Strojové učenie II

prednáška 6 – Aproximácia hodnotových funkcií

Ing. Ján Magyar, PhD.
Katedra kybernetiky a umelej inteligencie
Technická univerzita v Košiciach

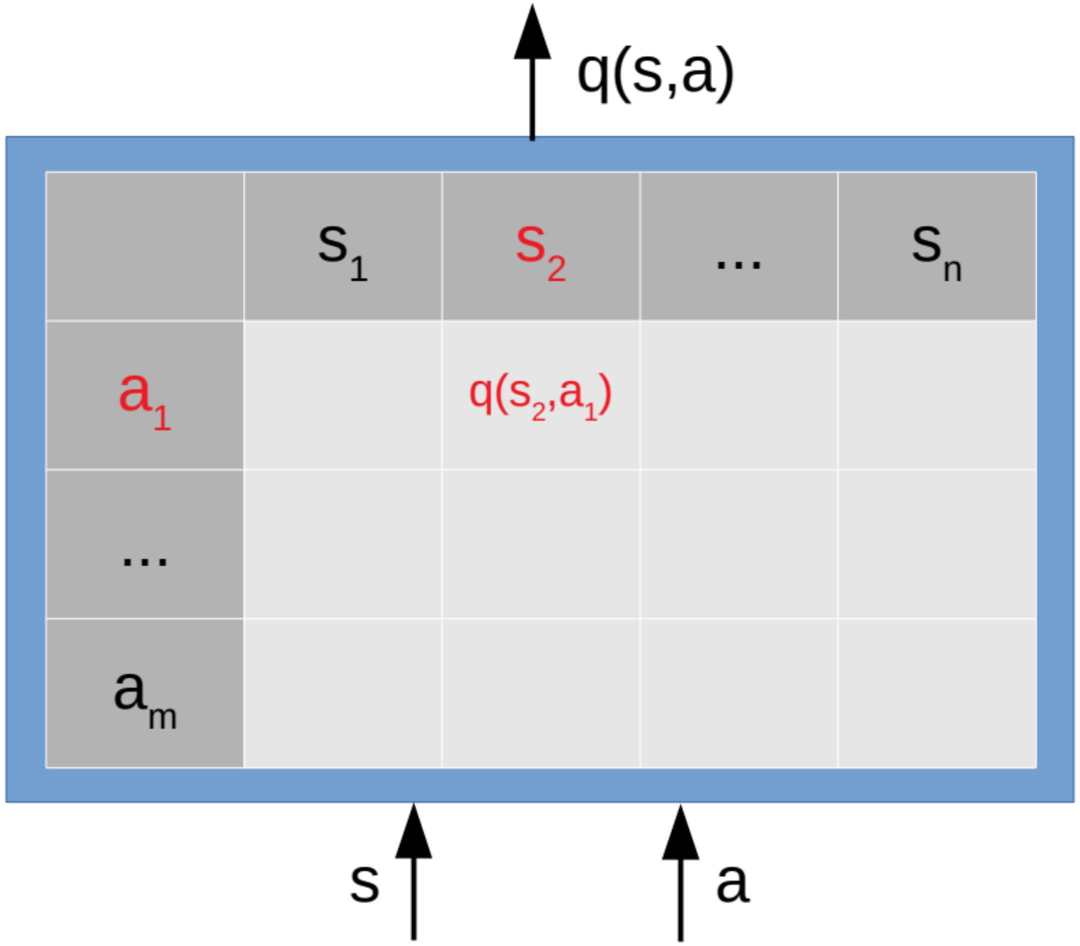
2021/2022 letný semester

Problém škálovania

- najčastejšie škálovanie počtu stavov
  - konečný počet stavov
    - backgammon: 10^{20} stavov
    - go: 10^{170} stavov
  - nekonečný počet stavov – spojité problémy

Presná reprezentácia hodnotovej funkcie

- každý stav / pár (stav, akcia) má jedinečnú reprezentáciu
  - s rastúcim počtom stavov/akcií sa tabuľka zväčšuje
- určenie hodnoty = vyhľadanie príslušného údaja v tabuľke
- problémy
  - príliš veľa miest v pamäti
  - nutnosť uvažovať osobitne každý stav / pár (stav, akcia)

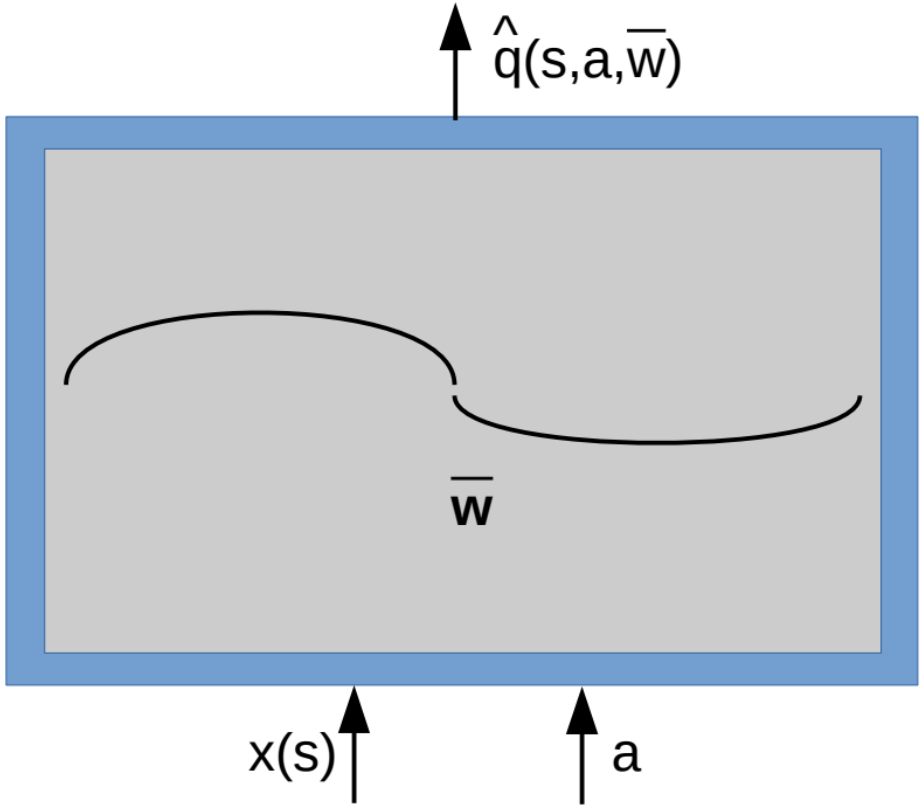


Aproximátor

- vytvorenie na základe príkladov tvaru $s \to v(s)$
- parametrický aproximátor $\hat{v}(s, \bar{w})$ má tvar vhodne zvolenej parametrickej funkcie
  - lineárny
    - lineárna kombinácia príznakov
  - nelineárny
    - umelá neurónová sieť
    - rozhodovací strom
- pamäťový (neparametrický) aproximátor $\hat{v}(s, \mathcal{M})$ má tvar množiny príkladov
  - $\hat{v}(s, \mathcal{M}) = \sum_{s' \in \mathcal{M}} k(s, s')\, v(s')$
    - metóda najbližšieho suseda
    - metóda váženého priemeru

Parametrická aproximácia funkcie

- namiesto enumeračnej tabuľky použitý aproximátor
- stav je reprezentovaný príznakmi
- namiesto hodnôt *v* / *q* sa učí súbor váh $\overline{w}$
  - váh je menej, ako počet stavov
  - hodnoty *v* / *q* sa odvodzujú výpočtom
- zovšeobecnenie skúmaných stavov na neskúmané



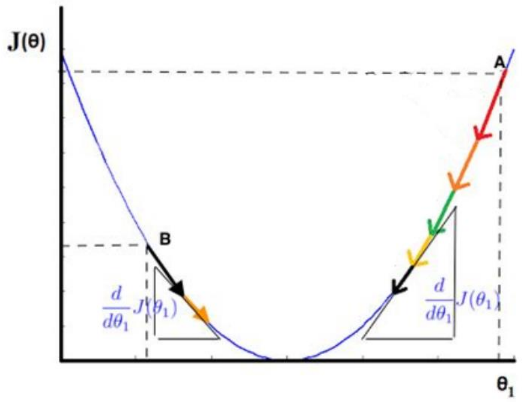
$$\hat{v}(s,\overline{w}) \approx v_\pi(s)$$
$$\hat{q}(s,a,\overline{w}) \approx q_\pi(s,a)$$

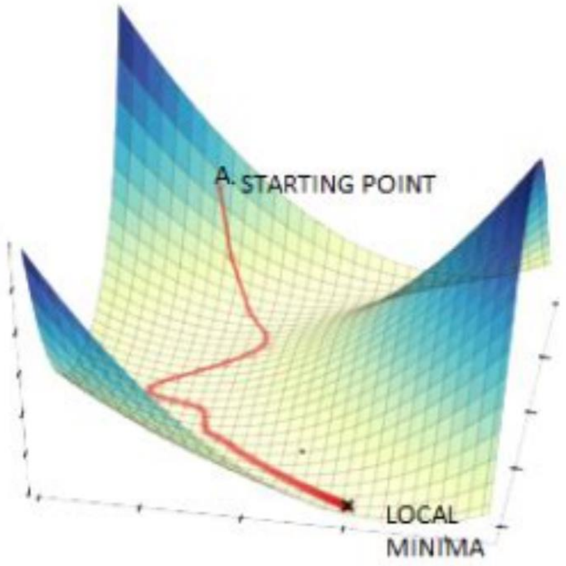
Aproximačný rámec

- individuálna aktualizácia: $v(S_t) \longmapsto U_t$
  - dynamické programovanie:
    $$v(S_t) \longmapsto E_\pi[R_{t+1} + \gamma\hat{v}(S_{t+1}, \overline{w}_t) | S_t = s]$$
  - Monte Carlo: $v(S_t) \longmapsto G_t$
  - time-difference: $v(S_t) \longmapsto R_{t+1} + \gamma\hat{v}(S_{t+1}, \overline{w}_t)$
- aktualizácia hodnoty stavu musí byť urobená ako aktualizácia aproximátora
  - zmenou váhového vektora
  - aktualizácia zasiahne súčasné aj iné stavy (generalizácia)
- požadovaný posun sa vyjadrí ako príklad žiadaného vstupno-výstupného chovania aproximátora
  - aktualizácia sa realizuje pomocou metódou kontrolovaného učenia, ktorá podporuje
    - inkrementálne učenie
    - nestacionárne príklady

Gradientová minimalizácia

- $J(\overline{w})$ je diferencovateľná funkcia parametra $\overline{w}$
- gradient
  $$\nabla J(\overline{w}) = \left(\frac{\partial J(\overline{w})}{\partial w_1}, \dots, \frac{\partial J(\overline{w})}{\partial w_n}\right)^T$$
- iteračné hľadanie lokálneho minima funkcie $J(\overline{w})$
- aktualizácia parametra $\overline{w}$ v smere gradientu $\triangle \overline{w} = -\alpha \nabla J(\overline{w})$
- metóda GD





Aktualizácia aproximátora

- predpokladajme, že chceme aktualizáciu v zmysle $\hat{v}(S_t, \overline{w}_t) \longmapsto U_t$
- chyba aproximátora pre stav S_t je $\frac{1}{2}(U_t - \hat{v}(S_t, \overline{w}_t))^2$
- aktualizácia váhového vektora

$$
\begin{aligned}
\overline{w}_{t+1} &= \overline{w}_t - \alpha \nabla \left(\frac{1}{2} \big(U_t - \hat{v}(S_t, \overline{w}_t) \big)^2 \right) \\
&= \overline{w}_t + \alpha \big(U_t - \hat{v}(S_t, \overline{w}_t) \big) \nabla \hat{v}(S_t, \overline{w}_t)
\end{aligned}
$$

- použitie metódy SGD
  - aktualizácia robená iba podľa jedného príkladu, nie viacerých naraz
  - príklad získavaný z interakcie s prostredím (použitím metód MC alebo TD)

Algoritmus TD odhadu v_π

Semi-gradient TD(0) for estimating $\hat{v} \approx v_\pi$

Input: the policy π to be evaluated
Input: a differentiable function $\hat{v} : \mathcal{S}^+ \times \mathbb{R}^d \to \mathbb{R}$ such that $\hat{v}(\text{terminal},\cdot) = 0$
Algorithm parameter: step size $\alpha > 0$
Initialize value-function weights $\mathbf{w} \in \mathbb{R}^d$ arbitrarily (e.g., $\mathbf{w} = \mathbf{0}$)

Loop for each episode:
    Initialize S
    Loop for each step of episode:
        Choose $A \sim \pi(\cdot|S)$
        Take action A, observe R, S'
        $\mathbf{w} \leftarrow \mathbf{w} + \alpha\big[R + \gamma\hat{v}(S',\mathbf{w}) - \hat{v}(S,\mathbf{w})\big]\nabla\hat{v}(S,\mathbf{w})$
        $S \leftarrow S'$
    until S is terminal

Zdroj: Sutton-Barto: Reinforcement Learning, 2nd ed., 2018

Konvergencia aproximátora

- neminimalizujeme chybovú funkciu
  - iba malý posun smerom k minimu chybovej funkcie
    - minimalizácia chyby pre nejaký stav by znamenala zväčšenie chyby pre iné stavy
  - potrebné pre vybalansovanie chýb pre rôzne stavy
- cieľová hodnota U_t pre stav S_t nie je správna hodnota $v_\pi(S_t)$ ale iba jej náhodný odhad
  - ak $E[U_t|S_t = s] = v_\pi(S_t)$, tak $\overline{w}_t$ konverguje k lokálnemu optimu ak α sa postupne zmenšuje tak, že platia vzťahy $\sum_{n=1}^{\infty} \alpha_n = \infty$ a $\sum_{n=1}^{\infty} \alpha_n^2 < \infty$
  - ak U_t závisí na $\overline{w}$ (lebo závisí na odhade $\hat{v}(S_t, \overline{w})$), tak konvergencia nie je garantovaná
    - semi-gradientové metódy

Lineárny aproximátor

- aproximátor má tvar lineárnej kombinácie príznakov reprezentujúcich stav
$$\hat{v}(s, \overline{w}) = \overline{w}^T \bar{x}(s) = \sum_{i=1}^{n} w_i x_i(s)$$
kde $\bar{x}(s)$ je príznakový vektor stavu s
- gradient s ohľadom na $\overline{w}$
$$\nabla \hat{v}(s, \overline{w}) = \bar{x}(s)$$
- lineárna funkcia nemá lokálne extrémy
- aktualizácia parametrov podľa
$$\overline{w}_{t+1} = \overline{w}_t + \alpha\big(U_t - \overline{w}^T \bar{x}(S_t)\big)\bar{x}(S_t)$$

TD konvergencia lineárneho aproximátora

$$\bar{w}_{t+1} = \bar{w}_t + \alpha\left(R_{t+1} + \gamma \bar{w}^T \bar{x}(S_{t+1}) - \bar{w}^T \bar{x}(S_t)\right)\bar{x}(S_t)$$
$$= \bar{w}_t + \alpha(R_{t+1}\bar{x}(S_t) - \bar{x}(S_t)(\bar{x}(S_t) - \gamma \bar{x}(S_{t+1}))^T \bar{w}_t)$$

$$E[\bar{w}_{t+1}|\bar{w}_t] = \bar{w}_t + \alpha(E[R_{t+1}\bar{x}(S_t)] - E[\bar{x}(S_t)\left(\bar{x}(S_t) - \gamma \bar{x}(S_{t+1})\right)^T]\bar{w}_t)$$

$$E[\bar{w}_{t+1}|\bar{w}_t] = \bar{w}_t$$

$$0 = E[R_{t+1}\bar{x}(S_t)] - E[\bar{x}(S_t)(\bar{x}(S_t) - \gamma \bar{x}(S_{t+1}))^T]\bar{w}_{TD}$$

$$\bar{w}_{TD} = (E[\bar{x}(S_t)(\bar{x}(S_t) - \gamma \bar{x}(S_{t+1}))^T])^{-1} E[R_{t+1}\bar{x}(S_t)]$$

kde $\bar{w}_{TD}$ je fixný bod

doporučené nastavenie:

$\alpha = 1/(\tau E[\bar{x}^T \bar{x}])$

Epizodické učenie politiky

- aproximovaná funkcia $\hat{q}(S_t, A_t, \overline{w})$
  - aproximátor $x(s, a) \mapsto \hat{q}(s, a, \overline{w})$ na základe $\overline{w}$
- aktualizácia v zmysle $\hat{q}(S_t, A_t, \overline{w}) \mapsto U_t$
- chyba aproximátora pre dvojicu S_t, A_t je $\frac{1}{2}\left(U_t - \hat{q}(S_t, A_t, \overline{w}_t)\right)^2$
- aktualizácia váhového vektora
$$\overline{w}_{t+1} = \overline{w}_t - \alpha\left(U_t - \hat{q}(S_t, A_t, \overline{w}_t)\right)\nabla\hat{q}(S_t, A_t, \overline{w}_t)$$
- aktualizácia váhového vektora pre Sarsu
$$\overline{w}_{t+1} = \overline{w}_t - \alpha\left(R_{t+1} + \gamma\hat{q}(S_{t+1}, A_{t+1}, \overline{w}_t) - \hat{q}(S_t, A_t, \overline{w}_t)\right)\nabla\hat{q}(S_t, A_t, \overline{w}_t)$$
- zlepšenie politiky $A_t^* = \underset{a}{\operatorname{argmax}}\, \hat{q}(S_t, a, \overline{w}_{t-1})$

Algoritmus semi-gradient Sarsa

Episodic Semi-gradient Sarsa for Estimating $\hat{q} \approx q_*$

Input: a differentiable action-value function parameterization $\hat{q} : \mathcal{S} \times \mathcal{A} \times \mathbb{R}^d \to \mathbb{R}$
Algorithm parameters: step size $\alpha > 0$, small $\varepsilon > 0$
Initialize value-function weights $\mathbf{w} \in \mathbb{R}^d$ arbitrarily (e.g., $\mathbf{w} = \mathbf{0}$)

Loop for each episode:
    $S, A \leftarrow$ initial state and action of episode (e.g., ε-greedy)
    Loop for each step of episode:
        Take action A, observe R, S'
        If S' is terminal:
            $\mathbf{w} \leftarrow \mathbf{w} + \alpha\big[R - \hat{q}(S, A, \mathbf{w})\big]\nabla\hat{q}(S, A, \mathbf{w})$
            Go to next episode
        Choose A' as a function of $\hat{q}(S', \cdot, \mathbf{w})$ (e.g., ε-greedy)
        $\mathbf{w} \leftarrow \mathbf{w} + \alpha\big[R + \gamma\hat{q}(S', A', \mathbf{w}) - \hat{q}(S, A, \mathbf{w})\big]\nabla\hat{q}(S, A, \mathbf{w})$
        $S \leftarrow S'$
        $A \leftarrow A'$

Zdroj: Sutton-Barto: Reinforcement Learning, 2nd ed., 2018

Kontinuálne učenie politiky

- prístup založený na γ je problematický
- použitie aproximácie hodnotovej funkcie spôsobuje
  - zmena politiky zlepšujúca diskontovanú hodnotu nejakého stavu negarantuje zlepšenie politiky ako celku
  - **teoréma zlepšovania politiky neplatí**
  - ε-greedifikácia môže niekedy vyústiť do menej kvalitnej politiky
- diskontný prístup sa nahrádza prístupom založeným na priemernej odmene

Priemerná odmena

- priemerná odmena pri použití politiky π

$$r(\pi) = \lim_{h\to\infty} \frac{1}{h} \sum_{t=1}^{h} E[R_t | S_0, A_{0:t-1} \sim \pi]$$

$$= \sum_{s} \mu_\pi(s) \sum_{a} \pi(a|s) \sum_{r} r \sum_{s'} p(s', r | s, a)$$

$$\mu_\pi(s) = \lim_{t\to\infty} P[S_s = s | A_{0:t-1} \sim \pi]$$

- predpokladá sa ergodický MDP
    - z ľubovoľného stavu možno prejsť do ľubovoľného stavu (nemusí bezprostredne v jednom kroku)

Priemerná odmena – použitie

- $r(\pi)$ reprezentuje kvalitu politiky π
- zotriedenie politík podľa dosiahnutej $r(\pi)$
- za optimálnu politiku bude považovaná tá, ktorá dosahuje maximálne $r(\pi)$

TD pre kontinuálne úlohy

- diferenčná odmena $G_t = R_{t+1} - r(\pi) + R_{t+2} - r(\pi) + \ldots$
- diferenčné hodnotové funkcie
  $v_\pi(s) = E_\pi[G_t | S_t = s]$, $q_\pi(s, a) = E_\pi[G_t | S_t = s, A_t = a]$
- diferenčná forma TD chyby
  $$\delta_t = \left(R_{t+1} - \overline{R_t} + \hat{v}(S_{t+1}, \overline{w}_t)\right) - \hat{v}(S_t, \overline{w}_t)$$
  $$\delta_t = \left(R_{t+1} - \overline{R_t} + \hat{q}(S_{t+1}, A_{t+1}, \overline{w}_t)\right) - \hat{q}(S_t, A_t, \overline{w}_t)$$
  kde $\overline{R_t}$ je odhad $r(\pi)$ v čase t
- aktualizácia parametrického vektora aproximátora
  $$\overline{w}_{t+1} = \overline{w}_t + \alpha \delta_t \nabla \hat{q}(S_t, A_t, \overline{w}_t)$$

Algoritmus diferencálna Sarsa

Differential semi-gradient Sarsa for estimating $\hat{q} \approx q_*$

Input: a differentiable action-value function parameterization $\hat{q} : \mathcal{S} \times \mathcal{A} \times \mathbb{R}^d \to \mathbb{R}$
Algorithm parameters: step sizes $\alpha, \beta > 0$
Initialize value-function weights $\mathbf{w} \in \mathbb{R}^d$ arbitrarily (e.g., $\mathbf{w} = \mathbf{0}$)
Initialize average reward estimate $\bar{R} \in \mathbb{R}$ arbitrarily (e.g., $\bar{R} = 0$)

Initialize state S, and action A
Loop for each step:
    Take action A, observe R, S'
    Choose A' as a function of $\hat{q}(S', \cdot, \mathbf{w})$ (e.g., ε-greedy)
    $\delta \leftarrow R - \bar{R} + \hat{q}(S', A', \mathbf{w}) - \hat{q}(S, A, \mathbf{w})$
    $\bar{R} \leftarrow \bar{R} + \beta\delta$
    $\mathbf{w} \leftarrow \mathbf{w} + \alpha\delta\nabla\hat{q}(S, A, \mathbf{w})$
    $S \leftarrow S'$
    $A \leftarrow A'$

Zdroj: Sutton-Barto: Reinforcement Learning, 2nd ed., 2018

Triáda nestability

- aproximácia hodnotovej funkcie
  - zovšeobecnenie v priestore stavov (neuvažovanie stavov oddelene, ich vzájomné ovplyvňovanie)
- bootstrapping
  - aktualizácia hodnôt založená na aktuálnom stave týchto hodnôt
- off-policy učenie
  - rozdielnosť medzi cieľovou a exploračnou politikou

Dávkový vs inkrementálny prístup

- minimalizácia chyby pre jeden stav alebo pár (stav, akcia) – inkrementálny prístup
  - SGD
  - oprava pre jeden stav znamená pokazenie pre iný stav
  - preto pohyb iba malý kúsok v smere opravy
  - skúsenosť sa po jednorazovom použití zahodí
- minimalizácia chyby pre viac stavov alebo párov (stav, akcia) – dávkový prístup
  - LS (least squares) algoritmy
  - SGD + opakované prehrávanie
    - skúsenosť sa nezahadzuje ale sa pamätá
    - z pamätnej skúsenosti sa vyberá vzorka (dávka)
    - skúsenosť môže byť opakovane vzorkovaná do rôznych dávok