Fictitious Self-Play in Extensive-Form Games

Johannes Heinrich, Marc Lanctot, David Silver.
ICML 2015
报告人：汪永毅

1. Fictitious Play

两人猜拳：A出招（👊），B反制（👋），A再反制（✌️），B也反制（👊）.....

问题：策略不能收敛到Nash均衡，而是在石头剪刀布之间轮转。

解决方法：每次对对手的历史平均策略取最优对策，则双方的历史平均策略收敛到Nash均衡。⇒ **Fictitious Play**

适用范围：双人零和博弈、位势博弈 (Potential game)：
$\exists f \in \mathbb{R}^A, \forall i : f(a_i', a_{-i}) - f(a_i, a_{-i}) = u_i(a_i', a_{-i}) - u_i(a_i, a_{-i})$ .....

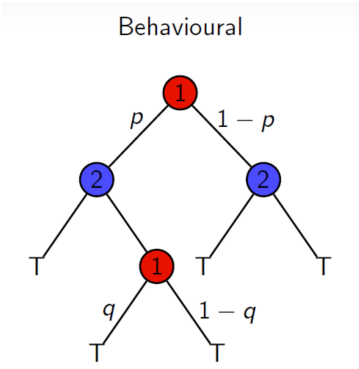
2. XFP (Full Width Extensive-Form Fictitious Play)

Fictitious play的局限性：只能用于Normal-form game，即矩阵博弈。

如何用于Extensive-form game（扩展型博弈）：转为矩阵表示？丢失时序信息、混合策略表示消耗空间过大。

⇒ 改变策略表示方式，使用Behavioural strategy（行为策略）来替代展开为矩阵后的混合策略表示。

扩展型博弈的混合策略与行为策略：

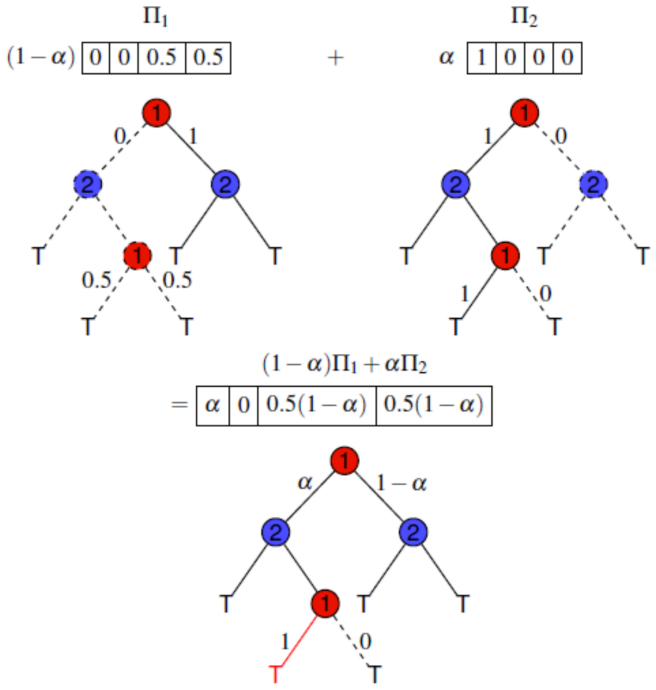


Mixed

pq	$p(1-q)$	$1-p$	0

行为策略：信息集下动作的概率分布
混合策略：路径的概率分布（仅考虑某一玩家）

⇒ 行为策略比混合策略更容易表示



混合策略可直接进行线性组合

⇒ 混合策略可直接线性组合，而行为策略不可（逐点混合会导致出现两种策略下都不会产生的路径）

使用行为策略表示计算策略线性组合：行为策略和混合策略可一一对应，结果等价。

设Π, B是玩家i的两个混合策略，则与$\mathrm{M} := \lambda_1\Pi + \lambda_2\mathrm{B}$（$\lambda_1 + \lambda_2 = 1$）等价的行为策略为：

$$\mu(u)(a) = \frac{\lambda_1 x_\pi(\sigma_u)\pi(u)(a) + \lambda_2 x_\beta(\sigma_u)\beta(u)(a)}{\lambda_1 x_\pi(\sigma_u) + \lambda_2 x_\beta(\sigma_u)} \tag{1}$$

其中u为i的信息集，a为该信息集下的动作；$x_\pi(\sigma_u), x_\beta(\sigma_u)$分别表示行为策略$\pi, \beta$下到达$u$的概率（仅考虑$i$的决策）。

XFP算法：

Algorithm 1 Full-width extensive-form fictitious play

function FICTITIOUSPLAY(Γ)
  Initialize π_1 arbitrarily
  $j \leftarrow 1$
  while within computational budget **do**
    $\beta_{j+1} \leftarrow$ COMPUTEBRS(π_j)
    $\pi_{j+1} \leftarrow$ UPDATEAVGSTRATEGIES(π_j, β_{j+1})
    $j \leftarrow j + 1$
  end while
  return π_j
end function

function COMPUTEBRS(π)
  Recursively parse the game's state tree to compute a best response strategy profile, $\beta \in b(\pi)$.
  return β
end function

function UPDATEAVGSTRATEGIES(π_j, β_{j+1})
  Compute an updated strategy profile π_{j+1} according to Theorem 7.
  return π_{j+1}
end function

注：定理7即使用行为策略表示，计算最优对策和平均策略的线性组合
组合系数可随迭代步变化，但需满足一定条件才可收敛

3. FSP (Fictitious Self-Play)

XFP存在的问题：XFP需要对每个状态递归计算出精确的最优对策，难以用于大的博弈树。

FSP针对XFP的改进：分别用**强化学习**和**监督学习**方法代替XFP中**ComputeBRs**和**UpdateAVGStrategies**.

使用强化学习方法(Fitted Q Iteration, FQI)来近似计算最优对策β，监督学习方法（统计加权转移频数）计算平均策略π.

Algorithm 2 General Fictitious Self-Play

function FICTITIOUSSELFPLAY(Γ, n, m)
  Initialize completely mixed π_1
  $\beta_2 \leftarrow \pi_1$
  $j \leftarrow 2$
  while within computational budget **do**
    $\eta_j \leftarrow$ MIXINGPARAMETER(j)
    $\mathcal{D} \leftarrow$ GENERATEDATA($\pi_{j-1}, \beta_j, n, m, \eta_j$)
    for each player $i \in \mathcal{N}$ **do**
      $\mathcal{M}^i_{RL} \leftarrow$ UPDATERLMEMORY($\mathcal{M}^i_{RL}, \mathcal{D}^i$)
      $\mathcal{M}^i_{SL} \leftarrow$ UPDATESLMEMORY($\mathcal{M}^i_{SL}, \mathcal{D}^i$)
      $\beta^i_{j+1} \leftarrow$ REINFORCEMENTLEARNING($\mathcal{M}^i_{RL}$)
      $\pi^i_j \leftarrow$ SUPERVISEDLEARNING($\mathcal{M}^i_{SL}$)
    end for
    $j \leftarrow j + 1$
  end while
  return π_{j-1}
end function

function GENERATEDATA(π, β, n, m, η)
  $\sigma \leftarrow (1 - \eta)\pi + \eta\beta$
  $\mathcal{D} \leftarrow n$ episodes $\{t_k\}_{1 \le k \le n}$, sampled from strategy profile σ
  for each player $i \in \mathcal{N}$ **do**
    $\mathcal{D}^i \leftarrow m$ episodes $\{t^i_k\}_{1 \le k \le m}$, sampled from strategy profile (β^i, σ^{-i})
    $\mathcal{D}^i \leftarrow \mathcal{D}^i \cup \mathcal{D}$
  end for
  return $\{\mathcal{D}^k\}_{1 \le k \le N}$
end function

Tuple Memory一部分用于保存对手决策，RL计算最佳应答
另一部分用于保存自身决策，SL计算自己的历史平均策略

4. 总结:

FSP的意义在于提供了一种理想的自对弈RL训练框架，用以近似求解部分扩展式博弈的Nash均衡。

优点：适用于较大规模的扩展式博弈，使用近似代替XFP的递归求解，提高了效率。

缺点：对一般的扩展式博弈，不保证收敛性。