# No-regret dinamika

## Bevezetés

Kutatómunkám során a no-regret algoritmusokkal és no-regret dinamikával foglalkoztam.
A beszámoló az első részében definiáljuk a regret fogalmát és a no-regret algoritmust egy játékos esetén. A második részben bebizonyítjuk, hogy létezik no-regret algoritmus, a harmadik rész pedig a no-regret dinamikáról szól, ahol már a több játékos esetet vizsgáljuk.

## No-regret algoritmus

Először az egyszemélyes esettel foglalkozunk, egy játékos a „természet" (ellenfél) ellen. Ilyen például a banki befektetés.
Legyen A a játékos lehetséges cselekedeteinek halmaza, $|A| = n$, $n \geq 2$
Minden $t = 1, 2, \ldots, T$ lépésben

- a játékos választ egy $p^t$ kevert stratégiát, azaz egy valószínűségi eloszlást a lehetséges cselekedetein
- az ellenfél választ egy $c^t : A \longrightarrow [-1, 1]$ költségvektort (ez a játékos választása után történik)
- $a^t$ cselekedet kiválasztása $p^t$ alapján. A játékos megtudja $c^t$ költségvektort, a költsége $c^t(a^t)$.

A játékos célja, hogy minimalizálja a költségeit. A nehézség az, hogy a $c^t$ vektor a $p^t$ kevert stratégia után kerül kiválasztásra.
A célunk egy jó online algoritmus megtalálása a problémára.

**Definíció.** *Online algoritmus*ról beszélünk, ha az algoritmus futás közben folyamatosan kap új input-értékeket (ellentétben az *offline algoritmus*sal, ahol az egész input rendelkezésre áll kezdettől fogva).

Megpróbálhatnánk az algoritmusunk költségét a cselekedetek lehetséges legjobb sorozatához, azaz $\sum_{t=1}^{T} \min_{a \in A} c^t(a)$ értékhez hasonlítani, de ekkor problémába ütköznénk, ugyanis tekintsük a következő példát: legyen a lehetséges cselekedeteink száma 2 és nézzünk egy tetszőleges algoritmust. Minden $t$ lépésben legyen a nagyobb valószínűséggel választott cselekedet költsége 1, a másiké 0. Ekkor a várható költségünk legalább $\frac{T}{2}$, még a lehető legolcsóbb cselekedet-sorozat költsége 0 lesz. Tehát ha az algoritmusunk jóságát a

$\sum_{t=1}^{T} \min_{a\in A} c^t(a)$ értékhez szeretnénk viszonyítani, akkor „túl sokat” fogunk várni az algoritmusunktól. Ezért inkább a „gyengébb”, $\min_{a\in A} \sum_{t=1}^{T} c^t(a)$ értékhez fogunk viszonyítani, azaz a legjobb rögzített cselekedethez.

**Definíció.** Rögzített $c^1, c^2, \ldots, c^T$ költségek mellett az $a^1, a^2, \ldots, a^T$ cselekedetek sorozatának *regret*-je az

$$\frac{1}{T}\left[\sum_{t=1}^{T} c^t(a^t) - \min_{a\in A}\sum_{t=1}^{T} c^t(a)\right]$$

érték.
Tehát informálisan ez azt jelenti, hogy mennyivel jártunk volna jobban, ha a választott $a^1, a^2, \ldots, a^T$ cselekedetek helyett minden lépésben $a$-t választjuk.

**Definíció.** *No-regret algoritmus*

- az ellenfél inputként kapja $t$-t, az algoritmus által választott $p^1, p^2, \ldots, p^t$ kevert stratégiákat, az $a^1, a^2, \ldots, a^{t-1}$ cselekedeteket és ebből meghatároz egy $c^t$ költségvektort.
- no-regret algoritmusról beszélünk, ha $\forall\varepsilon > 0$-hoz létezik egy elég nagy $T = T(\varepsilon)$, hogy minden ellenfélre a regret legfeljebb $\varepsilon$.

**Megfigyelés.** Az algoritmusnak randomizáltnak kell lennie.
Ha determinisztikus algoritmussal próbálkoznánk, azaz minden lépésben egy rögzített $a^t$ cselekedetet hajtanánk végre, akkor az ellenfél (tudva az algoritmus stratégiáját) megtehetné, hogy minden lépésben a választott cselekedet költségét 1-re, minden más cselekedet költségét 0-ra állítja. Ezzel biztosítaná, hogy minden $t$ időpillanatban a lehető legköltségesebb cselekedetet hajtsuk végre. Az algoritmusunk költsége így $T$ lenne, de a lehető legjobb rögzített cselekedet költsége legfeljebb $\frac{T}{n}$. Ha csak két cselekedet van, akkor is a regret legalább $\frac{1}{2}$, tetszőleges nagy $T$-re.

A következő példa azt mutatja, hogy van egy alsó korlát arra, hogy a regret milyen gyorsan tarthat 0-hoz.
**Példa.** Legyen $n = 2$ a lehetséges cselekedetek száma és minden $c^t$ költségvektor legyen $\frac{1}{2}$ - $\frac{1}{2}$ valószínűséggel $(0,1)$ vagy $(1,0)$. Ekkor bármely online algoritmus várható költsége a $t$-edik lépésben $\frac{1}{2}$, a várható összköltség $\frac{T}{2}$. Viszont a legjobb rögzített cselekedet várható költsége csak $\frac{T}{2} - b\sqrt{T}$, ahol $b$ egy $T$-től független konstans. Ennek az oka, hogy a $T$-szer ismételt költségvektor-választás binomiális eloszlást követ, mindkét vektort várhatóan $\frac{T}{2}$ -szer választjuk és a szórás $\frac{1}{2}\sqrt{T}$. Így tetszőleges online algoritmusra lesz

olyan költségvektor, amire a várható regret legalább $b\sqrt{T}$. Mivel ez az alsó korlát igaz már véletlen költségvektor-választás esetén is, természetesen igaz lesz minden, az ellenfél által „tudatosan" választott költségvektorra is.
Hasonlóan megmutatható, hogy $n$ lehetséges cselekedet esetén a várható regret $b\sqrt{\ln n/T}$, ahol $b > 0$ egy $T$-től és $n$-től független konstans.

**Tétel (No-regret algoritmus létezése).** Ha $n$ a lehetséges cselekedetek száma és $T \geq 4 \ln n$, akkor létezik online algoritmus, amire a várható regret legfeljebb $2\sqrt{\ln n/T}$ bármilyen ellenfél esetén.

Egy ilyen algoritmus a *Multiplicative weights*, a következő részben ezt mutatjuk be. Az elemzésből kapjuk majd a tétel bizonyítását.

## Multiplicative weights algoritmus

Az algoritmus működési elve azon alapul, hogy a cselekedetekhez súlyokat rendelünk, és a súlyozás határozza meg, hogy egy $t$ lépésben melyik cselekedetet választjuk. Minden lépésben az algoritmus egy cselekedetet a súlyának arányával azonos valószínűséggel választ. A súlyok csak csökkenhetnek, a csökkenés mértéke attól függ, hogy az adott cselekedet költsége mekkora volt a korábbiakban.
Tehát formálisan:

1. Kezdetben állítsuk minden cselekedet súlyát 1-re, azaz legyen $w^1(a) = 1$ $\forall a \in A$-ra.

2. $t = 1, 2, \ldots, T$-re:

   - Legyen $p^t := \frac{w^t}{\Gamma^t}$, ahol $\Gamma^t = \sum_{a \in A} w^t(a)$, cselekedjünk $p^t$ szerint.
   - A $c^t$ költségek megismerése után változtassuk a súlyokat a következőképpen: $w^{t+1}(a) := w^t(a)(1 - \varepsilon c^t(a))$ $\forall a \in A$-ra.

Itt $\varepsilon$ egy 0 és $\frac{1}{2}$ közötti érték, pontos értékét később választjuk meg.

**Elemzés**
Elég, ha azt az esetet vizsgáljuk, amikor a $c^1, \ldots, c^T$ költségvektorok előre rögzítettek. Ennek az az oka, hogy az algoritmus működése nem függ a korábban választott cselekedetektől, csak a költségvektoroktól.
Legyen $A$ a lehetséges cselekedetek halmaza, $|A| = n$ és $T \geq 4 \ln n$. Rögzítsük a $c^1, c^2, \ldots, c^T$ költségvektorokat.
Legyen $\nu^t$ az algoritmus várható költsége a $t$-edik lépésben, azaz:

$$\nu^t = \sum_{a \in A} p^t(a) c^t(a) = \sum_{a \in A} \frac{w^t(a)}{\Gamma^t} c^t(a).$$

A célunk, hogy kapjunk egy felső becslést $\sum_{t=1}^{T} \nu^t$-re. Könnyen látható, hogy

$$\Gamma^{t+1} = \sum_{a \in A} w^{t+1}(a) = \sum_{a \in A} w^t(a)(1 - \varepsilon c^t(a)) = \Gamma^t(1 - \varepsilon \nu^t).$$

Ez felülről becsülhető a következőképpen:

$$\Gamma^{t+1} \leq \Gamma^t e^{-\varepsilon \nu^t},$$

mivel $1 + x \leq e^x$ minden valós $x$-re.
Így kapjuk, hogy

$$\Gamma^{T+1} \leq \Gamma^1 \prod_{t=1}^{T} e^{-\varepsilon \nu^t} = n e^{-\varepsilon \sum_{t=1}^{T} \nu^t}.$$

Itt $\Gamma^1 = n$, mivel $w^1(a) = 1 \ \forall a \in A$.
Legyen $a^*$ a legjobb rögzített cselekedet és legyen $OPT := \sum_{t=1}^{T} c^t(a^*)$. Ekkor

$$\Gamma^{T+1} \geq w^{T+1}(a^*) = w^1(a^*) \prod_{t=1}^{T}(1 - \varepsilon c^t(a^*)) = \prod_{t=1}^{T}(1 - \varepsilon c^t(a^*)).$$

Felhasználva az $1 - x \geq e^{-x-x^2}$, ha $|x| \leq \frac{1}{2}$ összefüggést ($\ln(1-x) = -x - \frac{x^2}{2} - \frac{x^3}{3} - \frac{x^4}{4} - \ldots$ Taylor-sorfejtésből), a következőt kapjuk:

$$\Gamma^{T+1} \geq \prod_{t=1}^{T} e^{-\varepsilon c^t(a^*) - \varepsilon^2 (c^t(a^*))^2} \geq e^{-\varepsilon OPT - \varepsilon^2 T}.$$

Az utolsó egyenlőtlenség azért teljesül, mert $(c^t(a^*))^2 \leq 1 \ \forall t$.
A két becslést összevetve

$$n e^{-\varepsilon \sum_{t=1}^{T} \nu^t} \geq \Gamma^{T+1} \geq e^{-\varepsilon OPT - \varepsilon^2 T}$$

adódik. Mindkét oldal természetes alapú logaritmusát véve és $-\varepsilon$-nal leosztva kapjuk, hogy

$$\sum_{t=1}^{T} \nu^t \leq OPT + \varepsilon T + \frac{\ln n}{\varepsilon}.$$

Most $\varepsilon = \sqrt{\ln n / T}$ választással, ha $T \geq 4 \ln n$, akkor $0 \leq \varepsilon \leq \frac{1}{2}$. Tehát az algoritmus várható költsége legfeljebb $2\sqrt{T \ln n}$-vel több, mint a legjobb rögzített cselekedet költsége, amivel a no-regret algoritmus létezésére vonatkozó tételt beláttuk.

## No-regret dinamika

Most áttérünk annak a vizsgálatára, hogy mi mondható el több játékos esetén, véges, költség-minimalizáló játékokban.

**Definíció (No-regret dinamika).** Minden $t = 1, 2, \ldots, T$ lépésben:

1. Minden játékos választ egy kevert stratégiát valamilyen no-regret algoritmus alapján. Az $i$. játékos stratégiája: $p_i^t$.
2. Minden játékos kap egy $c_i^t$ költségvektort, ahol $c_i^t(s_i)$ az $s_i$ tiszta stratégia várható költsége a többi játékos választott kevert stratégiáira nézve, azaz:
$$c_i^t(s_i) = \mathbf{E}_{\mathbf{s}_{-i}^t \sim \sigma_{-i}^t}[C_i(s_i, \mathbf{s}_{-i}^t)],$$
ahol $\sigma_{-i}^t = \prod_{j \neq i} p_j^t$.

A játékosok a $c_i^t$ költségvektorokat használják a no-regret algoritmusukban a súlyok módosítására.

**Definíció.** Egy $\sigma$ eloszlás az $S_1 \times \cdots \times S_k$ lehetséges kimenetelek halmazán egy költség-minimalizáló játékban *gyenge korrelált egyensúly*, ha $\forall i \in \{1, 2, \ldots, k\}$ játékosra és $\forall s_i' \in S_i$-re

$$\mathbf{E}_{\mathbf{s} \sim \sigma}[C_i(\mathbf{s})] \leq \mathbf{E}_{\mathbf{s} \sim \sigma}[C_i(s_i', \mathbf{s}_{-i})].$$

**Tétel.** Tegyük fel, hogy egy költség-minimalizáló játékban $k$ játékos esetén $T$ lépés után minden játékos várható regret-je legfeljebb $\varepsilon$.
Legyen $\sigma^t = \prod_{i=1}^k p_i^t$ az eloszlás a lehetséges kimeneteleken a $t$-edik lépésben és $\sigma = \frac{1}{T} \sum_{t=1}^T \sigma^t$. Ekkor $\sigma$ gyenge korrelált egyensúlyhoz tart olyan értelemben, hogy

$$\mathbf{E}_{\mathbf{s} \sim \sigma}[C_i(\mathbf{s})] \leq \mathbf{E}_{\mathbf{s} \sim \sigma}[C_i(s_i', \mathbf{s}_{-i})] + \varepsilon$$

minden $i$ játékosra és $s_i' \in S_i$-re.
*Bizonyítás.* Minden $i$ játékosra

$$\mathbf{E}_{\mathbf{s} \sim \sigma}[C_i(\mathbf{s})] = \frac{1}{T} \sum_{t=1}^T \mathbf{E}_{\mathbf{s} \sim \sigma^t}[C_i(\mathbf{s})]$$

és

$$\mathbf{E}_{\mathbf{s} \sim \sigma}[C_i(s_i', \mathbf{s}_{-i})] = \frac{1}{T} \sum_{t=1}^T \mathbf{E}_{\mathbf{s} \sim \sigma^t}[C_i(s_i', \mathbf{s}_{-i})]$$

$\sigma$ definíciója alapján.
Az első egyenlőség jobb oldalán szereplő érték az $i$ játékos várható költsége, ha a no-regret algoritmusa szerint cselekszik, a második egyenlőség jobb oldalán szereplő érték pedig a várható költség egy rögzített $s'_i$ cselekedetre nézve. A tétel feltétele szerint mindenkinek legfeljebb $\varepsilon$ a regret-je, tehát a no-regret algoritmus szerinti érték legfeljebb $\varepsilon$-nal lehet több. Ezzel a tételt beláttuk.

# Hivatkozások

[1] Roughgarden, T. (2016). Twenty Lectures on Algorithmic Game Theory. Cambridge: Cambridge University Press.

[2] Shoham, Y. - Leyton-Brown, K. (2008). Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations. Cambridge University Press, USA.