

ÚVOD DO UMĚLÉ INTELIGENCE (CVIČENÍ 11)

Simona Ondrčková

8. DOMÁCÍ ÚKOL

2 hráči, kteří odebírají kameny očíslované $1, 2, \dots, n$.

Hráč může vzít jakýkoliv kámen, jež je násobkem nebo dělitelem předchozího odebraného čísla.

Pokud hráč nemůže odebrat kámen prohrál.

Napište funkci, která dostane seznam zbývajících kamenů a poslední odebraný kámen a určí zda je situace vyhrávající a vrátí jaký kámen se má odebrat v dalším tahu.

9. DOMACÍ ÚKOL

Chceme rozpoznat, zda má žena cukrovku.

Máme k dispozici soubor s diagnostikami lidí včetně cukrovky k učení i testování.

Nastavte parametry rozhodovacího stromu, tak aby byla co největší úspěšnost.

Výsledek musí být statisticky relevantní

Odevzdáte zprávu v PDF:

Popíšete parametry co ovlivnily výsledek

Napište nejlepší výsledky a parametry, kterých jste dosáhli.

Nakreslete nejlepší rozhodovací strom.

Vytvořte graf závislosti a úplnosti na velikosti testovací množiny.

ZPĚTNOVAZEBNÉ UČENÍ

Máme nedeterministické akce.

Stavy mají odměny, které dokážeme detekovat.

Pasivní učení: Máme danou strategii (policy) a snažíme se zjistit užitek stavů nebo akcí.

Aktivní učení: Učíme se co dělat (policy).

PŘÍMÝ MODEL PASIVNÍHO UČENÍ

Chceme vypočítat užitkovou funkci, ale nevíme ani $R(s)$ ani $P(s' | a, s)$.

Chceme: $U^\pi(s) = E(\sum_{t=0}^{\infty} \gamma^t R(s_t))$

Agent bude procházet prostředí dle strategie.

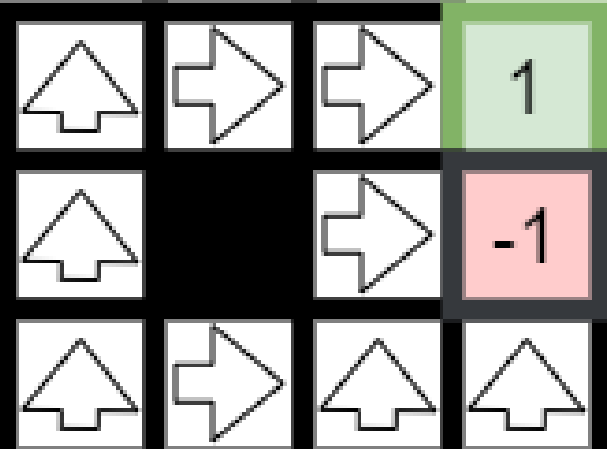
V každém stavu zjistí odměny dle vjemu.

Potom vypočítá ocenění cesty do cíle (reward to go).

Pak vypočítáme průměr přes všechny výskyty stavu.

(Direct utility estimation)

PŘÍMÝ MODEL



Začínáme na pozici $(1,1)$. Robot projde 2 cesty:

1: $(1,1)_{-0,08}, (1,2)_{-0,08}, (1,3)_{-0,08}, (1,3)_{-0,08}, (2,3)_{-0,08}, (3,3)_{-0,08}, (4,3)_{+1}$

2: $(1,1)_{-0,08}, (2,1)_{-0,08}, (3,1)_{-0,08}, (2,1)_{-0,08}, (3,1)_{-0,08}, (3,2)_{-0,08}, (3,3)_{-0,08}, (4,3)_{+1}$

Vypočítejte ocenění do cíle pro stavy $(1,1), (3,1), (3,3)$.

$\gamma=1$.

NEVÝHODY PŘÍMÉHO MODELU

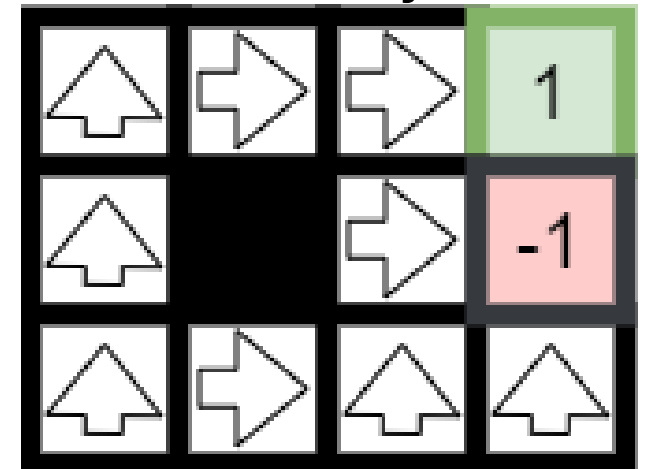
Nevyužívá informace z Bellmanovi rovnice.

My víme, že pokud stav (3,3) je velice dobrý, tak stav

(2,3) bude také dobrý, ale tento model to nevyužívá. Počítá stavy jakoby byly nezávislé.

To znamená, že konverguje pomaleji.

$$U^\pi(s) = R(s) + \gamma \sum_{s'} P(s'|s, \pi(s)) * U^\pi(s')$$



ADAPTIVNÍ DYNAMICKÉ PROGRAMOVÁNÍ (ADP)

Učíme se $R(s)$, ale i $P(s'|s,a)$. $R(s)$ aktualizujeme podle vjemu.

Jak určíme $P(s'|s,a)$? Podle frekvence ve vzorcích.

Užitek pak vypočteme dle Bellmanovy rovnice.

-1 je v bodě (2,1).

Vzorky: 1: $(1,1)_{-0,08}, (1,2)_{-0,08}, (1,1)_{-0,08}, (2,1)_{-1}$

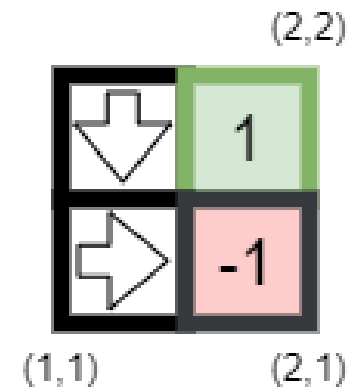
2: $(1,1)_{-0,08}, (1,2)_{-0,08}, (2,2)_1,$

3: $(1,1)_{-0,08}, (2,1)_{-1}$

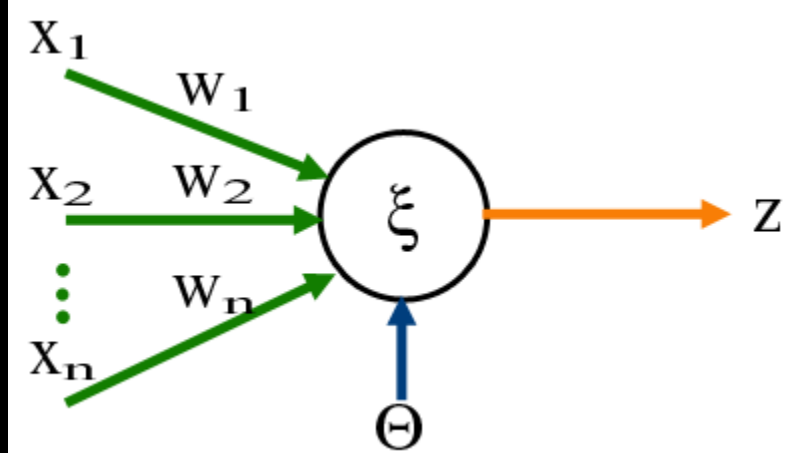
Vypočítejte $U(1,1)$ a $U(2,1)$.

$$U^\pi(s) = R(s) + \gamma \sum_{s'} P(s'|s, \pi(s)) * U^\pi(s')$$

Potom vygenerujete další vzorek (stačí i jeden krok) a aktualizujete pravděpodobnosti (případně funkci odměny). Poté můžete aktualizovat užitekovou funkci.



PERCEPTRON



Jednoduchá neuronová síť s jedním uzlem.

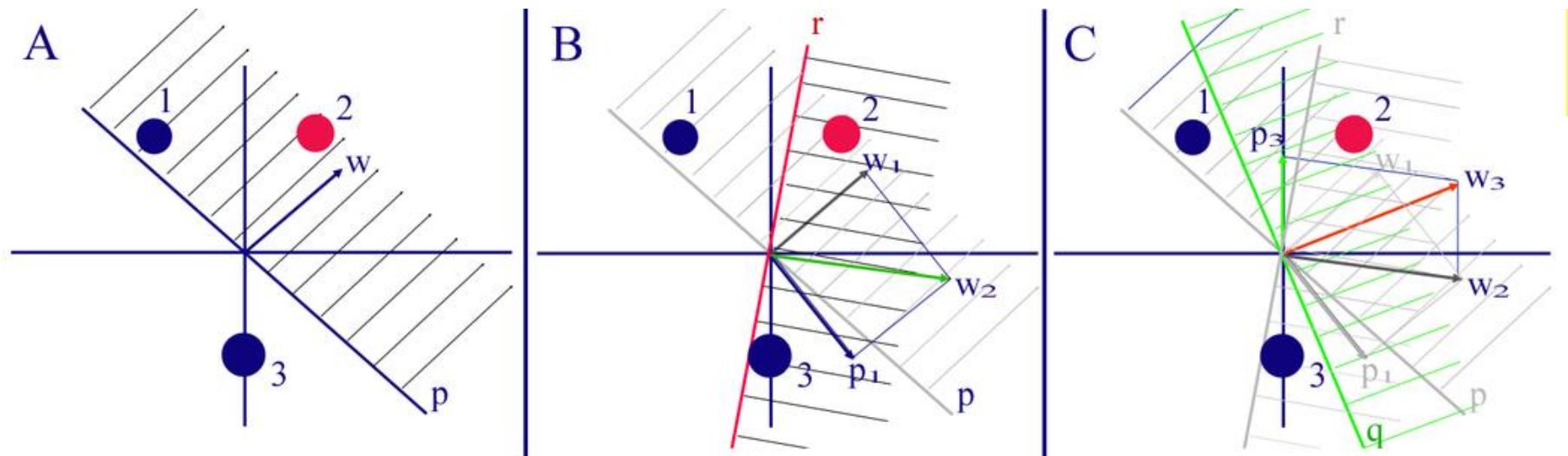
$$f(\xi) = \begin{cases} 1: \text{pro } w * x + b > 0 \\ 0 \end{cases}$$

Učení perceptronu:

- 1) Náhodně zvolíme váhy.
- 2) Dostaneme vstupní vektor a vypočteme výsledek: $f(\xi) = f(\sum_{i=0}^n w_i x_i + b)$
- 3) Vyhodnotí výsledek.
- 4) Pokud byl výsledek špatně upraví váhy: $w' = w + \alpha \epsilon * x$, kde ϵ je chyba perceptronu, $b' = b + \alpha \epsilon$.

PERCEPTRON

$$f(\xi) = \begin{cases} 1: & \text{pro } w * x + b > 0 \\ 0 & \end{cases}, w' = w + \alpha \epsilon * x, b' = b + \alpha \epsilon$$



PERCEPTRON

$$f(\xi) = \begin{cases} 1: \text{pro } w * x + b > 0 \\ 0 \end{cases}, w' = w + \alpha \epsilon * x, b' = b + \alpha \epsilon$$

Máme perceptron s váhami 0 a 0, rychlostí učení 0,5 a prahem -1.
Vypočítejte výstup perceptronu pro následující a aktualizujte váhy až dokud neskončí nebo se nezacyklí.

Vstup: ((0,0),0) , Vstup: ((0,1),0) , Vstup: ((1,0),0), Vstup((1,1),1)

PERCEPTRON

$$f(\xi) = \begin{cases} 1: \text{pro } w * x + b > 0 \\ 0 \end{cases}, w' = w + \alpha \epsilon * x, b' = b + \alpha \epsilon, w_1 = (0,0) \quad b_1 = -1 \quad \alpha = 0.5$$

Vstup: $((0,0),0)$: $f((0,0))$: $0 * 0 + 0 * 0 - 1 = -1, -1 \not> 0$: 0

Chceme 0, tedy výsledek správný můžeme pokračovat.

Vstup: $((0,1),0)$: $f((0,1))$: $0 * 0 + 0 * 1 - 1 = -1, -1 \not> 0$: 0 **OK**

Vstup: $((1,0),0)$: $f((0,1))$: $0 - 0 - 1 = -1, -1 \not> 0$: 0 **OK**

Vstup: $((1,1),1)$: $f((1,1))$: $0 - 0 - 1 = -1, -1 \not> 0$: 0 **Chceme 1**, tedy aktualizujeme: $w_2 = (0,0) + 0.5 * (1-0) * (1,1) = (0.5, 0.5)$ $b_2 = -1 + 0.5 * (1) = -0.5$

Začneme od začátku:

Vstup: $((0,0),0)$: $f((0,0))$: $0.5 * 0 + 0.5 * 0 - 0.5 = -0.5, -0.5 \not> 0$: 0 **Ok**, Vstup: $((0,1),0)$: $f((0,1))$: $0.5 * 0 + 0.5 * 1 - 0.5 = 0, 0 \not> 0$: 0 **OK** $f((1,0))$: $0.5 * 1 + 0.5 * 0 - 0.5 = 0, 0 \not> 0$: 0 **OK** $f((1,1))$: $0.5 * 1 + 0.5 * 1 - 0.5 = 0.5, 0.5 > 0$: **1 OK**

10. DOMÁCÍ ÚKOL

Máte natrénovanou neuronovou síť, která určuje témata zpráv (novinek).

Upravte vlastnosti sítě a napište zprávu v PDF o tom jak to ovlivnilo výsledky.

Implementujte validační set.

Změňte počet vrstev a skrytých neuronů.

Popište nejlepší řešení.

(Přesné zadání na gitu).