارش 5: الگوریتم Q-Learning از TD-Learning استفاده می‌کند، آن را با Monte Carlo مقایسه کنید و بیان کنید استفاده از هرکدام چه مزایا و چه معایبی دارد.

نحوه راه‌اندازی پروژه

ابتدا فایل پروژه رو دانلود کنید. (در کورسز همراه با این فایل قرار گرفته)

یک فایل requirements.txt به همراه پروژه قرار داده شده که شامل همه پکیج‌هایی هست که برای ران کردن پروژه لازمه. می‌توانید با دستور مرحله ۲ اون‌ها را نصب کنید. اما خیلی بهتره قبلش یه virtual environment پایتون بسازید و بعد دستور نصب رو داخل virtual environment اجرا کنید.

۱) نحوه ساخت virtual environment رو می‌تونید تو این لینک ببینید.

https://liara.ir/blog/آموزش-نصب-و-استفاده-از-virtual-environment-در-python

۲) دستور نصب و نحوه ران کردن پروژه:

بعد از اینکه وارد virtual environment شدید کافیه دستور زیر رو بزنید تا همه‌ی پکیج‌های لازم نصب بشن و بعد پروژه رو ران کنید.

```
pip install -r requirements.txt
```

اینجا هم یه شروع سریع برای آغاز کار با jupyter notebook و آشنایی با اون هست.

https://pylie.com/howto/jupyter_project_intro/

فایل requirements.txt شامل تعداد زیادی پکیج هست که ممکنه شما راحت باشین با روش دیگه‌ای اون‌ها را نصب کنید. اگه تو هر قسمتی به هر دلیلی به مسئله‌ای برخورد کردین به ما پیام بدین و ما در اسرع وقت پاسخ می‌دیم.

توضیحات تکمیلی

- انجام پروژه و تهیه گزارش باید به صورت فردی انجام شود. در صورت مشاهده تقلب، برای همه ی افراد نمره صفر لحاظ خواهد شد.
- گزارش خود را در قالب یک فایل PDF به همراه پروژه ی تکمیل شده در سامانه کورسز آپلود کنید.
- فرمت نام‌گذاری فایل آپلودی مانند AI_P3_9931099 باشد.
- در صورت هر گونه سوال یا ابهام می‌توانید از طریق ایمیل با تدریس یاران در ارتباط باشید. همچنین خواهشمند است در متن ایمیل به شماره دانشجویی خود اشاره کنید.
- همچنین می توانید از طریق تلگرام نیز با آیدی های زیر در تماس باشید و سوالاتتان را مطرح کنید:
  - @moved_on
  - @furfinden2
  - @maref02
  - @ahooragorji
- ددلاین این پروژه روز جمعه ۱۸ خرداد ساعت ۲۳:۵۵ است و امکان ارسال با تاخیر وجود ندارد، بنابراین بهتر است انجام پروژه را به روز های پایانی موکول نکنید.