Introducere în Reinforcement Learning

Cursul #4

Ștefan Iordache, Cătălina Iordache, Ciprian Păduraru

Cuprins

01

Recapitulare

Repetiția: mama învățăturii!

Recapitulare

- Procese Markov
- Lanțuri Markov
- Mars Rover – matrice tranziții, episoade
- MRP
- MDP

02

Politici MDP

Politici MDP

- **Politica** specifică acțiunea care trebuie luată în fiecare stare în parte.

Poate fi **determinist**ă sau **stochastic**ă.

- Pentru generalitate, se consideră a fi o ***distribuție conditionată***.

Fiind dată **o stare S, se specifică distribuția peste acțiuni.**

- **Politica:**

$$\pi(a|s) = P(a_t = a\,|s_t = s)$$

MDP + Politica

- MDP + $\boldsymbol{\pi(a|s)}$ $\Rightarrow$ Markov Reward Process (MRP)
- MRP (S, R^{π}, P^{π}, γ)
  - $R^{\pi} = \sum_{a \in A} \boldsymbol{\pi(a|s)} * R(s, a)$
  - $P^{\pi}(s'|s) = \sum_{a \in A} \boldsymbol{\pi(a|s)} * P(s'|s, a)$

PS: Totuși, putem folosi aceleași tehnici a evalua o politică pentru un MDP (algoritmul de programare dinamică).

MDP – Evaluarea politcii
- Algoritm Iterativ -

1. **Inițializăm $V_0(s) = 0, \forall s$**
2. **Pentru k = 1, până la convergență:**
    a. Pentru fiecare s din S:
        - $V_k^\pi(s) = r(s, \pi(s)) + \gamma \sum_{s' \epsilon S} P(s' | \pi(s)) V_{k-1}^\pi(s')$

- **“Bellman Backup” – se aplică unei politici particulare.**

? Întrebare rapidă ?

MDP – Evaluarea politicii

Care este diferența față de algoritmul anterior?

? Întrebare rapidă ?

MDP – Evaluarea politicii

Care este diferența față de algoritmul anterior?

I

Răspuns:

- **De această dată, se evaluează V sub o politică π.**

Exemplu MDP – Iterație a politicii de evaluare - Mars Rover

- **Dinamica:**
  - $p(s_6 \mid s_6, a_1) = 0.5$
  - $p(s_7 \mid s_7, a_1) = 0.5, \ldots$
- **Reward:**
  - +1, în starea s_1
  - +10, în starea s_7
  - 0, altfel
- **Fie: $\boldsymbol{\pi}(s) = a_1, \forall s$, presupunem că $V_k = [1\ 0\ 0\ 0\ 0\ 0\ 10]$, $k = 1$, $\gamma = 0.5$**

Exemplu MDP – Iterație a politicii de evaluare - Mars Rover – continuare

- **Fie: π(s) = a_1, ∀s, presupunem că V_k = [1 0 0 0 0 0 1 0], k= 1, γ = 0.5**
  - Pentru fiecare s din S:
    - $V_k^\pi(s) = r(s, \pi(s)) + \gamma \sum_{s' \epsilon S} P(s'|\pi(s)) V_{k-1}^\pi(s')$
  - $V_{k+1}(s_6) = r(s_6, a_1) + \gamma * 0.5 * V_k(s_6) + \gamma * 0.5 * V_k(s_7)$
  - $V_{k+1}(s_6) = 0 + 0.5 * 0.5 * 0 + \mathbf{0.5} * 0.5 * 10$
  - $V_{k+1}(s_6) = 2.5$

? Întrebări și mai rapide ?

Ipoteze:

- 7 stări discrete (locații ale roverului)
- 2 acțiuni (stânga/dreapta)

I

Q1: Câte politici deterministe există?

II

Q2: Politica optimă pentru un MDP este unică?

? Întrebare rapidă ?

Q1: Câte politici deterministe există?

I

2^7

Q2: Politica optimă pentru un MDP este unică?

II

Nu! pot exista 2 acțiuni cu aceeași valoare optimă (conformvalue function)

03

Control MDP

Politica optimă

$$\pi^*(s) = arg\ max_\pi V^\pi(s)$$

Despre controlul MDP

- **Există o singură funcție optimă de tip value function, dar multiple politici cu aceeași valoare optimă!**
- **Politica optimă pentru MDP cu orizont infinit:**
  - Este deterministă;
  - Este staționară;
  - Nu este mereu unică!

Brute-force

- Putem utiliza o metodă de tip brute-force pentru căutarea politicii optime.
- Complexitate: $|\boldsymbol{A}|^{|S|}$

Policy Iteration

1. **Setăm i = 0**
2. **Inițializăm $\pi_0(s)$ aleatorie pentru fiecare stare s.**
3. **Iterăm cât timp $||\pi_i - \pi_{i-1}|| > 0\ (L1 - norm)$:**
   a. $V^{\pi_i} \leftarrow MDP\ value\ function\ (evaluarea\ politicii\ \pi_i)$
   b. $\pi_{i+1} \leftarrow \text{Îmbunătățirea politicii}$
   c. $i \leftarrow i + 1$

04

În căutarea politicii

State-Action Value -> Q

- **Avansăm cu pași repezi și ajungem la ceea ce vom numi drept „State-Action Value” (Q).**

$$Q^{\pi}(s,a) = R(s,a) + \gamma \sum_{s' \in S} P(s'|s,a) V^{\pi}(s')$$

- **Traducere: În starea s, executăm acțiunea a, apoi urmăm politica π.**

Cum actualizăm? Policy Iteration

- **Calculăm Q pentru o politică π_i pentru fiecare stare s din S și fiecare acțiune a din A.**

$$Q^{\pi_i}(s,a) = R(s,a) + \gamma \sum_{s' \in S} P(s'|s,a) V^{\pi_i}(s')$$

- **Generăm politica nouă π_{i+1} pentru fiecare stare s din S.**

$$\pi_{i+1}(s) = \arg\max_a Q^{\pi_i}(s,a)$$

Aprofundare!

- **Îmbunătățirea politicii are loc la fiecare iterație!**

$$Q^{\pi_i}(s,a) = R(s,a) + \gamma \sum_{s' \in S} P(s'|s,a) V^{\pi_i}(s')$$

$$\max_a Q^{\pi_i}(s,a) \geq R(s,\pi_i(s)) + \gamma \sum_{s' \in S} P(s'|s,\pi_i(s)) V^{\pi_i}(s') = V^{\pi_i}(s)$$

$$\pi_{i+1}(s) = \arg\max_a Q^{\pi_i}(s,a)$$

Cum demonstrăm algoritmul?

- **Presupunere:**
  - $V^{\pi_{i+1}} \geq V^{\pi_i}$

unde π_i $nu\ este\ optimă, iar\ \pi_{i+1}\ este\ noua\ politică\ obținută\ prin\ îmbunătățirea\ \pi_i$

$$
\begin{aligned}
V^{\pi_i}(s) &\leq \max_a Q^{\pi_i}(s, a) \\
&= \max_a R(s, a) + \gamma \sum_{s' \in S} P(s'|s, a) V^{\pi_i}(s') \\
&= R(s, \pi_{i+1}(s)) + \gamma \sum_{s' \in S} P(s'|s, \pi_{i+1}(s)) V^{\pi_i}(s') \text{ //by the definition of } \pi_{i+1} \\
&\leq R(s, \pi_{i+1}(s)) + \gamma \sum_{s' \in S} P(s'|s, \pi_{i+1}(s)) \left(\max_{a'} Q^{\pi_i}(s', a') \right) \\
&= R(s, \pi_{i+1}(s)) + \gamma \sum_{s' \in S} P(s'|s, \pi_{i+1}(s)) \\
&\quad \left(R(s', \pi_{i+1}(s')) + \gamma \sum_{s'' \in S} P(s''|s', \pi_{i+1}(s')) V^{\pi_i}(s'') \right) \\
&\quad \vdots \\
&= V^{\pi_{i+1}}(s)
\end{aligned}
$$

Ce am aflat?

- **Policy Iteration** ne ajută să ajungem atât la valori optime, cât și la politici asociate!

Value Iteration

$$V^\pi(s) = R^\pi(s) + \gamma \sum_{s' \in S} P^\pi(s'|s) V^\pi(s')$$

- **Introducem un nou concept: „Bellman Backup Operator”**
  - Se aplică asupra functiei V.
  - Ne returnează o nouă funcție V.
  - Îmbunătățește valoarea dacă este posibil acest lucru.

$$BV(s) = \max_a R(s,a) + \gamma \sum_{s' \in S} p(s'|s,a) V(s')$$

Aprofundare

- **Cum aplicăm operațiunile de tip Bellman asupra unei politici?**

$$B^\pi V(s) = R^\pi(s) + \gamma \sum_{s' \in S} P^\pi(s'|s)V(s)$$

- **Cum evaluăm o politică? Repetat!, până când valoarea V nu se mai schimbă și nu observăm creșteri sau scăderi.**

$$V^\pi = B^\pi B^\pi \cdots B^\pi V$$

Value Iteration
- Algoritm Iterativ -

1. **Setăm k = 1**
2. **Inițializăm $V_0(s) = 0, \forall s$**
3. **Repetăm până la convergență:**
   a. Pentru fiecare s din S:

$$V_{k+1}(s) = \max_a R(s,a) + \gamma \sum_{s' \in S} P(s'|s,a) V_k(s')$$

Avem condiții de convergență?

- ***Da! Dacă factorul de discount este mai mic strict decât 1!*** $\gamma < 1$

Thanks!

Este timpul pentru întrebări!!!

stefan.iordache10@s.unibuc.ro
catalina.patilea@s.unibuc.ro
ciprian.paduraru@fmi.unibuc.ro

+40 7.. ... ...