强化学习-策略梯度

主讲TA：陈梓烨

2020.12.18

强化学习方法分类

- 基于值函数
  - ➢ 动态规划算法：p(s'|s,a)和r(s,a,s')已知，通过优化值函数来找最优策略
    - ✓ 策略迭代：先根据贝尔曼方程更新值函数，再改进策略
    - ✓ 值迭代：直接根据贝尔曼**最优**方程更新值函数
  - ➢ 蒙特卡罗方法：p(s'|s,a)和r(s,a,s')未知，需采样多条轨迹来估计Q函数
  - ➢ 时序差分学习方法：结合前两种
    - ✓ SARSA：同策略
    - ✓ Q学习（以及DQN）：异策略
- 基于策略函数

强化学习方法分类

- 基于值函数
  - ➢ 动态规划算法：p(s'|s,a)和r(s,a,s')已知，通过优化值函数来找最优策略
    - ✓ 策略迭代：先根据贝尔曼方程更新值函数，再改进策略
    - ✓ 值迭代：直接根据贝尔曼**最优**方程更新值函数
  - ➢ 蒙特卡罗方法：p(s'|s,a)和r(s,a,s')未知，需采样多条轨迹来估计Q函数
  - ➢ 时序差分学习方法：结合前两种
    - ✓ SARSA：同策略
    - ✓ Q学习（以及DQN）：异策略
- 基于策略函数：在策略空间直接搜索来得到最佳策略
  - ➢ 策略梯度
    - ✓ REINFORCE算法
    - ✓ 演员-评论员算法

用参数网络学习策略，输出动作的概率

- 连续状态及动作
- 随机性策略

策略梯度

强化学习的目标函数:

$$\mathcal{J}(\theta) = \mathbb{E}_{\tau \sim p_\theta(\tau)}[G(\tau)] = \mathbb{E}_{\tau \sim p_\theta(\tau)}[\sum_{t=0}^{T-1} \gamma^t r_{t+1}]$$

目标函数$\mathcal{J}(\theta)$ 关于策略参数θ 的导数为:

$$\begin{aligned}
\frac{\partial \mathcal{J}(\theta)}{\partial \theta} &= \frac{\partial}{\partial \theta} \int p_\theta(\tau) G(\tau) \mathrm{d}\tau \\
&= \int \left(\frac{\partial}{\partial \theta} p_\theta(\tau) \right) G(\tau) \mathrm{d}\tau \\
&= \int p_\theta(\tau) \left(\frac{1}{p_\theta(\tau)} \frac{\partial}{\partial \theta} p_\theta(\tau) \right) G(\tau) \mathrm{d}\tau \\
&= \int p_\theta(\tau) \left(\frac{\partial}{\partial \theta} \log p_\theta(\tau) \right) G(\tau) \mathrm{d}\tau \\
&= \mathbb{E}_{\tau \sim p_\theta(\tau)} \left[\frac{\partial}{\partial \theta} \log p_\theta(\tau) G(\tau) \right],
\end{aligned}$$

策略梯度

$\frac{\partial}{\partial\theta}\log p_\theta(\tau)$可以进一步分解为

$$\begin{aligned}\frac{\partial}{\partial\theta}\log p_\theta(\tau) &= \frac{\partial}{\partial\theta}\log\left(p(s_0)\prod_{t=0}^{T-1}\pi_\theta(a_t|s_t)p(s_{t+1}|s_t,a_t)\right)\\ &= \frac{\partial}{\partial\theta}\left(\log p(s_0)+\sum_{t=0}^{T-1}\log\pi_\theta(a_t|s_t)+\sum_{t=0}^{T-1}\log p(s_{t+1}|s_t,a_t)\right)\\ &= \sum_{t=0}^{T-1}\frac{\partial}{\partial\theta}\log\pi_\theta(a_t|s_t).\end{aligned}$$

可以看出，$\frac{\partial}{\partial\theta}\log p_\theta(\tau)$是和状态转移概率无关，只和策略函数相关.

策略梯度

因此，策略梯度 $\frac{\partial \mathcal{J}(\theta)}{\partial \theta}$ 可写为

$$\begin{aligned}\frac{\partial \mathcal{J}(\theta)}{\partial \theta} &= \mathbb{E}_{\tau \sim p_\theta(\tau)} \left[\left(\sum_{t=0}^{T-1} \frac{\partial}{\partial \theta} \log \pi_\theta(a_t|s_t) \right) G(\tau) \right] \\ &= \mathbb{E}_{\tau \sim p_\theta(\tau)} \left[\left(\sum_{t=0}^{T-1} \frac{\partial}{\partial \theta} \log \pi_\theta(a_t|s_t) \right) \left(G(\tau_{0:t}) + \gamma^t G(\tau_{t:T}) \right) \right] \\ &= \mathbb{E}_{\tau \sim p_\theta(\tau)} \left[\sum_{t=0}^{T-1} \left(\frac{\partial}{\partial \theta} \log \pi_\theta(a_t|s_t) \gamma^t G(\tau_{t:T}) \right) \right],\end{aligned}$$

其中 $G(\tau_{t:T})$ 为从时刻 t 作为起始时刻收到的总回报

$$G(\tau_{t:T}) = \sum_{t'=t}^{T-1} \gamma^{t'-t} r_{t'+1}.$$

REINFORCE 算法

采用随机游走方法采集多个轨迹:

$$\frac{\partial \mathcal{J}(\theta)}{\partial \theta} = \mathbb{E}_{\tau \sim p_\theta(\tau)} \left[\sum_{t=0}^{T-1} \left(\frac{\partial}{\partial \theta} \log \pi_\theta(a_t|s_t) \gamma^t G(\tau_{t:T}) \right) \right],$$

$$\frac{\partial \mathcal{J}(\theta)}{\partial \theta} \approx \frac{1}{N} \sum_{n=1}^{N} \left(\sum_{t=0}^{T-1} \frac{\partial}{\partial \theta} \log \pi_\theta(a_t^{(n)}|s_t^{(n)}) \gamma^t G_{\tau_{t:T}^{(n)}} \right).$$

REINFORCE 算法

算法 14.6　REINFORCE 算法

输入: 状态空间 $\mathcal{S}$, 动作空间 $\mathcal{A}$, 可微分的策略函数 $\pi_\theta(a|s)$, 折扣率 γ, 学习率 α;

随机初始化参数 θ;

repeat

　根据策略 $\pi_\theta(a|s)$ 生成一条轨迹: $\tau = s_0, a_0, s_1, a_1, \cdots, s_{T-1}, a_{T-1}, s_T$;

　for t=0 **to** T **do**

　　计算 $G(\tau_{t:T})$;

　　$\theta \leftarrow \theta + \alpha\gamma^t G(\tau_{t:T}) \frac{\partial}{\partial\theta} \log \pi_\theta(a_t|s_t)$;　　// 更新策略函数参数

　end

until π_θ 收敛;

输出: 策略 π_θ

演员-评论员算法

结合**策略梯度**和**时序差分学习**

- 演员：指策略函数$\pi_\theta(a|s)$，学习一个策略来得到尽可能高的回报。
- 评论员：指值函数$V_\phi(s)$，，对当前策略的状态值函数进行估计，即评估演员的好坏。

演员-评论员算法可以进行**单步**更新参数，不需要等到回合结束才进行更新。

演员-评论员算法

假设从时刻t 开始的回报$G(\tau_{t:T})$，我们用下面公式近似计算：

$$\hat{G}(\tau_{t:T}) = r_{t+1} + \gamma V_{\phi}(s_{t+1}),$$

在每步更新中，分别进行策略函数$\pi_{\theta}(s, a)$ 和值函数$V_{\phi}(s)$ 的学习。

一方面，更新参数ϕ使得值函数$V_{\phi}(s_t)$接近于估计的真实回报$\hat{G}(\tau_{t:T})$，即

$$\min_{\phi} \left(\hat{G}(\tau_{t:T}) - V_{\phi}(s_t) \right)^2,$$

另一方面，将值函数$V_{\phi}(s_t)$ 作为基线函数来更新参数θ，减少策略梯度的方差，即

$$\theta \leftarrow \theta + \alpha \gamma^t \left(\hat{G}(\tau_{t:T}) - V_{\phi}(s_t) \right) \frac{\partial}{\partial \theta} \log \pi_{\theta}(a_t | s_t).$$

演员-评论员算法

算法 14.8　演员-评论员算法

输入: 状态空间 $\mathcal{S}$, 动作空间 $\mathcal{A}$, 可微分的策略函数 $\pi_\theta(a|s)$, 可微分的状态值函数 $V_\phi(s)$, 折扣率 γ, 学习率 $\alpha > 0, \beta > 0$;

随机初始化参数 θ, ϕ;

repeat

  初始化起始状态 s; $\lambda = 1$;

  repeat

    在状态 s, 选择动作 $a = \pi_\theta(a|s)$;

    执行动作 a, 得到即时奖励 r 和新状态 s';

    $\delta \leftarrow r + \gamma V_\phi(s') - V_\phi(s)$;

    $\phi \leftarrow \phi + \beta\delta \frac{\partial}{\partial \phi} V_\phi(s)$;　　// 更新值函数参数

    $\theta \leftarrow \theta + \alpha\lambda\delta \frac{\partial}{\partial \theta} \log \pi_\theta(a|s)$;　　// 更新策略函数参数

    $\lambda \leftarrow \gamma\lambda$;

    $s \leftarrow s'$;

  until s 为终止状态;

until θ 收敛;

输出: 策略 π_θ

强化学习总结

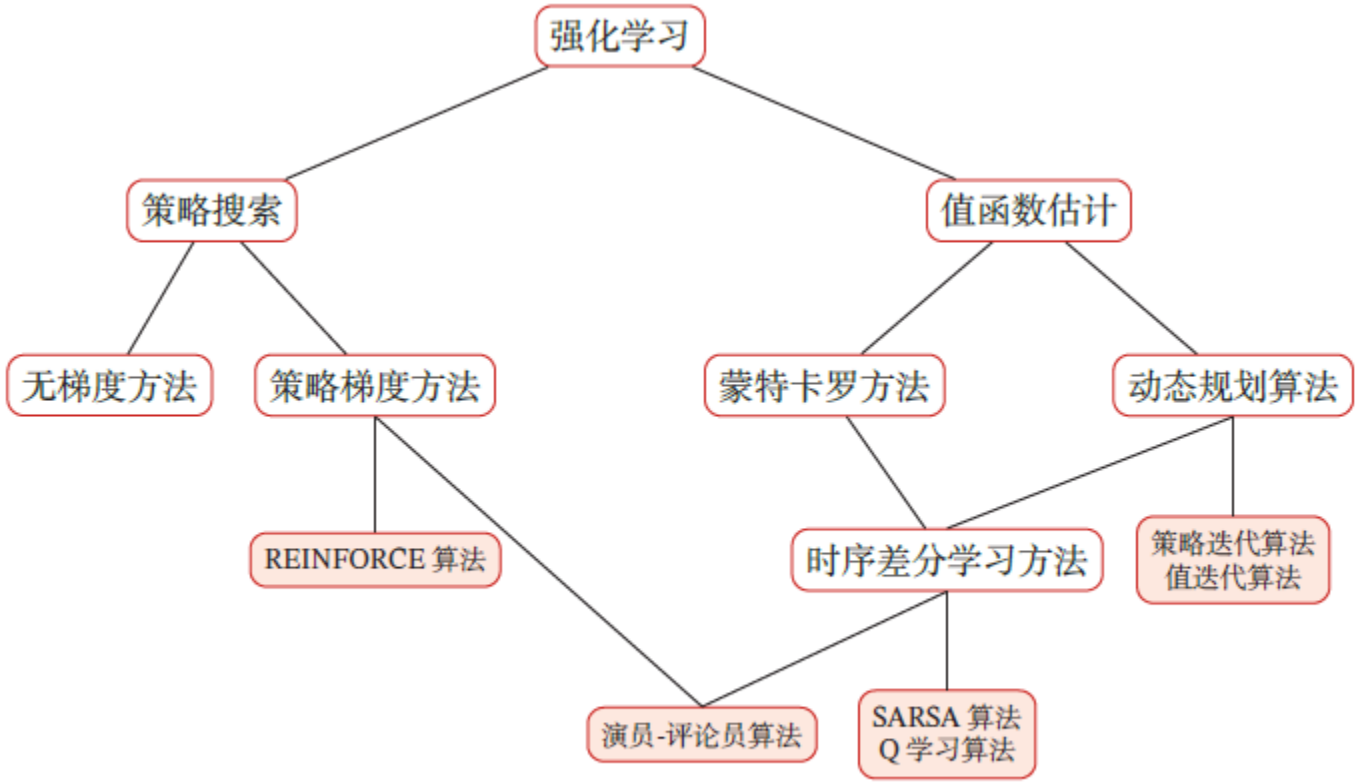


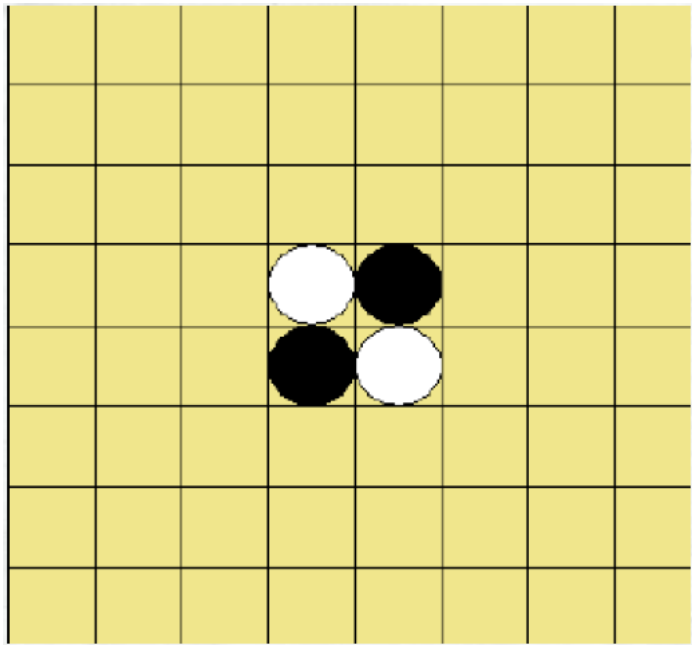
图 14.4　不同强化学习算法之间的关系

期末Project

实现8*8黑白棋的人机对战，要求：

- 横排、竖排、对角线均可翻转。
- 要求使用强化学习方法。
- 评价函数不限。

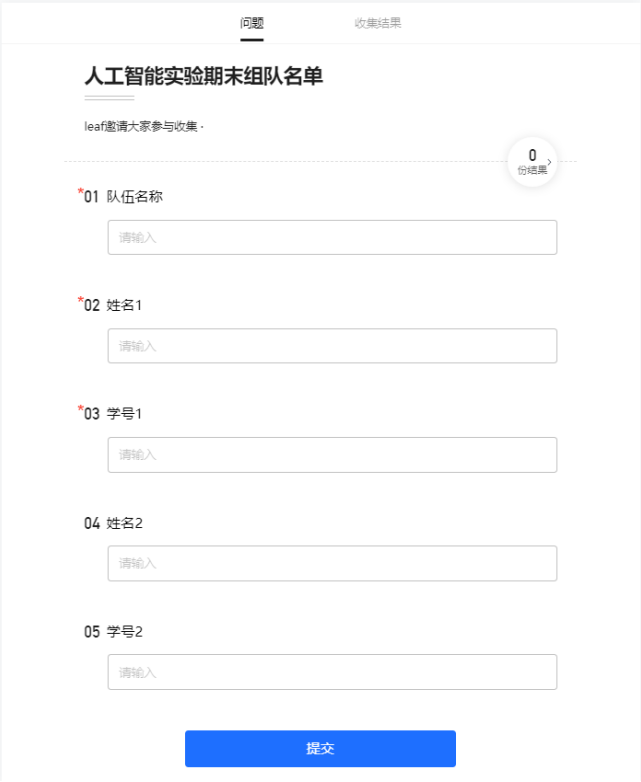
期末Project



初始状态如图所示，要求有电脑先手和电脑后手两种模式

组队

- 一组人数1~2人，提交队伍名字，共同评分，推荐组队完成
- 12月25日0点前提交组队名单：
  https://docs.qq.com/form/page/DT0ltVUpaZE9naEhV?_w_tencentdocx_form=1



评分标准

- 实验报告：80%
- Rank：20%

Rank

- 方式：分为四大组，组内进行车轮战对弈，四组积分第一进行决赛
- 时间：1月8日实验课

报告要求

- 实验原理
- 实现过程
- 实验结果分析
- 创新点