دانشگاه صنعتی شریف
دانشکده‌ی مهندسی کامپیوتر

هوش مصنوعی
پاییز ۱۳۹۹
استاد: محمدحسین رهبان

تمرین هفتم، بخش اول — **فرآیندهای تصادفی مارکوف** — مهلت ارسال: ۱۶ دی

- مهلت ارسال پاسخ تا ساعت ۵۹ : ۲۳ روز مشخص‌شده است.
- در طول ترم امکان ارسال با تاخیر پاسخ همه‌ی تمارین (به استثنای هفته‌ی امتحان میانترم) تا سقف پنج روز و در مجموع ۱۵ روز، وجود دارد. پس از گذشت این مدت، پاسخ‌های ارسال‌شده پذیرفته نخواهندبود.
- 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 منبع خارج از کتاب و اسلایدهای درس، نام هم‌فکران و آدرس منابع مورد استفاده‌برای حل سوال مورد نظر را ذکرکنید.
- لطفا تصویری واضح از پاسخ سوالات نظری بارگذاری کنید. در غیر این صورت پاسخ شما تصحیح نخواهد شد.
- تمرین مجموعا از ۱۳۴ نمره است که ۱۰۰ نمره از آن اجباری و باقی آن امتیازی است.

سوالات نظری (۸۴ نمره)

۱. (۱۰ نمره) فضای زیر با ابعاد ۳×۱۰۱ را در نظر بگیرید.

-50	1	1	1	...	1	1	1	1	1
Start									
50	-1	-1	-1	...	-1	-1	-1	-1	-1

در خانه شروع ایجنت می‌تواند به صورت deterministic به بالا یا پایین برود. در تمام خانه‌های دیگر که خانه پایانی نیستند، ایجنت به صورت deterministic به سمت راست حرکت می‌کند. خانه‌های پایانی با رنگ قرمز نشان داده شده‌اند.

- (۵ نمره) مقدار utility را برای حرکت اولیه بالا یا پایین به صورت تابعی از γ بنویسید.
- (۵ نمره) اگر تابع reward را به صورت discounted در نظر بگیریم، به ازای چه مقادیری از γ ایجنت باید بالا را به عنوان حرکت اولش انتخاب کند؟

۲. (۳۲ نمره) وحید مشغول یک بازی کارتی ساده است. در هر حرکت از این بازی، وحید می‌تواند یک کارت را از مجموعه کارت‌ها انتخاب کند، عدد آن کارت را به مجموع امتیازاتش اضافه کند و آن را به مجموعه کارت‌ها برگرداند. مجموعه کارت‌ها شامل ۳ کارت می‌شود که امتیاز آن‌ها به ترتیب ۲، ۳ و ۴ است. در هر مرحله، اگر مجموع امتیازات وحید کمتر از ۶ باشد، می‌تواند یا بازی را تمام کند، یا یک کارت جدید انتخاب کند. در این حالت وحید به اندازه مجموعه امتیازاتش درآمد کسب می‌کند (حداکثر ۵). اگر مجموع امتیازات او بزرگ‌تر یا مساوی ۶ شود، مجبور است بازی را تمام کند و درآمدی کسب نمی‌کند. در محاسبه درآمد discount نداریم، یعنی γ برابر صفر است.

- (۶ نمره) حالات (stateها) و حرکت‌ها (action ها) را برای این MDP مشخص کنید.
- (۶ نمره) تابع transition و تابع reward را برای این MDP مشخص کنید.
- (۱۰ نمره) سیاست بهینه این MDP را پیدا کنید.

- (۱۰ نمره) حداقل تعداد مراحلی که باید value iteration برای همگرا شدن به مقادیر دقیق این MDP اجرا شود را محاسبه کنید.

۳. (۴۲ نمره) فرض کنید در حال یک بازی جدید هستید که توضیحات آن در ادامه می‌آید. در هر مرحله، یک عدد رندوم از ۱ تا ۴ با احتمال برابر تولید می‌شود. همچنین اعداد تولید شده در هر مرحله، مستقل از اعداد مرحله قبل تولید می‌شوند. شما می‌توانید در هر مرحله از بازی به اندازه عدد تولید شده امتیاز بگیرید و بازی را تمام کنید، یا این که امتیازی برابر ۱− بگیرید و به بازی ادامه دهید. فرض کنید discount factor برابر ۰/۹ است.

- (۱۲ نمره) نشان دهید این بازی می‌تواند با یک MDP مدل شود و بخش‌های مختلف این MDP را مشخص کنید.
- (۱۵ نمره) با استفاده از الگوریتم value iteration تا ۳ مرحله (۳ بار انتخاب ادامه دادن یا ندادن)، سیاست بهینه را پیدا کنید.
- (۱۵ نمره) با استفاده از policy iteration تا ۳ مرحله سیاست بهینه را پیدا کنید. فرض کنید سیاست اولیه با داشتن اعداد ۱ یا ۲ پایان بازی، و با داشتن اعداد ۳ و ۴ ادامه بازی باشد.

سوالات عملی (۵۰ نمره)

۱. (۵۰ نمره) سیستم مسیریابی پهپادهای یک فروشگاه اینترنتی دچار مشکلاتی شده است. این پهپادها به جای این که در مسیر خواسته شده حرکت کنند، با احتمال $\frac{۸}{۱۰}$ در جهت خواسته شده، با احتمال $\frac{۱}{۱۰}$ به راست آن جهت و با احتمال $\frac{۱}{۱۰}$ به چپ آن جهت می‌روند. یعنی اگر یک پهپاد بخواهد به سمت شرق برود، با احتمال $\frac{۸}{۱۰}$ به همان شرق، با احتمال $\frac{۱}{۱۰}$ به جنوب، و با احتمال $\frac{۱}{۱۰}$ به شمال می‌رود. اگر در جهت نهایی حرکت این پهپادها یک مانع وجود داشته باشد، یا پهپاد بخواهد از فضا خارج شود، حرکتی انجام نخواهد شد. برای این که این فروشگاه پهپادهای خود را از دست ندهد، می‌خواهد یک سیاست بهینه برای بازگرداندن آن‌ها به انبار پیدا کند. به شما یک نقشه ۲ بعدی از منطقه، یک لیست از مکان‌های پرواز ممنوع، و یک لیست از مکان انبارها داده می‌شود. روی نقشه، موانع با None مشخص شده اند. پاداش بودن در فضاهای باز، انبارها، و مناطق پرواز ممنوع نیز مشخص شده است. همچنین، پاداش بودن در یک منطقه برای همه حرکت‌ها یکسان است. هر خانه از نقشه، یا فضای باز است، یا مانع، یا انبار، یا یک منطقه پرواز ممنوع. پهپادها می‌توانند به سمت شمال، جنوب، غرب، یا شرق حرکت کنند که این جهت‌ها به ترتیب با N ، S ، W و E نمایش داده می‌شود. انبارها و مناطق پرواز ممنوع وضعیت‌های نهایی (Terminal State) هستند.
در ادامه یک ورودی نمونه آمده است:

1	-0.1	2
-2	-0.1	-0.1
1	-0.1	None

Warehouses = [(0,0), (2,0)]
No-Fly-Zones = [(0,2), (1,0)]

توجه کنید که خانه (۲،۲) یک مانع است و تمام خانه‌هایی که پاداششان ۰/۱− است، فضای باز هستند.

- (۲۵ نمره) (Value Iteration) در فایل داده شده، تابع

valueiteration(rewards, ware, nofly)

را پیاده‌سازی کنید. در واقع باید با اجرای value iteration روی MDP داده شده، در هر iteration مقادیر محاسبه شده را خروجی دهید. همچنین باید در انتها سیاست بهینه‌ای که بدست می‌آورید نیز خروجی داده شود. از $\gamma = ۰/۹$ استفاده کنید و زمانی الگوریتم را متوقف کنید که

$$\delta \leq ۰/۰۰۱ * \times(۱ - discount) \times discount$$

شود. δ ماکسیموم قدرمطلق اختلاف utility برای یک state در دو iteration متوالی است. در صورتی که دو حرکت به اندازه هم مناسب بودند (یا اختلاف ناچیزی داشتند)، اولویت را به ترتیبِ N ، S ، W ، E اعمال کنید.
متغیر rewards یک لیست دو بعدی از string است که در خانه (i, j) آن، پاداش مربوط به قرار گرفتن در آن خانه در صورت فضای باز، انبار، یا منطقه پرواز ممنوع بودن آمده است. در صورتی که خانه مورد نظر مانع باشد، محتوای آن 'None' است. متغیرهای ware و nofly نیز لیست‌هایی دوبعدی هستند که در هر سطر از آن‌ها متخصات یک انبار یا یک منطقه پرواز ممنوع داده شده است. خروجی تابع شما باید یک tuple باشد که مقدار اول آن، لیستی flat شده از value های بدست آمده در هر iteration است. مقدار دوم آن نیز یک لیست flat شده از سیاست بهینه است. کد شما با استفاده از tester بررسی می‌شود و باید فرمت ورودی و خروجی تابع را رعایت کنید. اضافه کردن توابع دیگر مانعی ندارد.
ورودی و خروجی نمونه در فایل sampleinput.txt قرار داده شده است.

- (۲۵ نمره) (Policy Iteration) در فایل داده شده، تابع

policyiteration(rewards, ware, nofly)

را کامل کنید. در واقع باید روی MDP داده شده، الگوریتم policy iteration را اجرا کنید و در هر مرحله policy بدست آمده را به یک لیست اضافه کنید و خروجی دهید. گاما، شرط توقف الگوریتم در policy evaluation و اولویت action ها را مانند قسمت قبل در نظر بگیرید. فرض کنید سیاست اولیه تماما 'N' باشد. ورودی‌ها مثل قسمت قبل است و ورودی و خروجی نمونه هم در همان فایل sampleinput.txt آمده است.