Revize minulých rozhodnutí a jejich optimalita

Jan Zeman

AS UTIA

21. listopadu 2011

Rozhodovací úloha

Systém - část světa, která je zajímavá pro rozhodovače

Rozhodovač - člověk nebo stroj s **cíli** zaměřenými na systém

Rozhodnutí u_t je navrženo rozhodovačem a aplikováno na systém

Výstup y_t pozorování systému dostupná rozhodovači

Účelová funkce g_t vyjadřuje míru dosažení cíle

Cíl najít způsob, jak zvolit rozhodnutí $\{u_1, \ldots, u_T\}$, aby maximalizovala $\sum_{k=1}^{T} g_k$.

Znalost v čase t je

$$\mathbb{P}_{:t} = \{u_1, \ldots, u_{t-1}\} \cup \{y_1, \ldots, y_t\}.$$

Hledáme tedy pravidla ve tvaru

$$\pi_t : \mathbb{P}_{:t} \to u_t.$$

Suboptimální strategie

K návrhu rozhodnutí používáme maximální dostupnou znalost:

$$u_t = \pi_t(\mathbb{P}_{:t})$$

Zkusme udělat revizi v čase $t + 1$:

$$u_t = \pi_t(\mathbb{P}_{:t+1})$$

Takto lze revidovat všechna minulá rozohdnutí:

$$\begin{array}{ll} t = 1 & \pi_1(\mathbb{P}_{:1}), \\ t = 2 & \pi_1(\mathbb{P}_{:2}), \pi_2(\mathbb{P}_{:2}), \\ t = 3 & \pi_1(\mathbb{P}_{:3}), \pi_2(\mathbb{P}_{:3}), \pi_3(\mathbb{P}_{:3}), \\ & \vdots \\ t = t & \pi_1(\mathbb{P}_{:t}), \pi_2(\mathbb{P}_{:t}), \pi_3(\mathbb{P}_{:t}), \ldots, \pi_t(\mathbb{P}_{:t}). \end{array}$$

tyto posloupnosti rozhodovacích pravidel *suboptimální strategie*.

Suboptimální strategie - algoritmus

V čase t_1 mám data a minulá rozhodnutí:

$$\mathbb{P}_{:t_1} = \{u_1, \dots, u_{t_1-1}\} \cup \{y_1, \dots, y_{t_1}\}.$$

Chci použít dynamické programování:

$$\mathcal{V}_{t_1+1}(\mathbb{P}_{:t_1+1}) = 0,$$

$$\mathcal{V}_t(\mathbb{P}_{:t}) = \max_{u_t} \mathbb{E}[g_t + \mathcal{V}_{t+1}(\mathbb{P}_{:t+1}) | \mathbb{P}_{:t_1}, u_t],$$

$$\pi_t(\mathbb{P}_{:t}) = \arg\max_{u_t} \mathbb{E}[g_t + \mathcal{V}_{t_1+1}(\mathbb{P}_{:t+1}) | \mathbb{P}_{:t_1}, u_t],$$

Takto navrhnu 2 posloupnosti funkcí:

$$\mathcal{V}_{t_1+1}, \mathcal{V}_{t_1}, \mathcal{V}_{t_1-1}, \dots, \mathcal{V}_1$$

$$\pi_{t_1}, \pi_{t_1-1}, \dots, \pi_1$$

Suboptimální strategie - předpoklady

Pracujeme za následujících silných omezení:

P1 - Otevřená smyčka: rozhodnutí u_t nemají vliv na výstupy y_t

P2 - Vyčíslitelnost: $g_t : \mathbb{P}_{:t} \to \mathbb{R}$

P3 - Konečný vliv: $g_t : \mathbb{P}_{t-n:t} \to \mathbb{R}$

Předpoklady P1 a P2 jsou potřeba k napočítávání revizí. Navíc z nich plyne, že pro výpočet revize nejsou potřeba naše minulá rozhodnutí.

Předpoklad P3 je potřeba k rozhodnutí o optimalitě.

Suboptimální strategie - 2 různě dlouhé

Mějme 2 různé suboptimální strategie $t_2 > t_1$:

$$
\begin{array}{lllll}
\mathcal{V}_1(\mathbb{P}_{:t_1}) \ldots \mathcal{V}_{t_1}(\mathbb{P}_{:t_1}) & \mathcal{V}_{t_1+1}(\mathbb{P}_{:t_1}) = 0 & & & \\
\mathcal{V}_1(\mathbb{P}_{:t_2}) \ldots \mathcal{V}_{t_1}(\mathbb{P}_{:t_2}) & \mathcal{V}_{t_1+1}(\mathbb{P}_{:t_2}) & \ldots \mathcal{V}_{t_2}(\mathbb{P}_{:t_2}) & \mathcal{V}_{t_2+1}(\mathbb{P}_{:t_2}) = 0
\end{array}
$$

Může nám výsledek pro t_1 (tu kratší) říci něco o delší úloze pro t_2?

Suboptimální strategie - 2 různě dlouhé

Mějme 2 různé suboptimální strategie $t_2 > t_1$:

$$\mathcal{V}_1(\mathbb{P}_{:t_1}) \ldots \mathcal{V}_{t_1}(\mathbb{P}_{:t_1}) \quad \mathcal{V}_{t_1+1}(\mathbb{P}_{:t_1}) = 0$$
$$\mathcal{V}_1(\mathbb{P}_{:t_2}) \ldots \mathcal{V}_{t_1}(\mathbb{P}_{:t_2}) \quad \mathcal{V}_{t_1+1}(\mathbb{P}_{:t_2}) \qquad \ldots \mathcal{V}_{t_2}(\mathbb{P}_{:t_2}) \quad \mathcal{V}_{t_2+1}(\mathbb{P}_{:t_2}) = 0$$

Může nám výsledek pro t_1 (tu kratší) říci něco o delší úloze pro t_2? Ano, ale je nutné hledat propojení úloh. Toto propojení je v Bellmanově funkci $\mathcal{V}_{t_1+1}(.)$:

- V úloze s t_1 kroky je volena: $\mathcal{V}_{t_1+1}(.)$.
- V úloze s t_2 kroky je počítána rekurzivně a je obvykle nenulová $\mathcal{V}_{t_1+1}(.)$.

Suboptimální strategie - algoritmus

V čase t_1 mám data a minulá rozhodnutí:

$$\mathbb{P}_{:t_1} = \{u_1, \ldots, u_{t_1-1}\} \cup \{y_1, \ldots, y_{t_1}\}.$$

Chci použít dynamické programování:

$$\mathcal{V}_{t_1+1}(\mathbb{P}_{:t_1+1}) = 0,$$

$$\mathcal{V}_t(\mathbb{P}_{:t}) = \max_{u_t} \mathbb{E}[g_t + \mathcal{V}_{t+1}(\mathbb{P}_{:t+1}) | \mathbb{P}_{:t_1}, u_t],$$

$$\pi_t(\mathbb{P}_{:t}) = \arg\max_{u_t} \mathbb{E}[g_t + \mathcal{V}_{t_1+1}(\mathbb{P}_{:t+1}) | \mathbb{P}_{:t_1}, u_t],$$

Takto navrhnu 2 posloupnosti funkcí:

$$\mathcal{V}_{t_1+1}, \mathcal{V}_{t_1}, \mathcal{V}_{t_1-1}, \ldots, \mathcal{V}_1$$

$$\pi_{t_1}, \pi_{t_1-1}, \ldots, \pi_1$$

V čase t_1 mám data a minulá rozhodnutí:

$$\mathbb{P}_{:t_1} = \{u_1, \dots, u_{t_1-1}\} \cup \{y_1, \dots, y_{t_1}\}.$$

Chci použít dynamické programování:

$$\mathcal{V}_{t_1+1}(\mathbb{P}_{:t_1+1}) = f,$$

$$\mathcal{V}_t(\mathbb{P}_{:t}) = \max_{u_t} \mathbb{E}[g_t + \mathcal{V}_{t+1}(\mathbb{P}_{:t+1}) | \mathbb{P}_{:t_1}, u_t],$$

$$\pi_t(\mathbb{P}_{:t}) = \arg\max_{u_t} \mathbb{E}[g_t + \mathcal{V}_{t_1+1}(\mathbb{P}_{:t+1}) | \mathbb{P}_{:t_1}, u_t],$$

Takto navrhnu 2 posloupnosti funkcí:

$$\mathcal{V}_{t_1+1}(f), \mathcal{V}_{t_1}(f), \mathcal{V}_{t_1-1}(f), \dots, \mathcal{V}_1(f)$$

$$\pi_{t_1}(f), \pi_{t_1-1}(f), \dots, \pi_1(f)$$

Suboptimální strategie - kritérium optimality

Pravidlo $\pi_i(.)$ je optimální, pokud je nezávislé na koncové podmínce f.

$$\exists \pi_i(.) \quad \forall f \quad \pi_i(., f) = \pi_i(.)$$

Neboli, ať v budoucnu přijde co přijde, toto pravidlo se nezmění.

Poznámka: Jedná se o metodu hrubé síly, která však může být překvapivě jednoduchá pro některé úlohy.

Kritérium optimality - ukázka

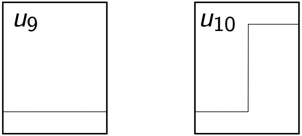
Na příkladu obchodvání máme $n = 1$ a $u_t \in \{-1, 1\}$, tzn. funkce f je udána dvěma hodnotami:

$$f: \quad \begin{aligned} f(1) &= 0, \\ f(-1) &= \alpha, \end{aligned}$$

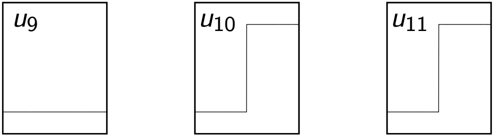
kde $\alpha \in \mathbb{R}$ - bez újmy na obecnosti.

Tedy pro náš příklad má funkce f jeden reálná parametr, u kterého musíme projít všechny hodnoty.

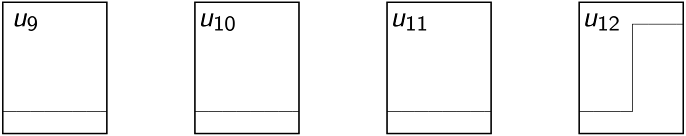
10-revision



11-revision



12-revision



Kritérium optimality - co dál?

- Kritérium říká, kdy je pravidlo optimální, může se stát, že nezvolí žádné pravidlo.
  $\Rightarrow$ Zpřesňování formulací, využívání predickí atd.
- Kritérium používá hrubou sílu.
  $\Rightarrow$ Pro příklad obchodování bude snadno formulovatelné a bude mít vazbu na skoky v datech.
- Mnoho omezení zabraňuje aplikaci na některé úlohy.
  $\Rightarrow$ Otázka uzavřené zpětné vazby.