Kratka predstavitev projekta: Iterativni izračun Nashevega ravnotežja v matričnih igrah

Eva Babnik

April, 2022

1 Nashevo ravnotežje in vrednost igre v matričnih igrah

Spodobi se, da najprej v nekaj stavkih opišem Nashevo ravnotežje ter vrednost igre v matričnih igrah za dva igralca. Matrično igro z dvema igralcema lahko predstavimo z matriko izplačil $A = (a_{ij})$, kjer prvi igralec izbere eno izmed m vrstic, drugi pa hkrati izbere enega izmed n stolpcev. $A_{i\cdot}$ naj označuje i-to vrstico, $A_{\cdot j}$ pa j - ti stolpec. Če igralca izbereta i-to vrstico in j-ti stolpec, potem drugi igralec plača prvemu igralcu a_{ij}.

Če prvi igralec izbere i-to vrstico z verjetnostjo x_i in drugi izbere j-ti stolpec z verjetostjo y_j, pri čemer velja:

$$x_i \geq 0, \tag{1}$$

$$\sum x_i = 1, \tag{2}$$

$$y_i \geq 0, \tag{3}$$

$$\sum y_i = 1. \tag{4}$$

Potem je pričakovano izplačilo prvemu igralcu $\sum\sum a_{ij}x_iy_j$. Poleg tega velja tudi:

$$\min_j \sum_i a_{ij}x_i \leq \max_i \sum_j a_{ij}y_j.$$

Trditev o minimaksu nam pove, da za neka vektorja verjetnosti $X = (x_1, \cdots, x_m)$ in $Y = (y_1, \cdots, y_n)$ v zgornji enačbi velja enakost. Tak par (X^*, Y^*) se imenuje Nashevo ravnotežje. Vrednost igre v pa je definirana kot:

$$v = \min_j \sum_i a_{ij}x_i = \max_i \sum_j a_{ij}y_j.$$

2 Iterativno računanje rešitve igre

V projektu bom implementirala več iterativnih algoritmov, ki nam vrnejo Nashevo ravnovesje in vrednost igre ter analizirala kako hitra je konvergenca posameznih metod. Za implementacijo bom uporabila programski jezik *Python*. Trenutno sem že implementirala dva algoritma, ki ju bom v nadaljevanju na kratko opisala.

2.1 Metoda I

Naj bo $V(t)$ vektor in $v_j(t)$ naj bo njegova j-ta komponenta. Označimo $\max V(t) = \max_j v_j(t)$ in $\min V(t) = min_j v_j(t)$. Naj bo sistem (U, V) sestavljen iz zaporedja n-dimenzionalnih vektorjev $U(0), U(1), \cdots$ in zaporedja m-dimenzionalnih vektorjev $V(0), V(1), \cdots$ in naj velja $\min U(0) = \max V(0)$ in $U(t+1) = U(t) + A_{i\cdot}$ ter $V(t+1) = V(t) + A_{\cdot j}$, pri čemer i in j zadoščata pogojem:

$$v_i(t) = \max V(t),\ u_j(t) = \min U(t).$$

Potem za vrednost igre v velja:

$$\lim_{t\to\infty} \frac{\min U(t)}{t} = \lim_{t\to\infty} \frac{\max V(t)}{t} = v$$

Dokaz konvergence te trditve najdemo v [1].

2.2 Metoda II

Najprej uvedimo še nekaj nove notacije. Naj velja:

$$A_i = (a_{1i}, \cdots, a_{mi}, \underbrace{0, \cdots, 0}_{n-\text{komponent}}, -1),$$
$$A_{0i} = (\underbrace{1, \cdots, 1}_{m-\text{komponent}}, \underbrace{0, \cdots, 0}_{n-\text{komponent}}, 0)$$
$$\text{za } i = 1, \cdots, n$$

in

$$A_i = (\underbrace{0, \cdots, 0}_{m-\text{komponent}}, -a_{i-n,1}, \cdots, -a_{i-n,n}, 1),$$
$$A_{0i} = (\underbrace{0, \cdots, 0}_{m-\text{komponent}}, \underbrace{1, \cdots, 1}_{n-\text{komponent}}, 0)$$
$$\text{za } i = n+1, \cdots, m+n.$$

Definirajmo še vektor, ki predstavlja rešitev igre: $Z^* = (X^*, Y^*, v)$. Z^* mora poleg 1, 2, 3, 4, ustrezati še pogoju:

$$A_i \cdot Z^* \geq 0 \mathrm{za} i = 1, \cdots, m+n. \tag{5}$$

Metoda se začne s poljubnim vektorjem $Z^{(1)}$, ki zadošča 1, 2, 3 in 4. Sedaj predpostavimo, da smo prišli na k-ti korak iteracije, in dobili vektor $Z^{(k)}$, ki ustreza 1, 2, 3 in 4. Če velja tudi 5, je $Z^{(k)}$ rešitev igre in smo zaključili. Sicer pa naj bo j_k tak indeks, da bo veljalo $A_{j_k} \cdot Z^{(k)} \leq A_i \cdot Z^{(k)}$ za vse $i = 1, \cdots, m+n$. Če obstaja več takih indeksov, lahko poljubno izberemo. Če torej poznamo indeks j_k, lahko dobimo nov vektor $\bar{Z}^{(k+1)} = (\bar{X}^{(k+1)}, \bar{Y}^{(k+1)}, \bar{v}^{(k+1)})$ na sledeči način:

$$\bar{Z}^{(k+1)} = Z^{(k)} + \alpha B_{j_k} + \beta B_{0j_k},$$

kjer je

$$\begin{aligned}
\alpha &= -Z^{(k)} \cdot B_{j_k}[1 - \cos^2\theta_{j_k}]^{-1},\\
\beta &= b_{0j_k} - [Z^{(k)} + \alpha B_{j_k}] \cdot B_{0j_k},\\
b_{0j_k} &= \frac{1}{(A_{0j_k} \cdot A_{0j_k})^{1/2}},\\
B_{j_k} &= \frac{A_{j_k}}{(A_{j_k} \cdot A_{j_k})^{1/2}},\\
B_{0j_k} &= \frac{A_{0j_k}}{(A_{0j_k} \cdot A_{0j_k})^{1/2}}
\end{aligned}$$

in

$$\cos\theta_{j_k} = \frac{A_{0j_k} \cdot A_{j_k}}{(A_{0j_k} \cdot A_{0j_k})^{1/2}(A_{j_k} \cdot A_{j_k})^{1/2}}.$$

Sedaj predpostavimo, da velja $j_k < (n+1)$. Če $\bar{Z}^{k+1}$ ustreza 5, potem nastavimo $Z^{k+1} = \bar{Z}^{k+1}$, v nasprotnem primeru pa moramo, da dobimo Z^{k+1}, narediti še nekaj korakov. Najprej vse negativne x-komponente vektorja $\bar{Z}^{k+1}$ nastavimo na 0. Predpostavimo, da so $\bar{x}_1^{(k+1)}, \cdots, \bar{x}_r^{(k+1)}$, $r < m$ negativne komponente vektorja $\bar{Z}^{(k+1)}$. Nato izračunamo vse vsote $\bar{x}_i^{(k+1)} + \frac{\sum_{i=1}^{r} \bar{x}_i^{(k+1)}}{m-r}$ za $i = r+1, \cdots, m$. Za vsak tak i, za katerega je vsota negativna, nastavimo $x_i^{(k+1)} = 0$. Če nobena vsota ni negativna, lahko tvorimo preostanek vektorja $Z^{(k+1)}$. Spet predpostavimo, da so nekatere vsote za $i = r+1, \cdots, r+s$ negativne. Ponovno izračunamo vsote $\bar{x}_i^{(k+1)} + \frac{\sum_{i=1}^{r+} \bar{x}_i^{(k+1)}}{m-(r+s)}$ za $i = r+s, \cdots, m$.. Če nobena vsota ni negativna, tvorimo preostanek vektorja $Z^{(k+1)}$, sicer pa ponavljamo zgornji postopek, dokler nobena od vsot ni negativna. Predpostavimo, da za $i = 1, \cdots, t$ velja, da je $\bar{x}_i^{(k+1)} \leq$ ali pa, da je $\bar{x}_i^{(k+1)}$ tak, da je zanj katera od zgoraj definiranih vsot negativna. Potem lahko vektor $Z^{(k+1)}$ tvorimo na sledeči način:

$$\begin{aligned}
x_1^{(k+1)} = \cdots = x_t^{(k+1)} &= 0,\\
x_i^{(k+1)} &= \bar{x}^{(k+1)} + \frac{\sum_{i=1}^{t} \bar{x}_i^{(k+1)}}{m-t} \ \text{za}\ i = t+1, \cdots, m,\\
y_j^{(k+1)} &= \bar{y}_j^{(k+1)} \ \text{za}\ j = 1, \cdots, n,\\
v^{(k+1)} &= \bar{v}^{(k+1)}.
\end{aligned}$$

Podrobnejšo izpeljavo algoritma in dokaz konvergence najedemo v [2].

3 Načrt nadaljnega dela

V nadaljevanju želim implementirati še kakšen iterativen algoritem ter algortime nato med seboj primerjati. Prav tako želim

Literatura

[1] J. Robinson, *An Iterative Method of Solving a Game*, Annals of Mathematics, **1951** strani od 296 do 301. Dostopno na: https://www.jstor.org/stable/1969530

[2] R. J. Jirka, *An iterative method for finding a solution to a zero-sum two person rectangular game*, **1959**.