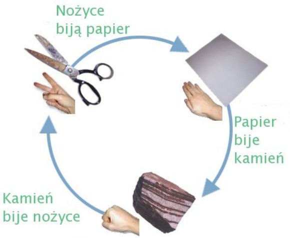
O szukaniu szczęścia w niepewnym świecie: procesy decyzyjne Markowa

Paweł Rychlikowski

Instytut Informatyki UWr

30 kwietnia 2019

Papier, nożyce, kamień



Źródło: Wikipedia

Macierz wypłat

Max/Min	Papier	Nożyce	Kamień
Papier	0	+1	-1
Nożyce	-1	0	+1
Kamień	+1	-1	0

Papier, nożyce, kamień

Pewne rzeczy, o których mówiliśmy tydzień temu, są wyraźnie widoczne dla gry papier-nożyce-kamień.

1. Dla każdej strategii jest optymalna stała odpowiedź:
   Mój przeciwnik gra losowo, ale z przewagą kamienia – zatem ja daję **zawsze** papier
2. Optymalna strategia jest mieszana (w tej grze każde z $p = \frac{1}{3}$)
3. Znajomość optymalnej strategii mieszanej gracza A, nie daje żadnej przewagi graczowi B (i odwrotnie)

Papier, nożyce, kamień – uwagi końcowe

- Istnieją zawody w PNK, ktoś w nich wygrywa
- Co ciekawe: niektórzy istotnie częściej!

Kluczowe sposrzeżenie

Który człowiek (nie dysponując kostką do gry), przegrawszy 3 razy z rzędu jako papier pokaże papier?

Wyobraźmy sobie turniej, w którym gra N ludzi i K programów. Mecz to wiele tur. Jak napisać taki program?

Macierz wypłat Zmyłka i Zgadywacza. Przypomnienie

Nasza gra:

Zg/Zm	1 palec	2 palce
1 palec	2	-3
2 palce	-3	4

Optymalna strategia:

- Zmyłek: trochę częściej 1 palec (bo jak wygra, to więcej)
- Zgadywacz: trochę częściej 1 palec (bo Zmyłek daje go częściej)

Policzyliśmy, że trochę częściej $=$ z prawdopowobieństwem $\frac{7}{12}$

Twierdzenie von Neumana. Przypomnienie

Twierdzenie, von Neuman, 1928

Dla każdej jednoczesnej gry dwuosobowej o sumie zerowej ze skończoną liczbą akcji mamy:

$$\max_{\pi_A} \min_{\pi_B} V(\pi_A, \pi_B) = \min_{\pi_B} \max_{\pi_A} V(\pi_A, \pi_B)$$

dla dowolnych mieszanych polityk π_A, π_B.

Gry wieloturowe

- Można o grze wieloturowej myśleć jako o grze jednoturowej
- Gracze na sygnał kładą przed sobą opis strategii (program)

Uwaga

Optymalną strategią jest MinMax (ExpectMinMax w grach losowych). Ale wiedząc o strategii gracza różnej od optymalnej możemy oczywiście ugrać więcej.

Co pomijamy

- Gry o sumie niezerowej, w których dochodzi możliwość kooperacji.
- Punkt równawagi Nasha (jest zawsze para strategii, że żaden gracz nie chce jej zmienić, wiedząc, że ten drugi nie zmienia).
- Agent musi zdecydować, czy ma być miły dla innego agenta (i budować reputację przy wielu rozgrywkach, słynny dylemat więźnia).

- Coś pomiędzy grami a zwykłym zadaniem przeszukiwania.
- a jednocześnie krok w stronę uczenia ze wzmocnieniem

... o **szukaniu** szczęścia w **niepewnym** świecie ...

MDP a przeszukiwanie

Standardowe przeszukiwanie

Znamy mechanikę świata i wiemy, że akcja w stanie da nam konretny rezultat (inny stan).

MDP

Znamy mechanikę świata i wiemy, że akcja w stanie da nam pewien rozkład prawdopodobieństwa na następnych stanach.

Nie wiemy, co dokładnie się stanie, ale wiemy co **może** się stać i z jakim prawdopodobieństwem.

Własność Markowa

- Przyszłość zależy od ostatniego stanu.
- Nie zależy od historii...
- Chyba, że jej fragment (o długości N) uznamy za część stanu.

Uwaga na wulkany (1)

- Dobrze omawia się MDP na prostych światach na prostokątnej kratce.
- I od takich modeli zaczniemy.

Generalnie myślimy na początku o przestrzeni stanów na tyle małej, że nie będzie kłopotów z pamiętaniem różnych wartości dla **każdego stanu**.

Uwaga na wulkany (2)

Volcano crossing



		-50	20
		-50	
2			

CS221 / Autumn 2017 / Liang & Ermon

Mechanika świata wulkanów

		-50	20
		-50	
2			

- Możliwe 4 akcje (UDLR)
- W normalnym przypadku efekt oczywisty (próba wyjścia poza planszę oznacza pozostanie na polu)
- Z prawdopodobieństwem p możemy się poślizgnąć, wówczas poruszamy się w losowym kierunku.
- Dojście do pola z liczbą kończy grę (i odpowiednią dostajemy wypłatę).

Inny przykład. Gra w kości

Uwaga

Nagroda może być przydzielana w sposób ciągły, nie tylko w stanie końcowym.

- Mamy dwie opcje: pozostanie albo rezygnacja.
- rezygnacja oznacza wypłatę **10$**
- pozostanie to wypłata **4$** po której rzucamy kostką.
- Interpretacja wyniku:
  - 1,2 – koniec gry
  - 3,4,5,6 – gramy dalej

Pytanie

Ile mamy stanów? Odpowiedź: 2

Dla gry w kości

- 1 stan z decyzją, dwie polityki (schemat na tablicy).
- Możemy policzyć oczekiwaną wartość dla każdej:
  - rezygnacja – 10
  - pozostanie – (na tablicy)

MDP – formalna definicja

Definicja

Markowowski proces decyzyjny (MDP) zawiera następujące składowe:

1. S – (skończony) zbiór stanów
2. Stan startowy, $s_{\text{start}} \in S$
3. Actions(s) – zbiór możliwych akcji w stanie s
4. T(s,a,s') – prawdopodobieństwo przejścia z s do s' w wyniku akcji a
5. Reward(s,a,s') – nagroda (wypłata) związana z tym przejściem
6. IsEnd(s) – czy stan jest końcowy?
7. Discount factor, $0 < \gamma \leq 1$ – sprawia, że nagrody w przyszłości cieszą mniej.

MDP – komentarz do definicji

- Można też myśleć, że dla pary (s, a) mamy rozkład prawdopodobieństw po parach (nowy-stan, nagroda).
- Nagroda może być pozytywna bądź negatywna

Uwaga

Oczywiście MDP jest ogólniejsze niż zadanie przeszukiwania (bo wystarczy przypisać niektórym rezultatom p-stwo 1, reszcie 0 i mamy zwykłe zadanie przeszukiwania)

Czym jest rozwiązanie MDP?

- Przypominamy: rozwiązaniem zadania przeszukiwania jest ciąg akcji (ale to nie tu nie działa, bo?)
  - (wyniki akcji są niedeterministyczne, więc nie wystarczy podać jednego ciągu akcji)
- Rozwiązanie: agent musi wiedzieć, co zrobić w każdym stanie.

Polityka

Definicja 1

Polityką deterministyczną nazwiemy funkcję, która każdemu stanowi przypisuje akcję (możliwą w tym stanie).

Definicja 2

Polityką nazwiemy funkcję, która każdemu stanowi przypisuje rozkład prawdopodobieństwa na akcjach (możliwych w tym stanie).

Wartościowanie polityki

- Gdy używamy polityki, otrzymujemy ciąg stanów, akcji i nagród
- Dla takiej ścieżki możemy zsumować nagrody, otrzymując użyteczność dla tej ścieżki
- Wartością polityki jest oczekiwana użyteczność polityki (tzn. wartość oczekiwana zmiennej losowej wyrażającej użyteczność takiej ścieżki)

Discounting

- Realizując politykę, otrzymaliśmy ciąg stanów, nagród i akcji
  - $s_0, a_1, r_1, s_1, a_2, r_2, s_2 \ldots, s_n, a_{n+1}, r_{n+1}, s_{n+1} \ldots$
- Nagroda po uwzględnieniu zniżek:

$$r_1 + \gamma r_2 + \gamma^2 r_3 + \gamma^3 r_4 + \ldots$$

- Widzimy że:
  - Dla $\gamma = 1$ po prostu sumujemy nagrody cząstkowe
  - Dla $0 < \gamma < 1$ mamy możliwość mówienia o wartości nieskończonych ciągów akcji.

Uwaga o γ

- Zwróćmy uwagę, że discounting ma sens również w przypadku, gdy nagroda wypłacana jest jedynie w stanie końcowym.
- Jeżeli wypłata jest tylko w ostatnim stanie, to:
  a) Agent, który wygrywa ($R > 0$) woli dostać ją wcześniej,
  b) agent, który przegrywa ($R < 0$) woli dostać ją później.

Przyśpieszanie zwycięstwa i opóźnianie porażki jest „sensownym” zachowaniem.

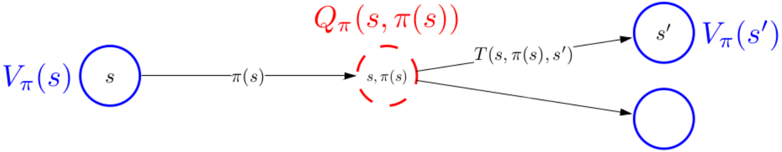
Wartość polityki

Definicja

Wartość $V_\pi(s)$ jest oczekiwaną użytecznością dla agenta startującego w stanie s i działającego zgodnie z polityką π

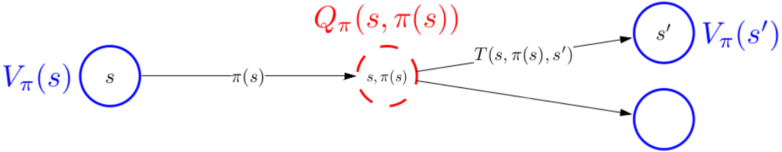
Definicja

Wartość $Q_\pi(s, a)$ jest oczekiwaną użytecznością dla agenta startującego w stanie s, wykonującego w tym stanie akcję a i **dalej** działającego zgodnie z polityką π



Źródło: CS221 / Autumn 2017 / Liang & Ermon

Zależności pomiędzy V i Q



$$Q_\pi(s, a) = \sum_{s'} T(s, a, s')(\mathrm{Reward}(s, a, s') + \gamma V_\pi(s'))$$

Algorytm: policy evaluation

- Napiszmy rekurencyjny wzór dla wartości V (przy zadanej polityce)
- $$V_\pi(s) = \sum_{s'} T(s, \pi(s), s')(\text{Reward}(s, \pi(s), s') + \gamma V_\pi(s'))$$
- Mamy układ równań (liniowych), który można rozwiązywać standardowymi metodami.
- Równań jest tyle co stanów (czyli potencjalnie sporo)

Algorytm: policy evaluation (2)

Możemy ten wzór zmodyfikować, mówiąc: zamiast nieznanego V po prawej stronie weźmiemy poprzednie przybliżenie V:

$$V_\pi^{(t+1)}(s) = \sum_{s'} T(s, \pi(s), s')(\text{Reward}(s, \pi(s), s') + \gamma V_\pi^{(t)}(s'))$$

Algorytm: policy evaluation (3)

1. Zainicjuj $V_\pi^{(0)}(s) \leftarrow 0$, dla wszystkich s
2. Powtarzaj dla $t = 1, \ldots, t_{PE}$
   - Powtarzaj dla każdego stanu s

$$V_\pi^{(t+1)}(s) \leftarrow \sum_{s'} T(s, \pi(s), s')(\text{Reward}(s, \pi(s), s') + \gamma V_\pi^{(t)}(s'))$$

Uwagi implementacyjne

- Kończymy, gdy dla każdego stanu zmiana mniejsza niż ε
- Oczywiście nie musimy pamiętać całej historii, tylko dwa ostatnie jej elementy (stany zmieniane i poprzednie)

Złożoność

$O(t_{PE} SS')$, gdzie S to liczba stanów, a S' (maksymalna) liczba stanów z niezerową $T(s, a, s')$.

Polityka optymalna

- Interesuje nas wyznaczanie polityki (a nie tylko ocenianie jej wartości.

Definicja

Optymalną wartością stanu $V_{opt}(s)$ jest maksymalna wartość stanu (ze względu na wszystkie polityki).

Rekurencja dla polityki optymalnej

Jaka polityka jest optymalna? Taka, która wybiera stany o optymalnej wartości

- Przypominamy, dla **każdej** polityki mamy:

$$Q_\pi(s, a) = \sum_{s'} T(s, a, s')(\text{Reward}(s, a, s') + \gamma V_\pi(s'))$$

- Dla polityki optymalnej: $V_{\text{opt}}(s) = \max_{a \in \text{Actions(s)}} Q_{\text{opt}}(s, a)$

Możemy podstawić do drugiego wzoru wzór na Q_π dla $\pi = \text{opt}$.

$$V_{\text{opt}}(s) = \max_{a \in \text{Actions(s)}} \sum_{s'} T(s, a, s')(\text{Reward}(s, a, s') + \gamma V_{\text{opt}}(s'))$$

Algorytm Iteracji wartości – Bellman, 1957

Polityka optymalna (do poprzedniego slajdu)

$\pi_{\text{opt}}(s) = \arg\max_{a \in \text{Actions(s)}} Q_{\text{opt}}(s, a)$

Nasz wzorek zmieniony na wersję do iterowania

$$V_{\text{opt}}^{(t+1)}(s) = \max_{a \in \text{Actions(s)}} \sum_{s'} T(s, a, s')(\text{Reward}(s, a, s') + \gamma V_{\text{opt}}^{(t)}(s'))$$

Algorytm Bellmana (value iteration)

- Mamy dodatkową pętlę wybierającą optymalną akcję (zamiast akcji danej przez politykę)
- Reszta bez zmian, tak jak w policy evaluation.

Warunki zbieżności

Algorytm jest zbieżny, jeżeli zachodzi któryś z warunków

- $\gamma < 1$
- Graf MDP jest acykliczny

Uwaga

W tym ostatnim przypadku wymagana jest jedna iteracja, w której stany przeglądane są w odwrotnym porządku topologicznym (wyjaśnienie na tablicy)

Uwaga

Zwróćmy uwagę na to ci się dzieje, jeżeli $\gamma = 1$ i mamy cykl.
Dla niezerowych nagród na krawędziach cyklu wartość oczekiwana może być nieokreślona