博弈论第十三讲

授课时间: 2021 年 12 月 10 日 授课教师: 张家琳 记录人: 梁伟 叶尔哈孜

1 均衡概念的等级

当我们谈论一个博弈的均衡时，我们通常会考虑以下四个方面：

- 均衡是否存在；
- 如果均衡存在，能否找到一个算法来计算求解均衡；
- 均衡的效率怎么样；
- 对均衡的质量进行评估。

考虑均衡存在性的问题，纯策略纳什均衡（PNE）在一个博弈中不一定存在，而混合策略纳什均衡（MNE）、相关均衡（CE）和粗相关均衡（CCE）在一个博弈中一定存在。

混合策略纳什均衡（MNE）是一个非常有用的均衡概念，但是在现实生活中我们很难能够在多项式时间内将它求解出来。为此，我们引入了相关均衡（CE）和粗相关均衡（CCE）的概念。粗相关均衡所包含的范围要大于相关均衡所包含的范围，二者均可以在多项式时间内找到有效的算法来进行求解。均衡概念的等级图为：

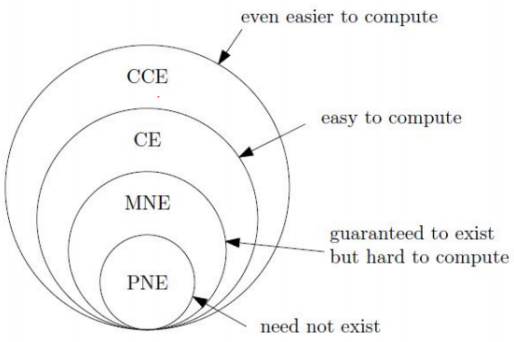

Figure: Hierarchy of equilibrium concepts

2 遗憾最小化问题

遗憾最小化问题（Regret Minimization problem）是一类不完全信息下的多轮决策问题，目标是设计策略使得最后的代价尽可能接近在完全信息下可以达到的最优值，其具体定义如下：

定义 1. 遗憾最小化问题是一个 T 轮决策问题，在任意第 t 轮中：

- 一个玩家（*player*）会在他的行动集合 A 中选择一个混合策略 p^t；

- 玩家在选择策略时相应会产生一个代价函数 $c^t : A \to [0,1]$；
- 由于混合策略的存在，玩家会根据混合策略 p^t 选择一个动作 a^t；
- 本轮玩家的代价为 $c^t(a^t)$；
- 在第 t 轮结束之后，玩家会知道整个代价函数 c^t。

玩家的目标是优化每一步的策略，最小化总代价 $cost = \sum_{t=1}^{T} c^t(a^t)$。但是，为了更精确地衡量每一步优化策略的好坏，我们还需要引入一个作为对比的参照函数。这个参照函数描述为：

假定我们在上述问题的 T 轮迭代过程中，在行动集合 A 中选择一个固定策略（fixed action）a，使得整个 T 轮迭代过程中玩家所付出的代价 $\sum_{t=1}^{T} c^t(a)$ 最小。于是，我们对总轮次 T 求平均，可以得到整个问题的优化目标为最小化后悔值：

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

3 无悔算法

无悔算法（No-regret algorithm）有时也叫做参考专家意见，能够不断根据当前结果进行修正和改进以达到最优值。其性质定义为：

定义 2. 对于在线决策算法，在 T 轮决策问题中，一个对手（*adversary*）在第 t 轮所定义的代价函数 c^t 要参考此前轮次（包括本轮）的所有概率分布 $p^1, \ldots, p^t$ 以及此前轮次的动作代价成本 $c^1(a^1), \ldots, c^{t-1}(a^{t-1})$。如果这个算法是无悔算法，那么该算法满足以下性质：

- 对于任意对手（*adversary*），当总轮次 T 趋向于无穷大时，该算法期望的后悔值为 $o(1)$（其实就是趋向于 0）。

乘积性权值算法（Multiplicative weights algorithm）是一种简单而自然的学习算法。它有两个原则：一是过去行为的表现会对即将选择的行为产生影响；二是一个坏的行为应该被严厉惩罚，以指数形式下降。其具体定义为：

定义 3. 算法为每一个动作 a 赋予了一个可信度权重，决定其概率分布。权重只会降低，并取决于该动作的代价成本。初始化，预置一个参数 $\eta \in (0,1)$，且对于 $\forall a \in A$，$w^0(a) = 1$。在第 $t+1$ 轮中：

- 对于所有 $a \in A$，根据第 t 轮的代价函数 c^t 更新权值 $w^{t+1}(a) = w^t(a)(1-\eta)^{c^t(a)}$；
- 根据 $w^{t+1}(a)$ 的值来更新第 $t+1$ 轮的概率分布：

$$p^{t+1}(a) = \frac{w^{t+1}(a)}{\sum\limits_{a' \in A} w^{t+1}(a')}$$

定理 1. 乘积性权值算法是一个无悔算法。

下面我们分三步来对这个定理进行证明, 记 $W^t = \sum_{a\in A} w^t(a)$。

Step 1. 首先证明 W^{T+1} 并不会特别小。

为了方便计算，我们先设 $Algo = \sum_{t=1}^{T} c^t(a^t)$，$OPT = \min_{a\in A} \sum_{t=1}^{T} c^t(a)$，那么我们的后悔值可以表示为：$regret = (Algo - OPT)/T$。令 $a^* = \arg\min_{a\in A} \sum_{t=1}^{T} c^t(a)$，则有

$$W^{T+1} = \sum_{a\in A} w^{T+1}(a) \geq w^{T+1}(a^*) \tag{1}$$

$$= w^0(a^*) \prod_{t=1}^{T} (1-\eta)^{c^t(a^*)} \tag{2}$$

$$= w^0(a^*)(1-\eta)^{\sum_{t=1}^{T} c^t(a^*)} \tag{3}$$

$$= (1-\eta)^{OPT} \tag{4}$$

Step 2. 其次证明在某一轮 t 的迭代过程中，如果该轮的期望动作代价成本 $\mathbb{E}\left(c^t(a^t)\right)$ 很大，那么从 t 轮到 $t+1$ 轮的归一化系数 $W^t \to W^{t+1}$ 会下降很快。

令 $cost^t = \mathbb{E}\left[c^t(a^t)\right]$ 来表示第 t 轮的期望动作代价成本，则根据期望的定义有：

$$cost^t = \mathbb{E}\left[c^t(a^t)\right] \tag{5}$$

$$= \sum_{a\in A} p^t(a) c^t(a) \tag{6}$$

$$= \sum_{a\in A} \frac{w^t(a)}{W^t} c^t(a) \tag{7}$$

接下来我们借助不等式 $(1-x)^y \leq (1-xy)$ 对 W^{t+1} 进行放缩：

$$W^{t+1} = \sum_{a\in A} w^{t+1}(a) \tag{8}$$

$$= \sum_{a\in A} w^t(a)(1-\eta)^{c^t(a)} \tag{9}$$

$$\leq \sum_{a\in A} w^t(a)(1-\eta c^t(a)) \tag{10}$$

$$\leq W^t - \eta W^t cost^t \tag{11}$$

$$= W^t(1-\eta cost^t) \tag{12}$$

由 (12) 式可得，当第 t 轮期望动作代价成本 $\mathbb{E}\left(c^t(a^t)\right)$ 很大时，即 $cost^t$ 很大时，从 t 轮到 $t+1$ 轮的归一化系数 $W^t \to W^{t+1}$ 会下降很快，证毕。

Step 3. 在第一步和第二步的证明中，我们得到以下结论：

$$(1-\eta)^{OPT} \leq W^{T+1} \leq W^0 \cdot \prod_{t=1}^{T} (1-\eta cost^t)$$

假设 m 为行动集合 A 中的个数，则 $W^0 = m$，那么有：

$$(1-\eta)^{OPT} \leq W^0 \cdot \prod_{t=1}^{T} (1-\eta cost^t) \tag{13}$$

$$= m \cdot \prod_{t=1}^{T} (1-\eta cost^t) \tag{14}$$

不等式两边同时取对数有：

$$OPT \cdot \ln(1-\eta) \leq \ln m + \sum_{t=1}^{T} \ln(1-\eta cost^t) \tag{15}$$

利用不等式 $\ln(1-x) \leq -x$ 对上述式子进行放缩有：

$$OPT \cdot \ln(1-\eta) \leq \ln m - \sum_{t=1}^{T} \eta cost^t \tag{16}$$

计算 Algo 的期望:

$$\mathbb{E}(Algo) = \mathbb{E}\left(\sum_{t=1}^{T} c^t(a^t)\right) = \sum_{t=1}^{T} cost^t \tag{17}$$

$$\leq \frac{1}{\eta}\left[\ln m - OPT \cdot \ln(1-\eta)\right] \tag{18}$$

利用不等式 $-x - x^2 \leq \ln(1-x)$ 对上述式子进行放缩有：

$$\mathbb{E}(Algo) \leq \frac{1}{\eta}\left[\ln m + OPT \cdot (\eta + \eta^2)\right] \tag{19}$$

$$= \frac{\ln m}{\eta} + OPT \cdot (1+\eta) \tag{20}$$

因此，后悔值 regret 的期望为：

$$\mathbb{E}(regret) = \frac{\mathbb{E}(Algo - OPT)}{T} \leq \frac{OPT \cdot \eta + \frac{\ln m}{\eta}}{T} \tag{21}$$

在定义代价函数 c^t 时，我们规定 $\forall t \in T$，都有 $c^t \in [0,1]$，则 $OPT = \min\limits_{a \in A} \sum\limits_{t=1}^{T} c^t(a) \leq T$，故

$$\mathbb{E}(regret) \leq \eta + \frac{\ln m}{T} \cdot \frac{1}{\eta} \tag{22}$$

取 $\eta = \sqrt{\frac{\ln m}{T}}$, 上述不等式可写为：

$$\mathbb{E}(regret) \leq 2\sqrt{\frac{\ln m}{T}} \tag{23}$$

综上，我们可以看出当 $T \to \infty$ 时，该算法的期望后悔值 $\mathbb{E}(regret)$ 趋向于 0。因此，乘积性权值算法是一个无悔算法。证毕！