

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania

Uczenie ze wzmocnieniem — generalizacja i zastosowania

Na podstawie: AIMA ch21 oraz Reinforcement Learning (Sutton i Barto)

Wojciech Jaśkowski

Instytut Informatyki,
Politechnika Poznańska

15 maja 2015

Uczenie ze wzmocnieniem — generalizacja i zastosowania
Na podstawie: AIMA ch21 oraz Reinforcement Learning (Sutton i Barto)

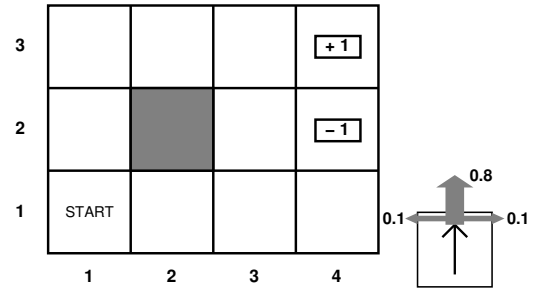
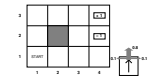
Wojciech Jaśkowski
Instytut Informatyki,
Politechnika Poznańska
15 maja 2015

Problem decyzyjny Markova

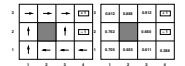
2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania
└ Przypomnienie

└ Problem decyzyjny Markova



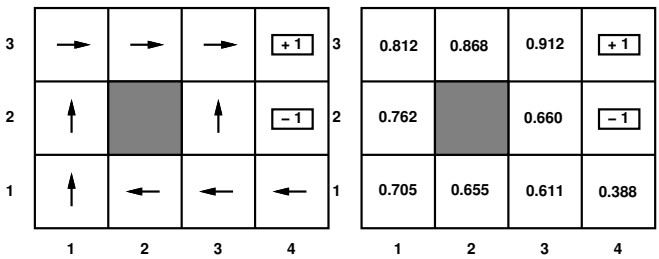
Rozwiązanie problemu decyzyjnego Markova



2015-05-15

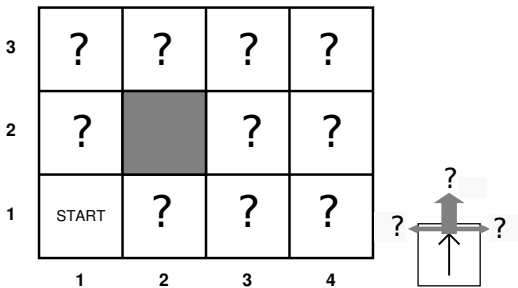
Uczenie ze wzmocnieniem — generalizacja i zastosowania
└ Przypomnienie

└ Rozwiązanie problemu decyzyjnego Markova



nieznany MDP

brak f. nagrody i modelu przejść

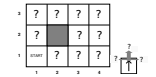


2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania

└ Przypomnienie

└nieznany MDP



1. Co jest gorsze? Brak f. nagrody, czy modelu przejść? Raczej: brak modelu przejść.

Uczenie ze wzmocnieniem

Uczenie:

- **pasywne (problem predykcji)** → ocena użyteczności danej polityki π
- **aktywne (problem sterowania)** → znalezienie optymalnej polityki π
 - eksploracja!

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania

└ Przypomnienie

└ Uczenie ze wzmocnieniem

Uczenie:

- **pasywne (problem predykcji)** → ocena użyteczności danej polityki π
- **aktywne (problem sterowania)** → znalezienie optymalnej polityki π
 - eksploracja!

Rodzaje uczenia

1. **Uczenie nadzorowane** (nauczyciel)

- 1. **klasyfikacja:** $stan \rightarrow [znana!] \text{ klasa decyzyjna}$
- 2. **regresja:** $stan \rightarrow [znana!] \text{ wartość}$

2. **Uczenie ze wzmocnieniem** (krytyk)

- 1. $stan \rightarrow [nieznana!] \text{ klasa (akcja), ale wzmocnienie.}$

3. **Uczenie nienadzorowane** (brak nauczyciela)

- 1. $stan \rightarrow [nieznana!] \text{ klasa}$

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania

Przypomnienie

Rodzaje uczenia

1. Stan jest zwykle opisywany atrybutami, cechami (np. typowa „tabelka” w problemie klasyfikacji)

- **Uczenie nadzorowane** (nauczyciel)
 - **klasyfikacja:** $stan \rightarrow [znana!] \text{ klasa decyzyjna}$
 - **regresja:** $stan \rightarrow [znana!] \text{ wartość}$
- **Uczenie ze wzmocnieniem** (krytyk)
 - $stan \rightarrow [nieznana!] \text{ klasa (akcja), ale wzmocnienie.}$
- **Uczenie nienadzorowane** (brak nauczyciela)
 - $stan \rightarrow [nieznana!] \text{ klasa}$

Podjęcia do uczenia ze wzmocnieniem

Równanie Bellman'a:

$$U(s) = R(s) + \gamma \max_{a \in A(s)} \sum_{s'} U(s') P(s'|s, a)$$

Podjęcia:

- 1 **agent odruchowy** (*direct policy search*)
 - uczy się polityki $\pi : S \rightarrow A$
 - np. algorytm ewolucyjny
- 2 **agent z funkcją użyteczności U**
 - uczy się $U(s)$
 - np. **adaptatywne programowanie dynamiczne (ADP)**,
metoda różnic czasowych (TDL)
- 3 **agent z funkcją Q**
 - uczy się $Q(s, a)$
 - np. **metody różnic czasowych** (Q-learning, SARSA)

Który agent potrzebuje modelu świata?[\[zadanie 1\]](#)

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania

Przypomnienie

Podjęcia do uczenia ze wzmocnieniem

1. Terazniejszość $R(s)$ + oczekiwana zdyskontowana przyszłość.
2. która mapuje bezpośrednio stany na akcje.
3. musi posiadać model środowiska, żeby podejmować decyzje.
4. nie musi posiadać modelu środowiska

Podjęcia do uczenia ze wzmocnieniem

Równanie Bellman'a:

$$U(s) = R(s) + \gamma \max_{a \in A(s)} \sum_{s'} U(s') P(s'|s, a)$$

Podjęcia:

- 1 **agent odruchowy** (*direct policy search*)
 - uczy się polityki $\pi : S \rightarrow A$
 - np. algorytm ewolucyjny
- 2 **agent z funkcją użyteczności U**
 - uczy się $U(s)$
 - np. **adaptatywne programowanie dynamiczne (ADP)**,
metoda różnic czasowych (TDL)
- 3 **agent z funkcją Q**
 - uczy się $Q(s, a)$
 - np. **metody różnic czasowych** (Q-learning, SARSA)

Który agent potrzebuje modelu świata?[\[zadanie 1\]](#)

Reguły uczenia

TD-Learning

$$U^\pi(s) \leftarrow U^\pi(s) + \alpha (R(s) + \gamma U^\pi(s') - U^\pi(s))$$

- α — współczynnik uczenia

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania

- └ Przypomnienie

- └ Reguły uczenia

Reguły uczenia

TD-Learning

$$U^\pi(s) \leftarrow U^\pi(s) + \alpha (R(s) + \gamma U^\pi(s') - U^\pi(s))$$

- α — współczynnik uczenia

Reguły uczenia

TD-Learning

$$U^\pi(s) \leftarrow U^\pi(s) + \alpha (R(s) + \gamma U^\pi(s') - U^\pi(s))$$

- α — współczynnik uczenia

Q-Learning

$$Q(s, a) \leftarrow Q(s, a) + \alpha (R(s) + \gamma \max_{a'} Q(s', a') - Q(s, a))$$

- α — współczynnik uczenia

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania

- └ Przypomnienie

- └ Reguły uczenia

Reguły uczenia

TD-Learning

$$U^\pi(s) \leftarrow U^\pi(s) + \alpha (R(s) + \gamma U^\pi(s') - U^\pi(s))$$

- α — współczynnik uczenia

Q-Learning

$$Q(s, a) \leftarrow Q(s, a) + \alpha (R(s) + \gamma \max_{a'} Q(s', a') - Q(s, a))$$

- α — współczynnik uczenia

Approksymator funkcji

Przestrzeń stanów:

- ADP działa rozsądnie dla problemów wielkości rzędu 10000 stanów.
 - tryktrak (backgammon): 10^{20}
 - szachy: 10^{40}
- Nie da się explicitie rozważać tylu stanów

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania

- └ Generalizacja

- └└ Approksymator funkcji

1. (jeśli dobrze się zaimplementuje),
2. Ale wcale nie musi być liniowa

Approksymator funkcji

Przestrzeń stanów:

- ADP działa rozsądnie dla problemów wielkości rzędu 10000 stanów.
 - tryktrak (backgammon): 10^{20}
 - szachy: 10^{40}
- Nie da się explicitie rozważać tylu stanów

Apoksymator funkcji

Przestrzeń stanów:

- ADP działa rozsądnie dla problemów wielkości rzędu 10000 stanów.
 - tryktrak (backgammon): 10^{20}
 - szachy: 10^{40}
- Nie da się explicite rozważać tylu stanów

Aproksymator funkcji:

- Nietablicowa, **parametryczna**, funkcja stanu lub pary stan-akcja.
- Przykład:
 - Stan reprezentowany jako cechy f_1, f_2, \dots, f_n .
 - Aproksymator funkcji \hat{U}_θ to ich liniowa kombinacja:

$$\hat{U}_\theta(s) = \theta_1 f_1(s) + \theta_2 f_2(s) + \dots + \theta_n f_n(s)$$

- Uczymy się tylko wartości parametrów $\theta = (\theta_1, \theta_2, \dots, \theta_n)$.

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania

- └ Generalizacja

- └ Aproksymator funkcji

1. (jeśli dobrze się zaimplementuje),
2. Ale wcale nie musi być liniowa

Aproksymator funkcji

Przestrzeń stanów:

- ADP działa rozsądnie dla problemów wielkości rzędu 10000 stanów.
 - tryktrak (backgammon): 10^{20}
 - szachy: 10^{40}
- Nie da się explicite rozważać tylu stanów

Aproksymator funkcji:

- Nietablicowa, **parametryczna**, funkcja stanu lub pary stan-akcja.

Przykład:

- Stan reprezentowany jako cechy f_1, f_2, \dots, f_n .
- Aproksymator funkcji \hat{U} , to ich liniowa kombinacja:

$$\hat{U}(s) = \theta_1 f_1(s) + \theta_2 f_2(s) + \dots + \theta_n f_n(s)$$

- Uczymy się tylko wartości parametrów $\theta = (\theta_1, \theta_2, \dots, \theta_n)$.

Przykład



$$\hat{U}_{\theta}(s) = \theta_1 \text{pionków}(s) + \theta_2 \text{figur_w_centrum}(s) + \theta_3 \text{hetman?}(s) + \theta_4 \text{szach?}(s)$$

10^{40} stanów \rightarrow 4 parametry

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania

- Generalizacja

- Przykład

Przykład



$$Q_i(s) = \theta_1 \text{pionków}(s) + \theta_2 \text{figur_w_centrum}(s) + \theta_3 \text{hetman?}(s) + \theta_4 \text{szach?}(s)$$

10^{40} stanów \rightarrow 4 parametry

Aproksymator funkcji

Właściwości:

- 1 musi być łatwo obliczalny,
 - Cechy oparte na np. przeszukiwaniu Minimax'em są dobre, ale kosztowne obliczeniowo. Potrzebujemy prostych cech.

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania
└ Generalizacja

└ Aproksymator funkcji

Aproksymator funkcji

Właściwości:

- musi być łatwo obliczalny.
- Cechy oparte na np. przeszukiwaniu Minimax'em są dobre, ale kosztowne obliczeniowo. Potrzebujemy prostych cech.

Aproksymator funkcji

Właściwości:

- 1 musi być łatwo obliczalny,
 - Cechy oparte na np. przeszukiwaniu Minimax'em są dobre, ale kosztowne obliczeniowo. Potrzebujemy prostych cech.
- 2 „**kompresuje**” (dużą) przestrzeń stanów w (małą) liczbę parametrów,
- 3 \implies **uogólniania wiedzę** (stany odwiedzone vs. nieodwiedzone),
 - Przykład: co 10^{12} stan \rightarrow „mistrzowski” gracz w tryktraka

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania

Generalizacja

Aproksymator funkcji

Aproksymator funkcji

Właściwości:

- 1 musi być łatwo obliczalny,
 - Cechy oparte na np. przeszukiwaniu Minimax'em są dobre, ale kosztowne obliczeniowo. Potrzebujemy prostych cech.
- 2 „**kompresuje**” (dużą) przestrzeń stanów w (małą) liczbę parametrów,
- 3 \implies **uogólniania wiedzę** (stany odwiedzone vs. nieodwiedzone),
 - Przykład: co 10^{12} stan \rightarrow „mistrzowski” gracz w tryktraka

Aproksymator funkcji

Właściwości:

- 1 musi być łatwo obliczalny,
 - Cechy oparte na np. przeszukiwaniu Minimax'em są dobre, ale kosztowne obliczeniowo. Potrzebujemy prostych cech.
- 2 „**kompresuje**” (dużą) przestrzeń stanów w (małą) liczbę parametrów,
- 3 \implies **uogólniania wiedzę** (stany odwiedzone vs. nieodwiedzone),
 - Przykład: co 10^{12} stan \rightarrow „mistrzowski” gracz w tryktraka
- 4 Kompromis: wielkość przestrzeni (jakość aproksymacji) vs. czas nauki / pamięć.

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania

└ Generalizacja

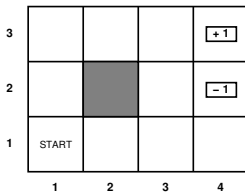
└ Aproksymator funkcji

Właściwości:

- musi być łatwo obliczalny.
 - Cechy oparte na np. przeszukiwaniu Minimax'em są dobre, ale kosztowne obliczeniowo. Potrzebujemy prostych cech.
- „**kompresuje**” (dużą) przestrzeń stanów w (małą) liczbę parametrów.
- \implies **uogólniania wiedzę** (stany odwiedzone vs. nieodwiedzone).
 - Przykład: co 10^{12} stan \rightarrow „mistrzowski” gracz w tryktraka
- Kompromis: wielkość przestrzeni (jakość aproksymacji) vs. czas nauki / pamięć.

Aproksymacja funkcji

Bezpośrednia estymacja użyteczności



Przykład. Niech:

$$\hat{U}_{\theta}(x, y) = \theta_0 + \theta_1 x + \theta_2 y$$

- Jeśli $\theta = (0.5, 0.2, 0.1)$, to ile wynosi $\hat{U}_{\theta}(1, 2)$? [zadanie 2]

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania

Generalizacja

Aproksymacja funkcji

1. $\hat{U}_{\theta}(1, 1) = 0.5 + 0.2 + 0.2 = 0.9$
2. $u(1, 2)$ jest sumarycznym zdyskontowanym wzmocnieniem.



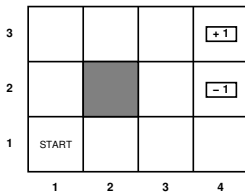
Przykład. Niech:

$$\hat{U}_{\theta}(x, y) = \theta_0 + \theta_1 x + \theta_2 y$$

- Jeśli $\theta = (0.5, 0.2, 0.1)$, to ile wynosi $\hat{U}_{\theta}(1, 2)$? [zadanie 2]

Aproksymacja funkcji

Bezpośrednia estymacja użyteczności



Przykład. Niech:

$$\hat{U}_\theta(x, y) = \theta_0 + \theta_1 x + \theta_2 y$$

- Jeśli $\theta = (0.5, 0.2, 0.1)$, to ile wynosi $\hat{U}_\theta(1, 2)$? [zadanie 2]
- Wykonaliśmy próbę od stanu (1, 2) i otrzymaliśmy sumaryczne wzmocnienie $u(1, 2) = 0.5$.
- **Wniosek:** $\hat{U}_\theta(1, 2) = 0.9$ to za dużo.

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania

Generalizacja

Aproksymacja funkcji

1. $\hat{U}_\theta(1, 1) = 0.5 + 0.2 + 0.2 = 0.9$
2. $u(1, 2)$ jest sumarycznym zdyskontowanym wzmocnieniem.



Przykład. Niech:

$$\hat{U}_\theta(x, y) = \theta_0 + \theta_1 x + \theta_2 y$$

- Jeśli $\theta = (0.5, 0.2, 0.1)$, to ile wynosi $\hat{U}_\theta(1, 2)$? [zadanie 2]
- Wykonaliśmy próbę od stanu (1, 2) i otrzymaliśmy sumaryczne wzmocnienie $u(1, 2) = 0.5$.
- **Wniosek:** $\hat{U}_\theta(1, 2) = 0.9$ to za dużo.

Uczenie offline

Możemy więc zebrać n par $\langle \hat{U}_\theta(s_i), u(s_i) \rangle$ i rozwiązać problem regresji minimalizując średni błąd kwadratowy:

$$\theta^* = \operatorname{argmin}_\theta \sum_{i=1}^n \left(\hat{U}_\theta(s_i) - u(s_i) \right)^2.$$

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania

└ Generalizacja

└ Uczenie offline

1. Np. regresja liniowa jest prosta obliczeniowo.

$$\theta^* = \operatorname{argmin}_\theta \sum_{i=1}^n \left(\hat{U}_\theta(s_i) - u(s_i) \right)^2.$$

Uczenie offline

Możemy więc zebrać n par $\langle \hat{U}_\theta(s_i), u(s_i) \rangle$ i rozwiązać problem regresji minimalizując średni błąd kwadratowy:

$$\theta^* = \operatorname{argmin}_\theta \sum_{i=1}^n \left(\hat{U}_\theta(s_i) - u(s_i) \right)^2.$$

Ale to zadziała tylko dla problemu predykcji. Dla problemu sterowania lepiej uczyć się **online** np. po każdej próbkce.

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania

Generalizacja

Uczenie offline

1. Np. regresja liniowa jest prosta obliczeniowo.

Możemy więc zebrać n par $\langle \hat{U}_\theta(s_i), u(s_i) \rangle$ i rozwiązać problem regresji minimalizując średni błąd kwadratowy:

$$\theta^* = \operatorname{argmin}_\theta \sum_{i=1}^n \left(\hat{U}_\theta(s_i) - u(s_i) \right)^2.$$

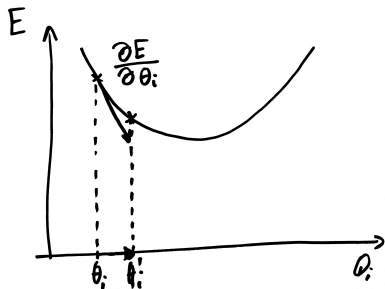
Ale to zadziała tylko dla problemu predykcji. Dla problemu sterowania lepiej uczyć się **online** np. po każdej próbkce.

Reguła Widrow-Hoff'a

Bezpośrednia estymacja użyteczności

Niech funkcja straty:

$$E(s, \theta) = \frac{1}{2} (\hat{U}_{\theta}(s) - u(s))^2$$



Szukamy takich parametrów, które minimalizują błąd (metoda gradientu prostego):

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania

Generalizacja

Reguła Widrow-Hoff'a

1. Dlaczego $\frac{1}{2}$ w funkcji błędu? Żeby się skróciły stałe przy różniczkowaniu. Generalnie, to nie ma znaczenia, bo jest α
2. α jest współczynnikiem, który mówi, jak bardzo chcemy się zbliżyć do celu. Jeśli α będzie maleć w czasie, zbiegniemy do optimum.

Niech funkcja straty:

$$E(s, \theta) = \frac{1}{2} (\hat{U}_{\theta}(s) - u(s))^2$$

Szukamy takich parametrów, które minimalizują błąd (metoda gradientu prostego):

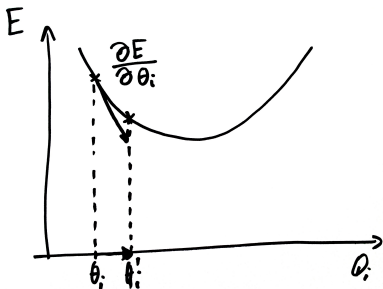


Reguła Widrow-Hoff'a

Bezpośrednia estymacja użyteczności

Niech funkcja straty:

$$E(s, \theta) = \frac{1}{2} (\hat{U}_{\theta}(s) - u(s))^2$$



Szukamy takich parametrów, które minimalizują błąd (metoda gradientu prostego):

$$\begin{aligned} \theta_i &\leftarrow \theta_i - \alpha \frac{\partial E(s, \theta)}{\partial \theta_i} = \theta_i - \alpha \frac{\partial \left(\frac{1}{2} (\hat{U}_{\theta}(s) - u(s))^2 \right)}{\partial \theta_i} \\ &= \theta_i + \alpha (u(s) - \hat{U}_{\theta}(s)) \frac{\partial \hat{U}_{\theta}(s)}{\partial \theta_i}. \end{aligned}$$

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania

Generalizacja

Reguła Widrow-Hoff'a

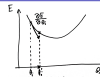
1. Dlaczego $\frac{1}{2}$ w funkcji błędów? Żeby się skróciły stałe przy różniczkowaniu. Generalnie, to nie ma znaczenia, bo jest α
2. α jest współczynnikiem, który mówi, jak bardzo chcemy się zbliżyć do celu. Jeśli α będzie maleć w czasie, zbiegniemy do optimum.

Niech funkcja straty:

$$E(s, \theta) = \frac{1}{2} (\hat{U}_{\theta}(s) - u(s))^2$$

Szukamy takich parametrów, które minimalizują błąd (metoda gradientu prostego):

$$\begin{aligned} \theta_i &\leftarrow \theta_i - \alpha \frac{\partial E(s, \theta)}{\partial \theta_i} = \theta_i - \alpha \frac{\partial \left(\frac{1}{2} (\hat{U}_{\theta}(s) - u(s))^2 \right)}{\partial \theta_i} \\ &= \theta_i + \alpha (u(s) - \hat{U}_{\theta}(s)) \frac{\partial \hat{U}_{\theta}(s)}{\partial \theta_i}. \end{aligned}$$



Przykład

Bezpośrednia estymacja użyteczności

$$\theta_i \leftarrow \theta_i + \alpha \left(u(s) - \hat{U}_\theta(s) \right) \frac{\partial \hat{U}_\theta(s)}{\partial \theta_i}$$

Przykład dla 4x3:

$$\hat{U}_\theta(x, y) = \theta_0 + \theta_1 x + \theta_2 y,$$

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania
└ Generalizacja

└ Przykład

Przykład

Bezpośrednia estymacja użyteczności

$$\theta_i \leftarrow \theta_i + \alpha \left(u(s) - \hat{U}_i(s) \right) \frac{\partial \hat{U}_i(s)}{\partial \theta_i}$$

Przykład dla 4x3:

$$\hat{U}_i(x, y) = \theta_0 + \theta_1 x + \theta_2 y,$$

Przykład

Bezpośrednia estymacja użyteczności

$$\theta_i \leftarrow \theta_i + \alpha \left(u(s) - \hat{U}_\theta(s) \right) \frac{\partial \hat{U}_\theta(s)}{\partial \theta_i}$$

Przykład dla 4x3:

$$\hat{U}_\theta(x, y) = \theta_0 + \theta_1 x + \theta_2 y,$$

więc:

$$\theta_0 \leftarrow \theta_0 + \alpha (u(s) - \hat{U}_\theta(s))$$

$$\theta_1 \leftarrow \theta_1 + \alpha (u(s) - \hat{U}_\theta(s))x$$

$$\theta_2 \leftarrow \theta_2 + \alpha (u(s) - \hat{U}_\theta(s))y$$

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania

- Generalizacja

└ Przykład

$$\theta_i \leftarrow \theta_i + \alpha \left(u(s) - \hat{U}_\theta(s) \right) \frac{\partial \hat{U}_\theta(s)}{\partial \theta_i}$$

Przykład dla 4x3:

$$\hat{U}_\theta(x, y) = \theta_0 + \theta_1 x + \theta_2 y,$$

więc:

$$\theta_0 \leftarrow \theta_0 + \alpha (u(s) - \hat{U}_\theta(s))$$

$$\theta_1 \leftarrow \theta_1 + \alpha (u(s) - \hat{U}_\theta(s))x$$

$$\theta_2 \leftarrow \theta_2 + \alpha (u(s) - \hat{U}_\theta(s))y$$

Przykład

Bezpośrednia estymacja użyteczności

$$\theta_0 \leftarrow \theta_0 + \alpha(u(s) - \hat{U}_\theta(s))$$

$$\theta_1 \leftarrow \theta_1 + \alpha(u(s) - \hat{U}_\theta(s))x$$

$$\theta_2 \leftarrow \theta_2 + \alpha(u(s) - \hat{U}_\theta(s))y$$

Niech:

- $(\theta_0, \theta_1, \theta_2) = (0.5, 0.2, 0.1)$
- $u(1, 2) = 0.5$

Pytania:

1. Ile będą wynosić wartości parametrów $(\theta_0, \theta_1, \theta_2)$ po aktualizacji ($\alpha = 0.25$)? [zadanie 3]

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania

Generalizacja

Przykład

1. $\alpha\delta = 0.25 \times 0.4 = 0.1$. więc
 $\theta = (0.5 - 0.1, 0.2 - 0.1 \times 1, 0.1 - 0.1 \times 2) = (0.4, 0.1, -0.1)$
2. Wtedy $\hat{U}_\theta(1, 2) = 0.4 + 0.1 \times 1 - 0.1 \times 2 = 0.3$
3. Zauważmy przy okazji: wartość aktualna wynosiła 0.9 a docelowa 0.5, ale ostatecznie ją przeskoczyliśmy (0.3). Wniosek: zbyt duży współczynnik α .
4. Tak! I to jest istota **generalizowania!**

$$\theta_0 \leftarrow \theta_0 + \alpha(u(s) - \hat{U}_\theta(s))$$

$$\theta_1 \leftarrow \theta_1 + \alpha(u(s) - \hat{U}_\theta(s))x$$

$$\theta_2 \leftarrow \theta_2 + \alpha(u(s) - \hat{U}_\theta(s))y$$

Niech:

- $(\theta_0, \theta_1, \theta_2) = (0.5, 0.2, 0.1)$
- $u(1, 2) = 0.5$

Pytania:

- Ile będą wynosić wartości parametrów $(\theta_0, \theta_1, \theta_2)$ po aktualizacji ($\alpha = 0.25$)? [zadanie 3]

Przykład

Bezpośrednia estymacja użyteczności

$$\theta_0 \leftarrow \theta_0 + \alpha(u(s) - \hat{U}_\theta(s))$$

$$\theta_1 \leftarrow \theta_1 + \alpha(u(s) - \hat{U}_\theta(s))x$$

$$\theta_2 \leftarrow \theta_2 + \alpha(u(s) - \hat{U}_\theta(s))y$$

Niech:

- $(\theta_0, \theta_1, \theta_2) = (0.5, 0.2, 0.1)$
- $u(1, 2) = 0.5$

Pytania:

1. Ile będą wynosić wartości parametrów $(\theta_0, \theta_1, \theta_2)$ po aktualizacji ($\alpha = 0.25$)? [zadanie 3]
2. Ile wyniesie $\hat{U}_\theta(1, 2)$ po aktualizacji parametrów? [zadanie 4]

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania

Generalizacja

Przykład

1. $\alpha\delta = 0.25 \times 0.4 = 0.1$. więc
 $\theta = (0.5 - 0.1, 0.2 - 0.1 \times 1, 0.1 - 0.1 \times 2) = (0.4, 0.1, -0.1)$
2. Wtedy $\hat{U}_\theta(1, 2) = 0.4 + 0.1 \times 1 - 0.1 \times 2 = 0.3$
3. Zauważmy przy okazji: wartość aktualna wynosiła 0.9 a docelowa 0.5, ale ostatecznie ją przeskoczyliśmy (0.3). Wniosek: zbyt duży współczynnik α .
4. Tak! I to jest istota **generalizowania!**

$$\theta_0 \leftarrow \theta_0 + \alpha(u(s) - \hat{U}_\theta(s))$$

$$\theta_1 \leftarrow \theta_1 + \alpha(u(s) - \hat{U}_\theta(s))x$$

$$\theta_2 \leftarrow \theta_2 + \alpha(u(s) - \hat{U}_\theta(s))y$$

Niech:

- $(\theta_0, \theta_1, \theta_2) = (0.5, 0.2, 0.1)$
- $u(1, 2) = 0.5$

Pytania:

- Ile będą wynosić wartości parametrów $(\theta_0, \theta_1, \theta_2)$ po aktualizacji ($\alpha = 0.25$)? [zadanie 3]
- Ile wyniesie $\hat{U}_\theta(1, 2)$ po aktualizacji parametrów? [zadanie 4]

Przykład

Bezpośrednia estymacja użyteczności

$$\theta_0 \leftarrow \theta_0 + \alpha(u(s) - \hat{U}_\theta(s))$$

$$\theta_1 \leftarrow \theta_1 + \alpha(u(s) - \hat{U}_\theta(s))x$$

$$\theta_2 \leftarrow \theta_2 + \alpha(u(s) - \hat{U}_\theta(s))y$$

Niech:

- $(\theta_0, \theta_1, \theta_2) = (0.5, 0.2, 0.1)$
- $u(1, 2) = 0.5$

Pytania:

1. Ile będą wynosić wartości parametrów $(\theta_0, \theta_1, \theta_2)$ po aktualizacji ($\alpha = 0.25$)? [zadanie 3]
2. Ile wyniesie $\hat{U}_\theta(1, 2)$ po aktualizacji parametrów? [zadanie 4]
3. Chcieliśmy, aby $\hat{U}_\theta(1, 2)$ się zmieniło. Czy zmieniło się także $\hat{U}_\theta(1, 1)$? [zadanie 5]

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania

Generalizacja

Przykład

1. $\alpha\delta = 0.25 \times 0.4 = 0.1$. więc
 $\theta = (0.5 - 0.1, 0.2 - 0.1 \times 1, 0.1 - 0.1 \times 2) = (0.4, 0.1, -0.1)$
2. Wtedy $\hat{U}_\theta(1, 2) = 0.4 + 0.1 \times 1 - 0.1 \times 2 = 0.3$
3. Zauważmy przy okazji: wartość aktualna wynosiła 0.9 a docelowa 0.5, ale ostatecznie ją przeskoczyliśmy (0.3). Wniosek: zbyt duży współczynnik α .
4. Tak! I to jest istota **generalizowania!**

Przykład
Bezpośrednia estymacja użyteczności

$$\theta_0 \leftarrow \theta_0 + \alpha(u(s) - \hat{U}_\theta(s))$$

$$\theta_1 \leftarrow \theta_1 + \alpha(u(s) - \hat{U}_\theta(s))x$$

$$\theta_2 \leftarrow \theta_2 + \alpha(u(s) - \hat{U}_\theta(s))y$$

Niech:

- $(\theta_0, \theta_1, \theta_2) = (0.5, 0.2, 0.1)$
- $u(1, 2) = 0.5$

Pytania:

- Ile będą wynosić wartości parametrów $(\theta_0, \theta_1, \theta_2)$ po aktualizacji ($\alpha = 0.25$)? [zadanie 3]
- Ile wyniesie $\hat{U}_\theta(1, 2)$ po aktualizacji parametrów? [zadanie 4]
- Chcieliśmy, aby $\hat{U}_\theta(1, 2)$ się zmieniło. Czy zmieniło się także $\hat{U}_\theta(1, 1)$? [zadanie 5]

Wybór aproksymatora — wiedza dziedzinowa

Generalizacja

Agent uczy się szybciej z aproksymatorem funkcji, bo może **generalizować**.

3				+1
2				-1
1	START			
	1	2	3	4

- Aproksymator funkcji

$$\hat{U}_\theta(x, y) = \theta_0 + \theta_1 x + \theta_2 y,$$

działa dobrze dla świata 10×10 z nagrodą +1 w (10, 10).

- A jak zadziała gdy +1 będzie w polu (5, 5)? [zadanie 6]

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania

Generalizacja

Wybór aproksymatora — wiedza dziedzinowa

Generalizacja

Agent uczy się szybciej z aproksymatorem funkcji, bo może **generalizować**.

3				
2				
1	START			
	1	2	3	4

- Aproksymator funkcji

$$\hat{U}_\theta(x, y) = \theta_0 + \theta_1 x + \theta_2 y,$$

działa dobrze dla świata 10×10 z nagrodą +1 w (10, 10).

• A jak zadziała gdy +1 będzie w polu (5, 5)? [zadanie 6]

1. Porażka. Dla świata 10×10 funkcja użyteczności jest liniowa względem wymiarów, ale gdy +1 jest w polu (5, 5) przypomina piramidę.
2. Zauważmy: cechy mogą być dowolnymi nieliniowymi elementami (np. liczba pionków, hetmanów, etc.)

Wybór aproksymatora — wiedza dziedzinowa

Generalizacja

Agent uczy się szybciej z aproksymatorem funkcji, bo może **generalizować**.

3				+1
2				-1
1	START			
	1	2	3	4

- Aproksymator funkcji

$$\hat{U}_\theta(x, y) = \theta_0 + \theta_1 x + \theta_2 y,$$

działa dobrze dla świata 10×10 z nagrodą $+1$ w $(10, 10)$.

- A jak zadziała gdy $+1$ będzie w polu $(5, 5)$? [zadanie 6]
- **Wiedza dziedzinowa**: dodać do $\hat{U}_\theta(x, y)$ składnik $\theta_3 f_3$:

$$f_3 = \sqrt{(x - 5)^2 + (y - 5)^2}$$

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania

Generalizacja

Wybór aproksymatora — wiedza dziedzinowa

Wybór aproksymatora — wiedza dziedzinowa

Generalizacja
Agent uczy się szybciej z aproksymatorem funkcji, bo może **generalizować**.

Aproksymator funkcji
 $\hat{U}_\theta(x, y) = \theta_0 + \theta_1 x + \theta_2 y,$
działa dobrze dla świata 10×10 z nagrodą $+1$ w $(10, 10)$.
A jak zadziała gdy $+1$ będzie w polu $(5, 5)$? [zadanie 6]
Wiedza dziedzinowa: dodać do $\hat{U}_\theta(x, y)$ składnik $\theta_3 f_3$.
 $f_3 = \sqrt{(x - 5)^2 + (y - 5)^2}$

1. Porażka. Dla świata 10×10 funkcja użyteczności jest liniowa względem wymiarów, ale gdy $+1$ jest w polu $(5, 5)$ przypomina piramidę.
2. Zauważmy: cechy mogą być dowolnymi nieliniowymi elementami (np. liczba pionków, hetmanów, etc.)

Metoda różnic czasowych (funkcja użyteczności)

Wersja oryginalna

$$U^\pi(s) \leftarrow U^\pi(s) + \alpha (R(s) + \gamma U^\pi(s') - U^\pi(s))$$

Z aproksymatorem funkcji

$$\theta_i \leftarrow \theta_i + \alpha \left(R(s) + \gamma \hat{U}_\theta(s') - \hat{U}_\theta(s) \right) \frac{\partial \hat{U}_\theta(s)}{\partial \theta_i}$$

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania

Generalizacja

Metoda różnic czasowych (funkcja użyteczności)

1. Zwróćmy uwagę: tutaj wartością **docelową** (target) jest zaznaczone na czerwono wyrażenie (wartość po wykonaniu jednego kroku), a nie $u(1, 2)$ — wartość po całym epizodzie tak jak w metodzie Monte Carlo.

Wersja oryginalna

$$U^\pi(s) \leftarrow U^\pi(s) + \alpha (R(s) + \gamma U^\pi(s') - U^\pi(s))$$

Z aproksymatorem funkcji

$$\theta_i \leftarrow \theta_i + \alpha \left(R(s) + \gamma \hat{U}_\theta(s') - \hat{U}_\theta(s) \right) \frac{\partial \hat{U}_\theta(s)}{\partial \theta_i}$$

Q-learning

Wersja oryginalna

$$Q(s, a) \leftarrow Q(s, a) + \alpha (R(s) + \gamma \max_{a'} Q(s', a') - Q(s, a))$$

Z aproksymatorem funkcji

$$\theta_i \leftarrow \theta_i + \alpha \left(R(s) + \gamma \max_{a'} \hat{Q}_\theta(s', a') - \hat{Q}_\theta(s, a) \right) \frac{\partial \hat{Q}_\theta(s, a)}{\partial \theta_i}$$

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania

Generalizacja

Q-learning

Q-learning

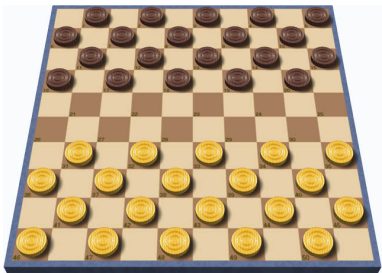
Wersja oryginalna

$$Q(s, a) \leftarrow Q(s, a) + \alpha (R(s) + \gamma \max_{a'} Q(s', a') - Q(s, a))$$

Z aproksymatorem funkcji

$$\theta_i \leftarrow \theta_i + \alpha (R(s) + \gamma \max_{a'} \hat{Q}_\theta(s', a') - \hat{Q}_\theta(s, a)) \frac{\partial \hat{Q}_\theta(s, a)}{\partial \theta_i}$$

Warcaby (Artur Samuel, 1959)



- liniowa aproksymator funkcji: 16 cech,
- wariant metody różnic czasowych (TDL).

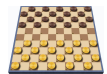
- Jako wartości docelowej używał nie kolejnego stanu, ale stanu na poziomie n -ply (minmax).
- Nie używał wzmocnienia dla stanów terminalnych (przyjął tylko pewne założenia dot. parametrów, co zaskakujące to wystarczyło)

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania

↳ Aplikacje

↳ Warcaby (Artur Samuel, 1959)



- liniowa aproksymator funkcji: 16 cech,
- wariant metody różnic czasowych (TDL).
- ◆ Jako wartości docelowej używał nie kolejnego stanu, ale stanu na poziomie n -ply (minmax).
- ◆ Nie używał wzmocnienia dla stanów terminalnych (przyjął tylko pewne założenia dot. parametrów, co zaskakujące to wystarczyło)

1. Mogło to doprowadzić do nauczenia się np. przegrywania.
2. Gra rozwiązana w 2007 (Schaeffer). Obaj gracze mają gwarantowany remis. Liczba stanów 5×10^{20} .

Tryktak

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania

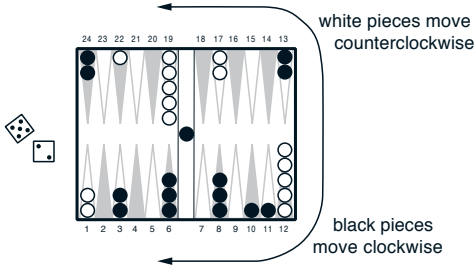
- Aplikacje

Tryktak

Tryktak

white pieces move counterclockwise
black pieces move clockwise

- Bardzo popularna na świecie
- Gra niedeterministyczna (rzuty kostkami)
- Branching factor ~ 400.



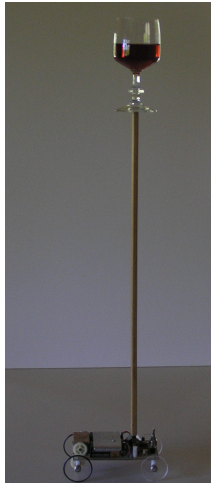
- Bardzo popularna na świecie
- Gra niedeterministyczna (rzuty kostkami)
- Branching factor ~ 400.

1. [Sutton & Barto, 1998] twierdzą, że bardziej popularna niż szachy (ale wydaje się, że nie jest to prawda jeśli chodzi o Polskę).

Balansowanie tyczką / odwrócone wahadło (Michie, Chambers, 1968)

ang. pole balancing / inverted pendulum

- Ciągła przestrzeń stanów
- Co jest stanem?[zadanie 7]



2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania

Aplikacje

└ Balansowanie tyczką / odwrócone wahadło (Michie, Chambers, 1968)

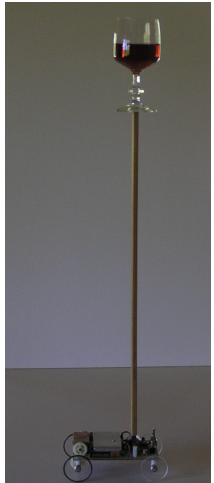
Balansowanie tyczką / odwrócone wahadło (Michie, Chambers, 1968)
ang. pole balancing / inverted pendulum



- Ciągła przestrzeń stanów
- Co jest stanem?[zadanie 7]

Balansowanie tyczką / odwrócone wahadło (Michie, Chambers, 1968)

ang. pole balancing / inverted pendulum



- Ciągła przestrzeń stanów
 - Co jest stanem? [zadanie 7]
- $$\langle x, \theta, \dot{x}, \dot{\theta} \rangle$$
- Jakie akcje są możliwe? [zadanie 8]


2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania

↳ Aplikacje

↳ Balansowanie tyczką / odwrócone wahadło (Michie, Chambers, 1968)

Balansowanie tyczką / odwrócone wahadło (Michie, Chambers, 1968)
ang. pole balancing / inverted pendulum

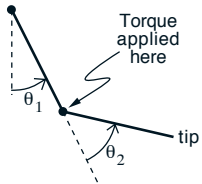


- Ciągła przestrzeń stanów
- Co jest stanem? [zadanie 7]
- Jakie akcje są możliwe? [zadanie 8]

$\langle x, \theta, \dot{x}, \dot{\theta} \rangle$

Acrobot (Spong, 1994)

„Robot-akrobata”



- Ciągła przestrzeń stanów. Co jest stanem? [\[zadanie 9\]](#)

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania

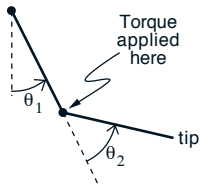
- └ Aplikacje

- └ Acrobot (Spong, 1994)



Acrobot (Spong, 1994)

„Robot-akrobata”



- Ciągła przestrzeń stanów. Co jest stanem? [zadanie 9]

$$\langle \theta_1, \theta_2, \dot{\theta}_1, \dot{\theta}_2 \rangle$$

- Akcje: $sita \in \{-1, 0, +1\}$
- Cel: „stanąć na głowie” w najkrótszym czasie
- Rozwiązanie [Sutton, 1996]:
 - Sarsa(λ) i tile coding
 - $\epsilon = 0$, ale eksploracja bo negatywne nagrody i inicjalizacja $Q = 0$.

Demo: [Acrobot](#), [Double](#) & [Tripple inverted pendulum on a cart](#)

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania

└ Aplikacje

└ Acrobot (Spong, 1994)

Acrobot (Spong, 1994)

„Robot-akrobata”

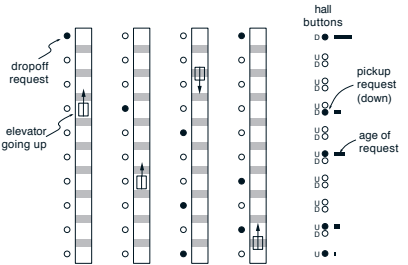


- Ciągła przestrzeń stanów. Co jest stanem? [zadanie 9]
- $\langle \theta_1, \theta_2, \dot{\theta}_1, \dot{\theta}_2 \rangle$
- Akcje: $sita \in \{-1, 0, +1\}$
- Cel: „stanąć na głowie” w najkrótszym czasie
- Rozwiązanie [Sutton, 1996]:
 - Sarsa(λ) i tile coding
 - $\epsilon = 0$, ale eksploracja bo negatywne nagrody i inicjalizacja $Q = 0$.

Demo: [Acrobot](#), [Double](#) & [Tripple inverted pendulum on a cart](#)

Sterowanie dźwigami wind (Crites i Barto, 1996)

ang. elevator dispatching problem



Źródło: Sutton & Barto, 1998

- 4 windy, 10 pięter, przestrzeń stanów: ca. 10^{22} stanów.
- Różne rozważane cele:
 - 1 średni czas oczekiwania,
 - 2 średni czas dostania się na miejsce
 - 3 procent pasażerów, którzy średnio czekają > niż 60s,
 - 4 kwadratowy czas oczekiwania.

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania

Aplikacje

↳ Sterowanie dźwigami wind (Crites i Barto, 1996)

1. A stochastic optimal control problem
2. Przestrzeń stanów jest ciągła (np. pozycja windy), więc podana tutaj liczba stanów jest po (arbitralnej) kwantyzacji.

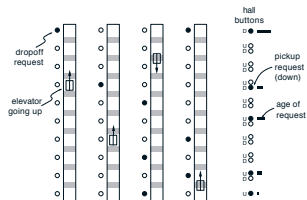


Źródło: Sutton & Barto, 1998

- 4 windy, 10 pięter, przestrzeń stanów: ca. 10^{22} stanów.
- Różne rozważane cele:
 - średni czas oczekiwania,
 - średni czas dostania się na miejsce
 - procent pasażerów, którzy średnio czekają > niż 60s,
 - kwadratowy czas oczekiwania.

Sterowanie dźwigami wind (Crites i Barto, 1996)

ang. elevator dispatching problem



● Przestrzeń akcji?

- Pewne uproszczenia: każda winda osobno (**multi agent reinforcement learning**)
- Zdroworozsądkowe ograniczenia
- Ograniczenia → jedyne akcje: czy zatrzymać się czy nie.

Stan z ciągłym czasem → stan z dyskretnym czasem
(**semi-markovski** proces decyzyjny)

Sieć neuronowa: 47 wejść, 20 węzłów ukrytych i 2 wyjścia

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania

Aplikacje

↳ Sterowanie dźwigami wind (Crites i Barto, 1996)



■ Przestrzeń akcji?

- Pewne uproszczenia: każda winda osobno (**multi agent reinforcement learning**)
- Zdroworozsądkowe ograniczenia
- Ograniczenia → jedyne akcje: czy zatrzymać się czy nie.

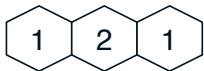
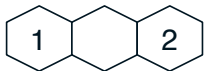
Stan z ciągłym czasem → stan z dyskretnym czasem
(**semi-markovski** proces decyzyjny)

Sieć neuronowa: 47 wejść, 20 węzłów ukrytych i 2 wyjścia

- 1) winda nie może nie zatrzymać się na piętrze koło, którego przejeżdża, jeśli ktoś chce tam wysiąść.
- 2) winda nie może zmienić kierunku, jeśli wszyscy pasażerowie, których ma na pokładzie i jadą w tę stronę jeszcze nie wysiedli
- 3) winda nie może zatrzymać się na piętrze, jeśli nikt tam nie wysiada
- 4) domyślnie zawsze jedzie do góry

Dynamic Channel Allocation (Singh & Bertsekas, 1997)

- Sieci komórkowe
- Całe pasmo częstotliwości jest podzielone na kanały
- 1 kanał może być używany przez wielu użytkowników, jeśli są oni dostatecznie daleko od siebie — minimalna odległość to *channel reuse constraint*).
- Przestrzeń podzielona na komórki (*cell*), każda ze stacją bazową.
- Kanały muszą być przypisane do komórek i do poszczególnych połączeń, tak aby ograniczenie odległościowe było zachowane.
- Cel: minimalizacja połączeń, które nie mogą się odbyć (ze względu na ograniczenie).



2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania

Aplikacje

Dynamic Channel Allocation (Singh & Bertsekas, 1997)

- Sieci komórkowe
- Całe pasmo częstotliwości jest podzielone na kanały
- 1 kanał może być używany przez wielu użytkowników, jeśli są oni dostatecznie daleko od siebie — minimalna odległość to *channel reuse constraint*.
- Przestrzeń podzielona na komórki (*cell*), każda ze stacją bazową.
- Kanały muszą być przypisane do komórek i do poszczególnych połączeń, tak aby ograniczenie odległościowe było zachowane.
- Cel: minimalizacja połączeń, które nie mogą się odbyć (ze względu na ograniczenie).



Dynamic Channel Allocation (Singh & Bertsekas, 1997)

- Fixed assignment vs. dynamic assignment
- Stan: 49 komórek i 70 kanałów, to $2^{49 \times 70}$ stanów sieci (kanał zajęty/wolny)
- Proces semi-markowski: tranzycje: nowe połączenie oraz (koniec połączenia)
- Akcje: który kanał przypisać przy nowym połączeniu

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania

- └ Aplikacje

- └ Dynamic Channel Allocation (Singh & Bertsekas, 1997)

1. [Sutton & Barto, 1998] piszą, że mamy tutaj 70^{49} stanów, ale wydaje się, że się mylą.

- Fixed assignment vs. dynamic assignment
- Stan: 49 komórek i 70 kanałów, to $2^{49 \times 70}$ stanów sieci (kanał zajęty/wolny)
- Proces semi-markowski: tranzycje: nowe połączenie oraz (koniec połączenia)
- Akcje: który kanał przypisać przy nowym połączeniu

2048 (Szubert & Jaśkowski, 2014)

Prezentacja

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania
└ Aplikacje

└ 2048 (Szubert & Jaśkowski, 2014)

Przypomnienie
○○○○○○

Generalizacja
○○○○○○○○○○

Aplikacje
○○○○○○○○○○○○●○○○○○○○○

Bezpośrednie szukanie polityki
○○○

Atari (Mnih et al., 2015)



2015-05-15

Uczenie ze wzmacnieniem — generalizacja i zastosowania
└ Aplikacje

└ Atari (Mnih et al., 2015)

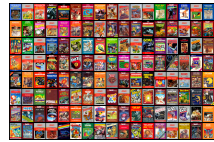
Atari (Mnih et al., 2015)



Atari (Mnih et al., 2015)



Atari (Mnih et al., 2015)



2015-05-15

Uczenie ze wzmacnieniem — generalizacja i zastosowania

- Aplikacje

- Atari (Mnih et al., 2015)

Atari (Mnih et al., 2015)

Motywacja:

- 1 dotychczas sukcesy RL głównie dla środowisk, gdzie:
 - 1 cechy mogą być zaprojektowane ręcznie lub
 - 2 stany niskowymiarowe i całkowicie obserwowalne.
- 2 Ostatnio duże osiągnięcia w rozpoznawaniu obrazów za pomocą uczenia **głębokich sieci neuronowych**.

Problem:

- Zbiór 49 gier Atari
- Wejście: 210×160 pixeli, 60 Hz.

Minimalna wiedza dziedzinowa:

- 1 wejście jest obrazem 2D
- 2 liczba akcji

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania

- Aplikacje

└ Atari (Mnih et al., 2015)

Motywacja:

- dotychczas sukcesy RL głównie dla środowisk, gdzie:
 - cechy mogą być zaprojektowane ręcznie lub
 - stany niskowymiarowe i całkowicie obserwowalne.
- Ostatnio duże osiągnięcia w rozpoznawaniu obrazów za pomocą uczenia **głębokich sieci neuronowych**.

Problem:

- Zbiór 49 gier Atari
- Wejście: 210×160 pixeli, 60 Hz.

Minimalna wiedza dziedzinowa:

- wejście jest obrazem 2D
- liczba akcji

Atari (Mnih et al., 2015)

Q-Learning jest często niestabilny dla nieliniowych aproksymatorów funkcji, a uczenie może się nawet rozbiegać. **Przyczyny:**

- 1 Obserwacje są „podawane” w kolejności: korelacje.
- 2 Małe zmiany $Q \rightarrow$ duża zmiana polityki $\pi \rightarrow$ duża zmiana rozkładu danych uczących (indukowanych przez π).
- 3 Korelacja pomiędzy aktualną wartością Q a wartością docelową $r + \gamma \max_{a'} Q(s', a')$

Rozwiązania:

- 1 **experience replay**: usuwa korelacje związane z kolejnością przetwarzania obserwacji (ad. 1), „wygładzając” zmiany w rozkładzie danych uczących (ad. 2)
- 2 **rzadka aktualizacja Q** : redukuje korelacje z celem (ad. 3)

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania

Aplikacje

└ Atari (Mnih et al., 2015)

1. IZmiana wartości Q powoduje zmiany wartości docelowej $r + \gamma \max_{a'} Q(s', a')$.

- 1 Obserwacje są „podawane” w kolejności: korelacje.
- 2 Małe zmiany $Q \rightarrow$ duża zmiana polityki $\pi \rightarrow$ duża zmiana rozkładu danych uczących (indukowanych przez π).
- 3 Korelacja pomiędzy aktualną wartością Q a wartością docelową $r + \gamma \max_{a'} Q(s', a')$

Rozwiązania:

- 1 **experience replay**: usuwa korelacje związane z kolejnością przetwarzania obserwacji (ad. 1), „wygładzając” zmiany w rozkładzie danych uczących (ad. 2)
- 2 **rzadka aktualizacja Q** : redukuje korelacje z celem (ad. 3)

Przypomnienie
○○○○○○○

Generalizacja
○○○○○○○○○○○

Aplikacje
○○○○○○○○○○○○○○○○○○●○○○○○○○

Bezpośrednie szukanie polityki
○○○

Atari (Mnih et al., 2015)

Demo

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania
└─ Aplikacje

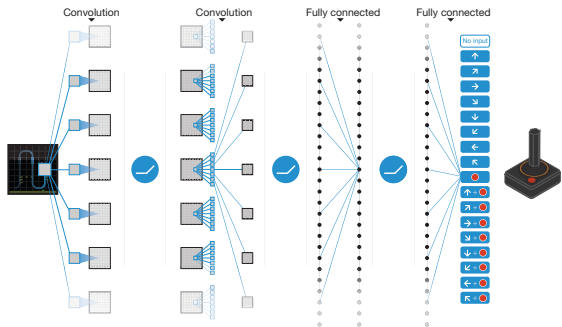
└─ Atari (Mnih et al., 2015)

Atari (Mnih et al., 2015)

Demo

Atari (Mnih et al., 2015)

Konwolucyjna sieć neuronowa



- Wejście: 4 obrazy 84×84
- Wyjścia: jedno wyjścia (zamiast osobnej sieci) dla każdej akcji
- Ukryte: 3 warstwy konwolucyjne + 1 „zwykła”:
 1. 32 filtry 8×8 co 4; 2. 64 filtry 4×4 co 2; 3. 64 filtry 3×3 co 1; 4. „zwykła” 512 neuronów
- Nieliniowość: $\max(0, x)$

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania

└ Aplikacje

└ Atari (Mnih et al., 2015)

Atari (Mnih et al., 2015)

Konwolucyjna sieć neuronowa



- Wejście: 4 obrazy 84×84
- Wyjścia: jedno wyjścia (zamiast osobnej sieci) dla każdej akcji
- Ukryte: 3 warstwy konwolucyjne + 1 „zwykła”:
 - 32 filtry 8×8 co 4; 2. 64 filtry 4×4 co 2; 3. 64 filtry 3×3 co 1; 4. „zwykła” 512 neuronów
- Nieliniowość: $\max(0, x)$

Atari (Mnih et al., 2015)

Przetwarzanie wstępne:

- Oryginalnie: 210×160 pikseli, 128 wartościowa paleta kolorów
- Max z dwóch kolejnych klatek (redukcja efektu mrugania)
- Decymacja (downsampling) do 84×84 .
- Tylko luminancja
- Sieć bierze pod uwagę 4 ostatnie ramki

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania
└ Aplikacje

└ Atari (Mnih et al., 2015)

Przetwarzanie wstępne:

- Oryginalnie: 210×160 pikseli, 128 wartościowa paleta kolorów
- Max z dwóch kolejnych klatek (redukcja efektu mrugania)
- Decymacja (downsampling) do 84×84 .
- Tylko luminancja
- Sieć bierze pod uwagę 4 ostatnie ramki

Atari (Mnih et al., 2015)

Uczenie

- Algorytm **RMSProp** (specjalizowany dla mini-batchy), mini-batch'e wielkości 32
- **50 milionów** ramek uczących (38 godzin rzeczywistego czasu gry)
- Pamięć: **1 mln** ostatnich ramek
- Polityka ϵ -zachłanna, gdzie początkowo $\epsilon = 1.0$ liniowo się zmniejszało do $\epsilon = 0.1$ po milionie pierwszych ramek.
- Nowa decyzja co $k = 4$ ramek (wydajność)
- Funkcja straty dla i -tej iteracji:

$$L_i(\theta_i) = \mathbb{E}_{(s,a,r,s') \sim U(D)} \left[\left(r + \gamma \max_{a'} Q_{\theta_i^-}(s', a') - Q_{\theta_i}(s, a) \right)^2 \right],$$

gdzie:

- 1 D jest **pamięcią** (zbiorem zapamiętanych doświadczeń)
- 2 θ_i^- jest „starym” zbiorem parametrów

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania
└ Aplikacje

└ Atari (Mnih et al., 2015)

Atari (Mnih et al., 2015)

Uczenie

- Algorytm **RMSProp** (specjalizowany dla mini-batchy), mini-batch'e wielkości 32
- **50 milionów** ramek uczących (38 godzin rzeczywistego czasu gry)
- Pamięć: **1 mln** ostatnich ramek
- Polityka ϵ -zachłanna, gdzie początkowo $\epsilon = 1.0$ liniowo się zmniejszało do $\epsilon = 0.1$ po milionie pierwszych ramek.
- Nowa decyzja co $k = 4$ ramek (wydajność)
- Funkcja straty dla i -tej iteracji:

$$L_i(\theta_i) = \mathbb{E}_{(s,a,r,s') \sim U(D)} \left[\left(r + \gamma \max_{a'} Q_{\theta_i^-}(s', a') - Q_{\theta_i}(s, a) \right)^2 \right],$$

gdzie:

- D jest **pamięcią** (zbiorem zapamiętanych doświadczeń)
- θ_i^- jest „starym” zbiorem parametrów

Atari (Mnih et al., 2015)

Q-Learning with experience replay

Algorithm 1: deep Q-learning with experience replay.Initialize replay memory D to capacity N Initialize action-value function Q with random weights θ Initialize target action-value function \hat{Q} with weights $\theta^- = \theta$ **For** episode = 1, M **do** Initialize sequence $s_1 = \{x_1\}$ and preprocessed sequence $\phi_1 = \phi(s_1)$ **For** $t = 1, T$ **do** With probability ϵ select a random action a_t otherwise select $a_t = \operatorname{argmax}_a Q(\phi(s_t), a; \theta)$ Execute action a_t in emulator and observe reward r_t and image x_{t+1} Set $s_{t+1} = s_t, a_t, x_{t+1}$ and preprocess $\phi_{t+1} = \phi(s_{t+1})$ Store transition $(\phi_t, a_t, r_t, \phi_{t+1})$ in D Sample random minibatch of transitions $(\phi_j, a_j, r_j, \phi_{j+1})$ from D Set $y_j = \begin{cases} r_j & \text{if episode terminates at step } j+1 \\ r_j + \gamma \max_{a'} \hat{Q}(\phi_{j+1}, a'; \theta^-) & \text{otherwise} \end{cases}$ Perform a gradient descent step on $(y_j - Q(\phi_j, a_j; \theta))^2$ with respect to the network parameters θ Every C steps reset $\hat{Q} = Q$ **End For****End For**

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania
└ Aplikacje

└ Atari (Mnih et al., 2015)

Atari (Mnih et al., 2015)
Q-Learning with experience replay

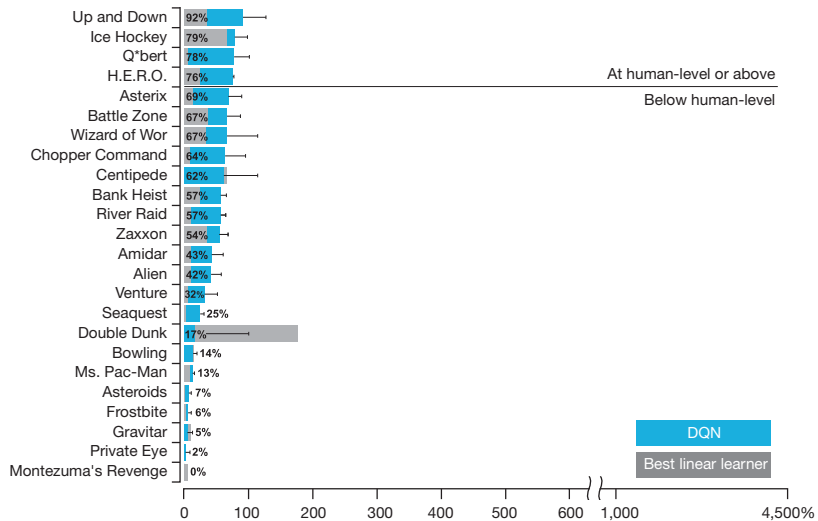
```

Algorithm 1: deep Q-learning with experience replay.
Initialize replay memory D to capacity N
Initialize action-value function Q with random weights θ
Initialize target action-value function Q̂ with weights θ⁻ = θ
For episode = 1, M do
  Initialize sequence s₁ = {x₁} and preprocessed sequence φ₁ = φ(s₁)
  For t = 1, T do
    With probability ε select a random action aₜ
    otherwise select aₜ = argmaxₐ Q(φ(sₜ), a; θ)
    Execute action aₜ in emulator and observe reward rₜ and image xₜ₊₁
    Set sₜ₊₁ = sₜ, aₜ, xₜ₊₁ and preprocess φₜ₊₁ = φ(sₜ₊₁)
    Store transition (φₜ, aₜ, rₜ, φₜ₊₁) in D
    Sample random minibatch of transitions (φⱼ, aⱼ, rⱼ, φⱼ₊₁) from D
    Set yⱼ = { rⱼ if episode terminates at step j + 1
              rⱼ + γ maxₐ' Q̂(φⱼ₊₁, a'; θ⁻) otherwise
    Perform a gradient descent step on (yⱼ - Q(φⱼ, aⱼ; θ))² with respect to the
    network parameters θ
    Every C steps reset Q̂ = Q
  End For
End For

```

Atari (Mnih et al., 2015)

Wyniki



100% = człowiek

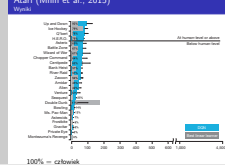
2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania

↳ Aplikacje

↳ Atari (Mnih et al., 2015)

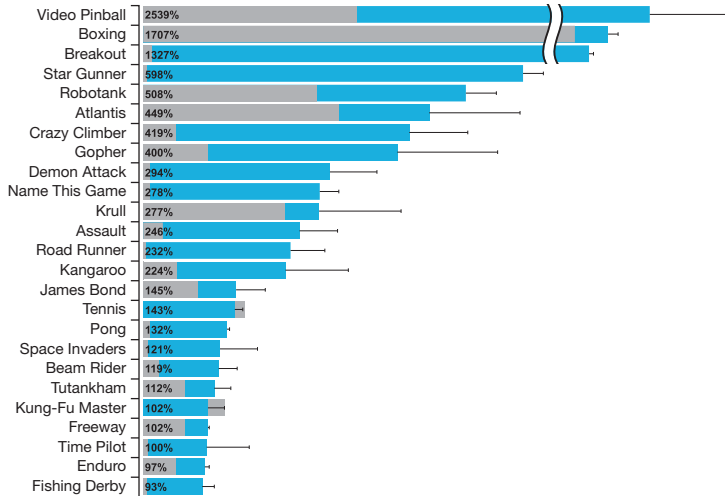
Atari (Mnih et al., 2015)



100% = człowiek

Atari (Mnih et al., 2015)

Wyniki



100% = człowiek

2015-05-15

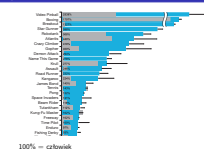
Uczenie ze wzmocnieniem — generalizacja i zastosowania

↳ Aplikacje

↳ Atari (Mnih et al., 2015)

Atari (Mnih et al., 2015)

Wyniki



100% = człowiek

Bezpośrednie szukanie polityki

- Polityka $\pi : S \rightarrow A$
- Chcemy reprezentować π nie dla każdego stanu, ale w sposób bardziej zwężony (np. zestaw parametrów θ)
- Np. możemy reprezentować politykę π jako zestaw aproksymatorów funkcji Q :

$$\pi(s) = \max_a \hat{Q}_\theta(s, a),$$

gdzie \hat{Q}_θ jest np. sumą jakichś funkcji ważoną parametrami θ (vide poprzednia sekcja)

- **Szukanie polityki** = dostosowuj θ , tak aby poprawiać działanie π .
 - Czyli: ucz się funkcji \hat{Q}_θ .
 - Czy to jest to samo, co Q-learning? [zadanie 10]

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania

└ Bezpośrednie szukanie polityki

└ Bezpośrednie szukanie polityki

1. Nie

- Polityka $\pi : S \rightarrow A$
- Chcemy reprezentować π nie dla każdego stanu, ale w sposób bardziej zwężony (np. zestaw parametrów θ)
- Np. możemy reprezentować politykę π jako zestaw aproksymatorów funkcji Q :

$$\pi(s) = \max_a \hat{Q}_\theta(s, a),$$

gdzie \hat{Q}_θ jest np. sumą jakichś funkcji ważoną parametrami θ (vide poprzednia sekcja)

- **Szukanie polityki** = dostosowuj θ , tak aby poprawiać działanie π .
 - Czyli: ucz się funkcji \hat{Q}_θ .
 - Czy to jest to samo, co Q-learning? [zadanie 10]

Reprezentacja polityki

$$\pi(s) = \max_a \hat{Q}_\theta(s, a)$$

- W **Q-learning'u** (z aproksymatorem funkcji) szukamy \hat{Q}_θ , które jest możliwie bliskie Q^* .
- W **szukaniu polityki** szukamy θ , które powoduje, że π „działa dobrze”.
 - Przykład: Czy $\hat{Q}_\theta(s, a) = Q^*(s, a)/10$ jest optymalnym rozwiązaniem?[\[zadanie 11\]](#)
- Problem: π jest nieciągłą funkcją parametrów θ , jeśli akcje są dyskretne
 - czasem minimalna zmiana w θ może spowodować, że $\pi(s)$ „przeskoczy” z jednej akcji na inną.
 - dlatego **uczenie gradientowe** π nie jest możliwe.

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania

└ Bezpośrednie szukanie polityki

└ Reprezentacja polityki

1. Tak. Mimo, że $\hat{Q}_\theta(s, a) \neq Q^*(s, a)$, bo liczy się \max_a

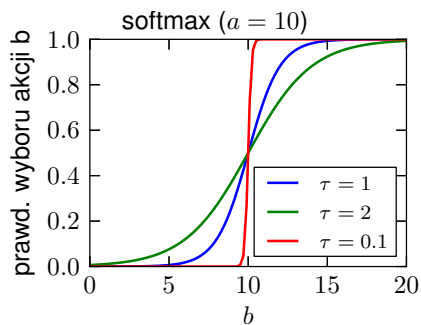
$$\pi(s) = \max_a \hat{Q}_\theta(s, a)$$

- W Q-learning'u (z aproksymatorem funkcji) szukamy \hat{Q}_θ , które jest możliwie bliskie Q^* .
- W **szukaniu polityki** szukamy θ , które powoduje, że π „działa dobrze”.
 - Przykład: Czy $\hat{Q}_\theta(s, a) = Q^*(s, a)/10$ jest optymalnym rozwiązaniem?[\[zadanie 11\]](#)
- Problem: π jest nieciągłą funkcją parametrów θ , jeśli akcje są dyskretne
 - czasem minimalna zmiana w θ może spowodować, że $\pi(s)$ „przeskoczy” z jednej akcji na inną.
 - dlatego **uczenie gradientowe** π nie jest możliwe.

Polityka stochastyczna

- Dlatego używa się **polityki stochastycznej** $\pi_\theta(s, a)$, reprezentującej prawd. wybrania akcji a w stanie s .
- Reprezentacja z użyciem **funkcji softmax**:

$$\pi_\theta(s, a) = e^{\hat{Q}_\theta(s, a)/\tau} / \sum_{a'} e^{\hat{Q}_\theta(s, a')/\tau}$$



- Zaleta: różniczkowalna

2015-05-15

Uczenie ze wzmocnieniem — generalizacja i zastosowania

└ Bezpośrednie szukanie polityki

└ Polityka stochastyczna

- Dlatego używa się **polityki stochastycznej** $\pi_\theta(s, a)$, reprezentującej prawd. wybrania akcji a w stanie s .
- Reprezentacja z użyciem **funkcji softmax**:

$$\pi_\theta(s, a) = e^{\hat{Q}_\theta(s, a)/\tau} / \sum_{a'} e^{\hat{Q}_\theta(s, a')/\tau}$$

