Zaawansowane Metody Inteligencji Obliczeniowej

Lab 4: Procesy Decyzyjne Markowa (MDP) - część 2.

Michał Kempka Marek Wydmuch

2021-04-04

"Akademia Innowacyjnych Zastosowań Technologii Cyfrowych (AI Tech)",
projekt finansowany ze środków Programu Operacyjnego Polska Cyfrowa POPC.03.02.00-00-0001/20

1 Przypomnienie

1.1 Równania Bellmana dla $v_\pi(s)$ i $q_\pi(s,a)$ dla dowolnej polityki π

Przypomnijmy, że funkcję wartości stanu i stanu-akcji dowolnej polityki π możemy opisać rekurencyjnym równaniem:

$$v_\pi(s) = \sum_{a \in \mathcal{A}} \pi(a|s) \sum_{s',r} \mathcal{P}(s',r|s,a) \Big[r + \gamma v_\pi(s') \Big] \tag{1}$$

$$q_\pi(s,a) = \sum_{s',r} \mathcal{P}(s',r|s,a) \Big[r + \gamma \sum_{a'} \pi(a'|s') q_\pi(s',a') \Big] \tag{2}$$

1.2 Optymalne równania Bellmana

Jeśli wybierzemy optymalną politykę (maksymalizującą oczekiwany zysk) otrzymamy optymalne równania Bellmana:

$$v_*(s) = \max_{a \in \mathcal{A}} \sum_{s',r} \mathcal{P}(s',r|s,a) \Big[r + \gamma v_*(s') \Big] \tag{3}$$

$$q_*(s,a) = \sum_{s',r} \mathcal{P}(s',r|s,a) \Big[r + \gamma \max_{a' \in \mathcal{A}} q_*(s',a') \Big] \tag{4}$$

2 Rozwiązanie skończonego MDP

2.1 Iteracja Polityki

Rozważmy skończone MDP (stany i akcje stanowią skończone zbiory i znamy model przejść). W tym wypadku możemy spróbować iteracyjnie znaleźć optymalną politykę π^* i aproksymakcję funkcji wartości dla wszystkich stanów v_* na podstawie równania Bellmana. Zakładamy, że utrzymujemy zarówno politykę jak i funkcję stanu w formie tabularycznej i inicjalizujemy je dowolnymi wartościami (np. 0). Pomysł polega na naprzemiennym ustalaniu funkcji wartości na podstawie aktualnej polityki (**ang. policy evaluation**) i uaktualnianiu polityki na podstawie obecnej aproksymacji funkcji wartości(**ang. policy improvement**). W dalszej części założymy, będziemy założymy, że polityki są dyskretne.

2.1.1 Ewaluacja polityki

W momencie gdy mamy dostępny pełny model przejść, wartość stanu dla danej polityki można wyznaczyć poprzez rozwiązanie układu równań z $|\mathcal{S}|$ równaniami i zmiennymi. Już jednak dla małej liczby $|\mathcal{S}|$ staje się to niepraktyczne z powodu kosztowności obliczeniowej. Dlatego często problem ten rozwiązuje się iteracyjnie poprzez używanie równania Bellmana jako reguły aktualizacji i poprzedniej estymacji funkcji wartości: $V_{t+1}(s) = \sum_{s',r} p(s', r|s, \pi(s))[r + \gamma V_t(s')]$

2.1.2 Ulepszenie polityki

Ulepszenie polega na wyznaczeniu optymalnej polityki na podstawie aktualnej aproksymacji funkcji wartości. Nie jest to już proces iteracyjny - wystarczy wybrać akcję, która maksymalizuje średni zysk z następników.

2.1.3 Warunek stopu

Zarówno główna procedura (naprzemienna ewaluacja i ulepszanie) jak i sama ewaluacja są metodami iteracyjnymi bez naturalnego warunku stopu. Musimy zatem takie warunki dodać sami. W przypadku głównej procedury będziemy sprawdzać czy polityka zmieniła się w czasie obecnej iteracji. W przypadku ewaluacji stanu możemy przestać gdy wartość funkcji stanu zmieniła się mniej niż zadany próg (mała dodatnia liczba).

Algorytm 1: Pseudokod dla algorytmu **iteracji polityki** (ang. Policy Iteration)

1. Inicjalizacja:
Dowolnie zainicjalizowane V(s) i $\pi(s)$ dla wszystkich $s \in \mathcal{S}$
$\Delta_{min} \leftarrow$ mała dodatnia liczba
2. Ewaluacja polityki
repeat
    $\Delta \leftarrow 0$
    for $s_i \in \mathcal{S}$ **do**
        $v \leftarrow V(s)$
        $V(s) \leftarrow \sum_{s',r} p(s', r|s, \pi(s))[r + \gamma V(s')]$
        $\Delta \leftarrow \max(\Delta, |v - V(s)|)$
until $\Delta < \Delta_{min}$;
3. Polepszenie polityki
$Stop \leftarrow True$
for $s_i \in \mathcal{S}$ **do**
    $a \leftarrow \pi(s)$
    $\pi(s) \leftarrow \arg\max_a \sum_{s',r} p(s', r|s, \pi(s))[r + \gamma V(s')]$
    if $a \neq \pi(s)$ **then**
        $Stop \leftarrow False$
if $Stop$ **then**
    return π
else
    goto 2

Pytanie: Jaka jest złożoność czasowa algorytmu **iteracji polityki**, jak się ma ona do złożoności rozwiązania układu równań?
Pytanie: Jakie problemy ma algorytm iteracji polityki (sytuacja gdzie nie zadziała poprawnie)? Jak rozwiązać te problemy?

2.2 Iteracja Wartości

Zauważmy, że polityka jest pochodną funkcji wartości stanu (w sensie niematematycznym), można by zatem sprawdzać ją na bieżąco w czasie iteracyjnego uaktualniania funkcji stanu. Pomysł taki prowadzi do algorytmu **iteracji wartości**, który iteracyjnie wyznacza wartość stanów na podstawie optymalnego równania Bellmana (i aktualnej estymaty).
Pytanie: Mamy dane MDP z $|\mathcal{S}|$ stanami i bez cykli (tzn. do raz odwiedzonego stanu nie jesteśmy w stanie już wrócić). Jaka jest optymistyczna i pesymistyczna liczba iteracji potrzebna do osiągnięcia zbieżności? Jak będzie w ogólności (dopuszczamy cykle)?

Algorytm 2: Pseudokod dla algorytmu **iteracji wartości** (ang. Value Iteration)

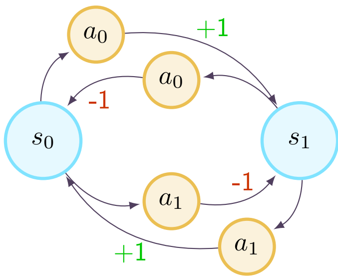
Inicjalizacja:
Dowolnie zainicjalizowane V(s) dla wszystkich $s \in \mathcal{S}$
$\Delta_{min} \leftarrow$ mała dodatnia liczba
repeat
  $\Delta \leftarrow 0$
  for $s_i \in \mathcal{S}$ **do**
    $v \leftarrow V(s)$
    $V(s) \leftarrow \max_a \sum_{s',r} p(s',r|s,a)[r + \gamma V(s')]$
    $\Delta \leftarrow \max(\Delta, |v - V(s)|)$
until $\Delta < \Delta_{min}$;
for $s_i \in \mathcal{S}$ **do**
  $\pi(s) \leftarrow \arg\max_a \sum_{s',r} p(s',r|s,a))[r + \gamma V(s')]$
return π

3 Zadania

3.1 Funkcja wartości akcji

Jak wyglądałby algorytm iteracji wartości obliczający $q_*(s, a)$ zamiast $v_*(s)$?

3.2 Iteracja Polityki



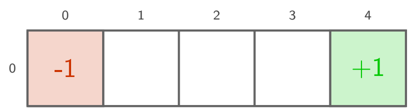
Rysunek 1: Diagram przedstawiający proste MDP z 2 stanami. Nagrody i następniki są deterministyczne i zależą tylko od dwóch akcji.

Używając algorytmu **iteracji polityki**, wyznacz optymalną politykę dla MDP z Rysunku 1, użyj współczynnika dyskontowego $\gamma = 0.5$. Załóż, że niezależnie od stanu początkowa polityki to a_0, a wartość stanu to 0.

3.3 Iteracja Wartości

Używając algorytmu **iteracji wartości**, wyznacz optymalną politykę dla zadanych problemów:

3.3.1 Prosty Korytarz

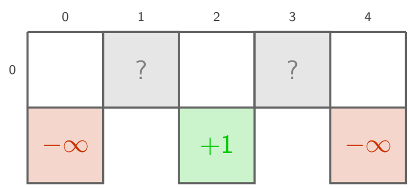


Rysunek 2: Diagram przedstawiający proste środowisko (ang. grid world) z 5 polami i dwoma polami z nagrodami.

Rozważ środowisko z Rysunku 2. Załóż, że agent może znajdować się na dowolnym polu i może iść w **lewo** lub w **prawo**. Skrajne stany są stanami terminalnymi, dotarcie do nich daje pokazane na obrazku nagrody (nie ma innych nagród). Przyjmij $\gamma = 0.5$. Rozważ także sytuację gdzie akcje agenta nie dają pewności przemieszczenia się w zamierzaną stronę: z prawdopodobieństwem $p_c = 0.7$ akcja powiedzie się, z $p_s = 0.2$ agent zostanie w miejscu, a z $p_w = 0.1$ pójdzie w przeciwną stronę.

Załóż, że początkowa funkcja wartości stanu zwraca 0 dla każdego stanu.

3.4 Quo vadis?



Rysunek 3: Przykład środowiska zasięgnięty z wykładu Daivda Silvera.

Rozważ środowisku z Rysunku 3. Agent może poruszać się w lewo, w prawo, lub w dół (tylko wtedy gdy jest to wykonalne). Po wejściu na dolne pole agent otrzyma odpowiednią nagrodę i skończy. Stany oznaczone '?' są nierozróżnialne dla agenta. Załóż $\gamma = 0.5$.

Rozważ też co zmieni się gdy dopuścimy pewne zmiany:

- akcje będą miały niedeterministyczne skutki (np. jak w poprzednim zadaniu)?
- usuniemy współczynnik dyskontowy, zmienimy karę na dolny polach na -100 i dodamy karę za życie?
- jak możemy zmodyfikować wyznaczanie polityki by rozwiązać ten problem?

Fundusze Europejskie Polska Cyfrowa
Rzeczpospolita Polska
Unia Europejska Europejski Fundusz Rozwoju Regionalnego

"Akademia Innowacyjnych Zastosowań Technologii Cyfrowych (AI Tech)",
projekt finansowany ze środków Programu Operacyjnego Polska Cyfrowa POPC.03.02.00-00-0001/20