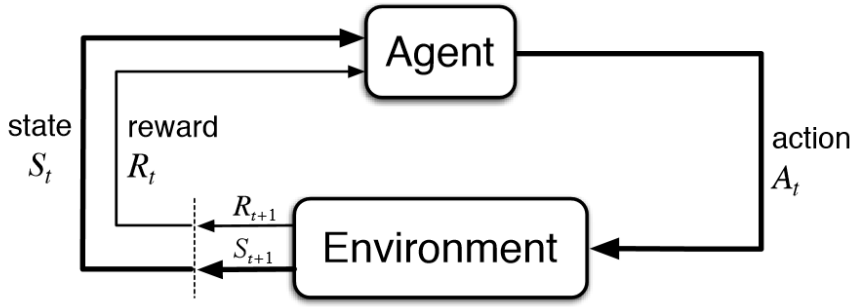
پیاده سازی الگوریتم های Q-Learning و SARSA (پیاده‌سازی)

مقدمه: در این تمرین می‌خواهیم با نحوه پیاده سازی الگوریتم های یادگیری تقویتی آشنا بشویم. یادگیری تقویتی نوعی از روش های یادگیری ماشین است که در آن یک عامل یاد می‌گیرد بر اساس بازخورد هایی که از محیط دریافت می‌کند تصمیم گیری کند. یادگیری تقویتی در زمان هایی که یادگیری با نظارت عملی نباشد و یا محیط پیچیده باشد بسیار کاربرد دارد.

بخش اول: آشنایی با مفاهیم اولیه یادگیری تقویتی

* با توجه به تصویر زیر هر کدام از مفاهیم Agent، Environment، State، Action، Reward و Policy را به صورت بسیار مختصر توضیح دهید.



تصویر۱ - الگوریتم یادگیری تقویتی

بخش ب: پیاده سازی

قسمت الف: آشنایی با محیط

برای پیاده سازی، از یکی از محیط های کتابخانه gym به نام Frozen Lake استفاده می‌کنیم. نمایی از محیط را در تصویر زیر مشاهده می‌کنید.

تصویر۲ – محیط Frozen Lake

محیط استفاده شده یک دریاچه یخ‌زده است که قرار است عامل با انتخاب اکشن های صحیح به خانه هدف برسد. این عامل در هر لحظه می‌تواند یکی از اکشن های چپ، راست، بالا و پایین را انتخاب کند. هرگاه عامل به خانه هدف برسد پاداش مثبت ۱ و به ازای هر حرکت در محیط و افتادن در دریاچه پاداش صفر دریافت می‌کند. برای استفاده از محیط از تکه کد زیر استفاده کنید.

```
import gymnasium as gym
env = gym.make('FrozenLake-v1', map_name="8x8", is_slippery=True,
render_mode='human' if render else None)
```

دریاچه را در حالت ۸ در ۸ قرار دهید. همچنین زمانی که می‌خواهید فرآیند یادگیری عامل را مشاهده کنید متغیر render را برابر True قرار بدهید.

* توضیح دهید که متغیر is_slippery چه تاثیری بر محیط دریاچه دارد.

قسمت ب: پیاده‌سازی عامل تصادفی

معمولا برای بررسی عملکرد الگوریتم های یادگیری تقویتی آن‌ها را با حالتی که عامل اکشن های رندوم انتخاب می‌کند مقایسه می‌کنیم.

* محیط را در حالتی که عامل اکشن های رندوم انتخاب می‌کند اجرا کنید. برای این کار می‌توانید از کد زیر کمک بگیرید.

```
action = env.action_space.sample()
```

* برای حالتی که عامل به صورت تصادفی عمل می‌کند متوسط پاداش دریافتی عامل به ازای هر episode را رسم کنید.

راهنمایی: هر اکشنی که عامل در محیط انجام می‌دهد یک گام یا یک step محسوب می‌شود. episode زمانی شروع می‌شود که عامل اولین اکشن خود را انتخاب کند و هرگاه عامل به هر دلیلی نتواند در محیط ادامه بدهد (در دریاچه بیافتد یا به خانه هدف برسد) episode به پایان می‌رسد.

قسمت ج: پیاده‌سازی عامل Q-Learning و SARSA

در این قسمت قرار است **یکی** از الگوریتم های Q-Learrning و یا SARSA را به دلخواه پیاده‌سازی کنیم. در یادگیری تقویتی انتخاب صحیح پارامتر ها بسیار حائز اهمیت می‌باشد.

* نقش پارامتر ε را بیان کنید و دلیل انتخاب مقدار در نظر گرفته خود را توضیح دهید.

* مفهوم Exploration-Exploitation Balance را توصیح دهید و چگونگی پیاده‌سازی آن با تغییر ε را بیان کنید.

* **یکی** از الگوریتم های Q-Learrning و SARSA را به دلخواه خود پیاده‌سازی کنید و سپس متوسط پاداش دریافتی عامل به ازای هر episode را رسم کنید. نتایج دریافت شده در این قسمت را با حالت عامل تصادفی مقایسه کنید.

* پس از یادگیری، حرکات عامل را با True کردن متغیر render به صورت گرافیکی مشاهده کنید و تصاویر مربوط به رسیدن عامل به خانه هدف را در گزارش کار خود بیاورید.

نکات پیاده‌سازی:

- ضریب تخفیف یا γ را ۰.۹۹ در نظر بگیرید.
- سیاست مورد استفاده عامل را epsilon-greedy در نظر بگیرید.
- محیط را برای هر یک از روش ها ۱۰ بار با حداقل ۱۵۰۰۰ episode اجرا کنید.

نکات تحویل

۱- مهلت تحویل این تمرین ۲۹ دی‌ماه می‌باشد.

۲- انجام این تمرین به صورت یک‌نفره است.

۳- برای انجام این تمرین تنها مجاز به استفاده از زبان برنامه نویسی پایتون هستید.

۴- در صورت وجود تقلب نمره تمامی افراد شرکت کننده در آن نمره صفر لحاظ می‌شود.

۵- در صورتی که از منبعی برای هر بخش استفاده می‌شود، حتماً لینک مربوط به آن در گزارش آورده شود. وجود شباهت بین منبع و پیاده‌سازی در صورت ذکر منبع بلامانع است. اما در صورت مشاهده شباهت با مطالب موجود در سایت‌های مرتبط نمره کسر می‌گردد.

۶- نتایج و تحلیل‌های شما در روند نمره‌دهی دستیاران آموزشی تأثیرگذار است.

۷- لطفا پاسخ تمرین خود را (به همراه کد/گزارش سوال کامپیوتری) به صورت زیر در صفحه درس آپلود نمایید:

HW[HW number]_[Last_name]_[Student number].zip

۸- در صورت وجود هر گونه ابهام یا مشکل می‌توانید از طریق ایمیل با طراحان تمرین در تماس باشید:

- امیرحسین بیرژندی: bizhii2000ut@gmail.com