控制与决策
Control and Decision

基于优势约束扩散策略的离线强化学习

王雪松, 张恒瑞, 张佳志, 程玉虎

引用本文:
王雪松, 张恒瑞, 张佳志, 等. 基于优势约束扩散策略的离线强化学习[J]. 控制与决策, 2025, 40(6): 1903-1912.

在线阅读 View online: https://doi.org/10.13195/j.kzyjc.2024.0618

基于优势约束扩散策略的离线强化学习

王雪松，张恒瑞，张佳志，程玉虎†

(中国矿业大学 信息与控制工程学院，江苏 徐州 221116)

摘　要: 离线强化学习旨在从静态的经验数据集中学习策略, 这种数据驱动的学习范式为强化学习在现实世界的应用提供了极大可能. 然而, 离线数据集通常由不同水平的策略收集而来, 其动作分布呈现出一种难以表达的多峰状态. 此外, 离线数据集中的高回报轨迹通常较为稀缺, 导致策略学习效率低下. 为此, 提出一种基于优势约束扩散策略的离线强化学习方法. 首先, 利用扩散模型的反向扩散步骤生成策略, 以更好地拟合多峰动作分布; 然后, 在策略提升阶段, 使用优势函数进行策略约束以帮助智能体更加专注于数量稀少的高回报轨迹, 并分别针对连续控制任务和稀疏奖励导航任务构建两种特定优势函数. 在 bandit 任务和 D4RL 基准上的实验结果表明: 所提方法能有效缓解行为策略表达能力受限及高回报轨迹稀缺的问题, 在大多数任务上获得最高的归一化得分.

关键词: 离线强化学习；扩散模型；多峰动作分布；策略约束；高回报轨迹；优势函数

中图分类号: TP18　　**文献标志码:** A

DOI: 10.13195/j.kzyjc.2024.0618

引用格式: 王雪松, 张恒瑞, 张佳志, 等. 基于优势约束扩散策略的离线强化学习 [J]. 控制与决策, 2025, 40(6): 1903-1912.

Offline reinforcement learning based on advantage-constrained diffusion policy

WANG Xue-song，*ZHANG Heng-rui*，*ZHANG Jia-zhi*，*CHENG Yu-hu*†

(School of Information and Control Engineering，China University of Mining and Technology，Xuzhou 221116，China)

Abstract: The goal of offline reinforcement learning is to learn policies from static datasets of previously collected experience. This data-driven learning paradigm greatly expands the potential for applying reinforcement learning in real-world scenarios. However, offline static datasets are often collected from policies of varying quality, leading to a multimodal action distribution that is difficult to model effectively. Furthermore, high-return trajectories are typically scarce within these datasets, which reduces the efficiency of policy learning. To address these challenges, this paper introduces an offline reinforcement learning method based on advantage-constrained diffusion policy. The diffusion model's reverse process is utilized to generate policies that better capture the multimodal distribution. During policy improvement, an advantage function is applied to constrain the policy, directing the agent's focus on the sparse high-return trajectories. Two specific advantage functions are designed for continuous control tasks and sparse reward navigation tasks. Experimental results on bandit tasks and D4RL benchmarks show that the proposed method successfully mitigates limitations in behavior policy expressiveness and the scarcity of high-return trajectories, achieving the highest normalized scores in most tasks.

Keywords: offline reinforcement learning； diffusion model； multimodal action distribution； policy constraint； high-return trajectories；advantage function

0　引　言

近年来, 强化学习 (RL) 迅速发展, 尤其是在视频游戏[1-2]、自动驾驶[3] 以及机器人领域[4-6] 拥有出色的表现. 在线强化学习通过不断与环境交互以收集奖励, 进而优化策略. 然而, 在线交互效率低下, 并且在应用于真实世界时会带来极大的安全隐患. 因此, 强化学习的研究重心逐渐由在线范式转向离线范式.

离线强化学习 (ORL) 也被称为批强化学习, 是

收稿日期: 2024-05-22；**录用日期:** 2024-10-07.

基金项目: 国家自然科学基金项目 (62373364, 62176259)；江苏省重点研发计划项目 (BE2022095).

责任编委: 卢剑权.

†通信作者. E-mail: chengyuhu@163.com.

一种从预先收集的离线数据集中学习策略的机器学习范式. 然而, 由于缺乏与环境的交互, 直接对离线数据集使用标准的策略提升方法会对分布外动作错误地乐观估计, 从而产生外推误差, 最终导致学习过程失败[7]. 为了缓解外推误差, 现有的方法大致可以分为 4 类: 1) 策略约束. 在策略提升阶段, 显式或隐式地约束习得策略偏离行为策略的程度[8-10]. 2) 值正则化. 将正则项嵌入到策略评估步骤中, 通过为分布外动作分配相对较低的Q值来学习值函数[11-12]. 3) 序列建模. 将离线强化学习重构为序列问题, 直接进行轨迹生成[13-15]. 4) 基于模型. 首先学习一个环境动力学模型, 随后在马尔科夫决策过程中执行悲观方法[16-18].

策略约束型离线强化学习方法通过直接或间接地限制策略搜索空间, 使策略能够快速收敛. 相比于其他方法, 策略约束方法在执行策略时会满足一定的强约束条件, 因而更具安全性. 现有的策略约束型离线强化学习方法通常假定行为策略是一个高斯分布. 然而, 离线强化学习所使用的数据集通常是多策略的混合数据集, 其对应行为策略的概率分布通常呈现出一种多峰状态, 简单地使用单峰的高斯分布无法对其准确地建模. 为此, Wang 等[19] 提出使用扩散模型以表达行为策略 (DiffusionQL), 迫使习得策略逼近更准确的行为策略分布. 但由于恒定的策略约束不仅限制了危险的动作, 也限制了正确的动作, DiffusionQL 在面对高回报轨迹稀缺的数据集时表现不佳. 虽然可以使用优先经验回放来频繁采样具有高回报的轨迹[20], 但由于智能体的训练缺乏低回报数据的惩罚, 表现出较大的方差. 那么该如何在利用高表达能力的生成模型捕获策略的同时, 又能促使策略对于高回报轨迹进行偏好学习呢?

本文提出一种基于优势约束扩散策略的离线强化学习 (ORL-ACDP) 框架. 首先, 分析优势函数对于策略学习的指导意义, 在保证行为策略被高度表达后, 最优策略正比于优势函数. 然后, 分别针对连续控制任务和稀疏奖励导航任务设计两种特定优势函数. 在连续控制任务中, 利用基于动作拼接的优势函数约束扩散策略, 使得策略选取具有更高优势的动作; 在稀疏奖励导航任务中, 利用基于状态拼接的优势函数约束扩散策略, 使得策略所做动作到达更高价值的状态, 从而达到终点. 最后, 在 bandit 任务和 D4RL 基准测试上进行详细的性能测试, 结果表明所提方法不仅在高轨迹回报稀缺的数据集上可以获得明显优于基线算法的归一化得分, 在其余数据集上也同样具有优异的性能表现.

1 基于优势约束扩散策略的离线强化学习

1.1 问题描述

强化学习通常被形式化为一个由 6 元组$\mathcal{M}=\{\mathcal{S},\mathcal{A},\mathcal{P},\mathcal{R},\gamma,d_0\}$构成的马尔科夫决策过程, 其中$m$维状态空间$\mathcal{S}\subset\mathbb{R}^m$, n维动作空间$\mathcal{A}\subset\mathbb{R}^n$, 环境动力学$\mathcal{P}:\mathcal{S}\times\mathcal{A}\to\mathcal{S}$, 奖励函数$\mathcal{R}:\mathcal{S}\times\mathcal{A}\to\mathbb{R}$, 折扣因子$\gamma\in[0,1)$, 以及初始状态分布为$d_0$. 强化学习的目标是学习一个策略$\pi(a|s)$以最大化折扣累计期望回报$\mathbb{E}_\pi\Big[\sum\limits_{t=0}^{\infty}\gamma^t\mathcal{R}(s_t,a_t)\Big]$. 状态-动作值函数$Q^\pi(s,a)=\mathbb{E}_\pi\Big[\sum\limits_{t=0}^{\infty}\gamma^t\mathcal{R}(s_t,a_t)|s_0=s,a_0=a\Big]$表示从状态$s$和动作$a$开始, 采用策略$\pi$继续执行直到终止状态所能获得的折扣累计期望回报. 与状态-动作值函数Q^π类似, 状态值函数$V^\pi(s)=\mathbb{E}_\pi\Big[\sum\limits_{t=0}^{\infty}\gamma^t\mathcal{R}(s_t,a_t)|s_0=s\Big]$表示从状态$s$开始, 策略$\pi$可以获得的折扣累计期望回报. 在离线强化学习的设定中, 智能体不依赖于环境交互, 仅通过由行为策略π_β交互形成的静态数据集$\mathcal{B}=\{(s,a,r,s')\}$来学习策略$\pi(a|s)$. 由于不允许与环境交互, 离线 Actor-Critic 方法使用经验贝尔曼算子$\widetilde{\mathcal{T}}^\pi$更新状态-动作值函数$Q^\pi(s,a)$和状态值函数$V^\pi(s)$. 其中: $(\widetilde{\mathcal{T}}^\pi Q)(s,a)=r+\gamma\mathbb{E}_{s'\sim\mathcal{B},a'\sim\pi(\cdot|s')}[Q(s',a')]$,$(\widetilde{\mathcal{T}}^\pi V)(s)=r+\gamma\mathbb{E}_{s'\sim\mathcal{B}}[V(s')]$.

1.2 基于优势约束扩散策略的策略提升

在标准的策略约束型离线强化学习方法中, 策略正则项的权重是一个恒定不变的常数标量, 这意味着所有的动作都将受到相同程度的约束. 然而, 这种仅通过简单地设置一个固定的超参数来约束动作是存在缺陷的. 在策略提升阶段更新习得策略时, 不仅要考虑对于Q函数的贪婪, 还需要考虑对于习得策略约束的松紧程度. 如果约束强度是固定的, 那么过紧或过松的正则项对于智能体策略的学习都是不利的. 进一步讲, 如果策略约束过强, 则策略学习就会退化为行为克隆; 反之, 又会近似成为Q学习. 一个标准的策略约束型 Actor-Critic 框架如下:

$$\phi_i\leftarrow\arg\min_{\phi_i}\mathbb{E}_{(s,a,r,s')\sim\mathcal{B},a'\sim\pi_{\bar{\theta}}(\cdot|s')}[(Q_{\phi_i}(s,a)-(r+\gamma\min_{i=1,2}Q_{\bar{\phi}_i}(s',a')))^2],\ i=1,2;\tag{1}$$

$$\theta\leftarrow\arg\max_{\theta}\mathbb{E}_{s\sim\mathcal{B},a\sim\pi_\theta(\cdot|s)}[Q_{\phi_1}(s,a)-\lambda D(\pi_\beta(\cdot|s),\pi_\theta(\cdot|s))].\tag{2}$$

其中: Critic 和 Actor 网络分别使用ϕ和θ参数化,

$\bar{\phi}$和$\bar{\theta}$分别表示 Critic 和 Actor 目标网络参数, 超参数λ用于权衡Q值贪婪与策略约束之间的强度, $D(\cdot,\cdot)$用于度量行为策略与习得策略之间的散度. 习得策略π_θ可利用条件扩散模型的反向过程表示为

$$\pi_\theta(a|s)=p_\theta(a^{0:T}|s)=\mathcal{N}(a^T;0,I)\prod_{t=1}^{T}p_\theta(a^{t-1}|a^t,s),\quad(3)$$

其中a^0表示反向扩散链末端的生成动作, 同时也是用于 Critic 评价的动作. 经验上, $p_\theta(a^{t-1}|a^t,s)$可以被一个高斯分布$\mathcal{N}(a^{t-1};\mu_\theta(a^t,s,t),\sigma_\theta(a^t,s,t))$建模, μ_θ和σ_θ分别表示由θ参数化的均值和方差. 采用去噪扩散概率模型[21]的条件扩散模型参数化技巧, 可以将$p_\theta(a^{t-1}|a^t,s)$设置为一个噪声预测模型. 采样一个噪声动作$a^T\sim\mathcal{N}(0,I)$, 并使用扩散模型从噪声动作中不断去噪, 最终生成动作a^0, 其步骤为

$$a^{t-1}|a^t=\frac{a^t}{\sqrt{\alpha_t}}-\frac{\beta_t}{\sqrt{\alpha_t(1-\overline{\alpha}_t)}}\epsilon_\theta(a^t,s,t)+\sqrt{\beta_t}\epsilon.\quad(4)$$

其中: $t\in\{1,2,\dots,T\}$表示扩散过程时间步, a^{t-1}表示反向扩散过程中的生成动作; 温度系数β_t随着扩散时间步不断变化, $\alpha_t=1-\beta_t$, $\overline{\alpha}_t=\prod_{j=1}^{t}\alpha_j$; $\epsilon_\theta(a^t,s)$是一个可学习的噪声网络; 随机噪声$\epsilon\sim\mathcal{N}(0,I)$, 特殊地, 当$t=1$时, 将$\epsilon$设置为 0 以提高采样质量. 条件噪声模型参数$\theta$的更新目标如下:

$$\theta\leftarrow\arg\min_\theta\mathbb{E}_{t\sim\mathcal{U}(1,T),\epsilon\sim\mathcal{N}(0,I),(s,a)\sim\mathcal{B}}[(\epsilon-\epsilon_\theta(\sqrt{\overline{\alpha}_t}a+\sqrt{1-\overline{\alpha}_t}\epsilon,s,t))^2],\quad(5)$$

其中$\mathcal{U}(1,T)$表示参数为 1 和T的离散型均匀分布. 为实现策略提升, 需要在训练阶段将Q值贪婪与反向扩散结合, 即在策略提升阶段, 需要在确保 Actor 输出的动作尽可能逼近行为策略的同时, 优先学习高Q值的动作.

在固定了策略约束型 Actor-Critic 框架后, 需要考虑如何权衡策略提升中的Q值贪婪和策略约束. 过于松弛的策略约束项会导致习得策略难以收敛, 而过于保守的策略约束项则会使习得策略陷入次优. 因此, 如果能在策略提升阶段, 使智能体根据当前状态-动作对的质量来选择约束强度, 那么策略将自适应地朝最优动作更新. 本文从优势加权回归(AWR)[22]得到启发: 习得策略可以回归到最大优势所对应的行为策略. 细节如下, 对于已知的异策略优势函数$A^\pi(s,a)$和行为策略$\pi_\beta(a|s)$, 求解策略的优化问题可以表示为

$$\pi^*(a|s)=\arg\max_{a\sim\pi(\cdot|s)}[\mathbb{E}_{a\sim\pi(\cdot|s)}[A^\pi(s,a)]];$$
$$\text{s.t. } D_{\mathrm{KL}}(\pi(\cdot|s)\|\pi_\beta(\cdot|s))\leqslant d,\quad\int_a\pi(a|s)\mathrm{d}a=1.\quad(6)$$

其中d表示行为策略与习得策略的最大约束距离.

通过施加 Karush-Kuhn-Tucker 条件, 可以得到上述优化问题的解析解. 具体而言, 采用拉格朗日乘数法可以构造如下的拉格朗日函数:

$$\mathcal{L}(\pi,\eta,\delta)=\mathbb{E}_{a\sim\pi(\cdot|s)}[A^\pi(s,a)]+\eta(d-D_{\mathrm{KL}}(\pi(\cdot|s)\|\pi_\beta(\cdot|s)))+\delta\Big(1-\int_a\pi(a|s)\mathrm{d}a\Big),\quad(7)$$

其中η和δ均为拉格朗日乘数. 对π求偏导可得

$$\frac{\partial\mathcal{L}}{\partial\pi}=A^\pi(s,a)-\eta\log\pi_\beta(a|s)+\eta\log\pi(a|s)+\eta-\delta.\quad(8)$$

将$\dfrac{\partial\mathcal{L}}{\partial\pi}$设为0, 可得最优习得策略$\pi^*$的封闭解

$$\pi^*(a|s)=\frac{1}{Z(s)}\pi_\beta(a|s)\exp\Big(\frac{1}{\eta}A^\pi(s,a)\Big).\quad(9)$$

最后以解耦的形式来简化式 (9), 得到

$$\pi^*(a|s)\propto\pi_\beta(a|s)\cdot\exp(\kappa A(s,a)),\quad(10)$$

其中κ为权衡行为策略与优势函数之间权重的超参数.

式 (10) 表明, 如果想要获取一个好的策略, 需要两个要素: 一个是需要对行为策略进行高还原表达, 另一个则是促使策略选择高优势所对应的状态-动作对. 为此, 本文提出一种简单易实现的策略约束型离线强化学习方法. 具体而言, 首先使用标准的策略评估损失以更新 Critic 网络; 随后在策略提升阶段, 将非对称优势加权后的反向扩散损失嵌入 Actor 损失中. 具体地, Critic 和 Actor 网络参数更新规则为

$$\phi_i\leftarrow\arg\max_{\phi_i}\mathbb{E}_{(s,a,r,s')\sim\mathcal{B},a'\sim\pi_{\bar{\theta}}(\cdot|s')}[(Q_{\phi_i}(s,a)-(r(s,a)+\gamma\min_{i=1,2}Q_{\bar{\phi}_i}(s',a')))^2],\quad(11)$$

$$\theta\leftarrow\arg\max_\theta\mathbb{E}_{t\sim\mathcal{U}(1,T),\epsilon\sim\mathcal{N}(0,I),(s,a)\sim\mathcal{B},a^0\sim\pi_\theta(\cdot|s)}[Q_{\phi_1}(s,a^0)-\mathcal{F}(A(s,a))(\epsilon-\epsilon_\theta(\sqrt{\overline{\alpha}_t}a+\sqrt{1-\overline{\alpha}_t}\epsilon,s,t))^2].\quad(12)$$

其中: $A(s,a)$表示状态-动作对所对应的优势函数, $\mathcal{F}$表示关于优势函数$A(s,a)$的一个非对称映射.

1.3　收敛性分析

引理 1 (琴生不等式)　对于任意的凸函数$f(x)$, 函数值的期望大于等于期望的函数值恒成立, 即$\mathbb{E}[f(x)]\geqslant f(\mathbb{E}[x])$.

引理 2 (收缩映射定理)　设(X,d)是一个非空

的完备度量空间, 并且$f: X \to X$是一个收缩映射, 即存在一个常数$0 \leqslant k < 1$, 使得对于所有的$x, y \in X$, 都有$d(f(x), f(y)) \leqslant k \cdot d(x, y)$, 则$f$在$X$中有唯一不动点, 即存在唯一的$x^* \in X$使得$f(x^*) = x^*$.

定理 1 (收敛性) 设$|\mathcal{S}| < \infty$, $|\mathcal{A}| < \infty$, $|\mathcal{R}| < \infty$, 在全部$\mathcal{S} \times \mathcal{A}$空间, 对于任意策略$\pi$, 算子$(\mathcal{T}^{\pi}Q_k)(s,a) = r(s,a) + \gamma_{s' \sim \mathcal{D}, a' \sim \pi(\cdot|s)}[Q_k(s',a')]$, 在$L_{\infty}$范数上是一个$\gamma$ - 收缩算子.

证明 设$Q_1(s,a)$和$Q_2(s,a)$为两个任意的Q函数, 根据引理 1 可得

$$
\begin{aligned}
&|(\mathcal{T}^{\pi}Q_1)(s,a) - (\mathcal{T}^{\pi}Q_2)(s,a)| = \\
&|r(s,a) + \gamma \mathbb{E}_{s' \sim D, a' \sim \pi(\cdot|s)}[Q_1(s',a')] - \\
&(r(s,a) + \gamma \mathbb{E}_{s' \sim D, a' \sim \pi(\cdot|s)}[Q_2(s',a')])| \leqslant \\
&\gamma \mathbb{E}_{s' \sim D, a' \sim \pi(\cdot|s)}[|Q_1(s',a') - Q_2(s',a')|] \leqslant \\
&\gamma \|Q_1(s',a') - Q_2(s',a')\|_{\infty},
\end{aligned}
$$

由此可得算子$\mathcal{T}^{\pi}$存在唯一不动点, 本文所提方法收敛. □

1.4 优势函数构造

由于强化学习中的值函数以及策略均被参数化, 本文根据任务的不同, 提供了基于动作拼接的连续控制任务和基于状态拼接的稀疏奖励导航任务的优势函数近似方法.

为了更直接地理解基于动作拼接的优势函数, 这里观察状态-动作值函数$Q^{\pi}(s,a) = \mathbb{E}_{\pi}\Big[\sum_{t=0}^{\infty}\gamma^t R(s_t,a_t)|s_t = s, a_t = a\Big]$以及状态值函数$V^{\pi}(s) = \mathbb{E}_{\pi}\Big[\sum_{t=0}^{\infty}\gamma^t R(s_t,a_t)|s_t = s\Big]$. 显然, 状态值函数$V^{\pi}(s)$表示了在状态$s$下的平均状态-动作值函数. 因此, 可以认为基于动作拼接的优势函数$A(s,a) = Q(s,a) - V(s)$一定程度上反映了给定一个时刻t下, 状态-动作对(s,a)所对应的值函数相对于平均值函数的差异.

基于动作拼接的状态值函数和优势函数的参数化更新式如下:

$$
\begin{aligned}
\psi \leftarrow \arg\min_{\psi} &\mathbb{E}_{(s,a,r,s') \sim \mathcal{B}, a' \sim \pi_{\bar{\theta}}(\cdot|s')}[(V_{\psi}(s) - \\
&(r + \gamma \min_{i=1,2} Q_{\bar{\phi}_i}(s',a')))^2],\ i = 1,2; \qquad (13)
\end{aligned}
$$

$$
A_{\phi,\psi}(s,a) = Q_{\phi}(s,a) - V_{\psi}(s). \qquad (14)
$$

其中状态值函数$V(s)$神经网络参数为ψ.

对于基于状态拼接的优势函数, 首先需要通过使用时序差分方法对状态值函数$V(s)$近似拟合, 即

$$
\psi \leftarrow \arg\min_{\psi} \mathbb{E}_{(s,a,r,s') \sim \mathcal{B}}[(V_{\psi}(s) - (r + \gamma V_{\psi}(s')))^2]. \qquad (15)
$$

那么, 仅包含状态s的参数化优势函数$A(s)$可构造为

$$
A_{\psi}(s) = r + V_{\psi}(s') - V_{\psi}(s). \qquad (16)
$$

基于状态拼接的优势函数无需考虑动作, 而是通过当前的状态s和下一状态s'来确定优势. 该类型的优势函数对于状态轨迹型任务具有优越性, 例如导航任务.

1.5 算法实现

为了直观地展现出优势函数对于策略学习的指导作用, 本文将非对称函数$\mathcal{F}$设定为指示函数$\mathcal{I}$. 指示函数$\mathcal{I}$因其二值性, 使得智能体在策略提升阶段会更直接地侧重那些具有显著优势的状态-动作对, 继而促使策略获得更快的收敛速度和更好的性能. 具体而言, 智能体会对高优势的状态-动作对进行策略约束, 在实现Q值贪婪时, 更专注于高优势的状态-动作对. 相反, 对于低优势的状态-动作对, 智能体会忽略掉约束项, 仅专注于Q值更高的状态-动作对.

$$
\mathcal{F}(A(s,a)) = \mathcal{I}(A(s,a)) = \begin{cases} 0, & A(s,a) \leqslant 0; \\ 1, & A(s,a) > 0. \end{cases} \qquad (17)
$$

本文直接将优势通过指示函数映射, 用作动态加权, 从而迫使智能体趋于学习高回报动作. 相比于优先经验回放频繁采样高回报轨迹, 基于优势约束扩散策略的离线强化学习方法除了注重高回报的轨迹, 同样会从低回报轨迹中学习策略, 使得习得策略更具鲁棒性. 现有的优势加权算法首先将行为策略和优势函数解耦, 随后进行策略提取, 而 ACDP 则是将优势加权作为一种策略约束指导. 综上所述, ACDP 的伪代码如下所示.

算法 1 ACDP

输入: 折扣率γ, 软更新系数τ, 扩散时间步T.

初始化: Actor网络参数θ, Critic网络参数ϕ_1、ϕ_2, 状态值网络参数ψ, 目标网络参数$\bar{\theta}$、$\bar{\phi}_1$、$\bar{\phi}_2$.

for $i = 1, 2, \ldots, N$ do

小批量采样$\{s, a, r, s'\} \sim \mathcal{B}$;

根据式(4)扩散目标Actor输出动作$a'^0 \sim \pi_{\bar{\theta}}(\cdot|s')$;

根据式(11)更新Critic网络参数ϕ_1、ϕ_2;

根据式(13)或(15)更新状态值网络ψ;

根据式(14)或(16)计算优势函数A;

根据式(4)扩散Actor输出动作$a^0 \sim \pi_{\theta}(\cdot|s)$;

根据式(12)更新Actor网络参数θ:

$$
\bar{\theta} \leftarrow (1-\tau)\bar{\theta} + \tau\theta,
$$

$$
\bar{\phi}_{1,2} \leftarrow (1-\tau)\bar{\phi}_{1,2} + \tau\phi_{1,2}
$$

end for

2　实　验

2.1　实验设置

为确保对比的公平性, 所有对比离线 RL 方法的实验结果均是采用原文提供的源码和超参数得到的. ACDP 的超参数设置如表 1 所示. 实验过程中, 硬件配置为: Intel I7-13700KF, Nvidia GeForce RTX4070; 软件配置为: Windows 10, Python 3.7.12, CUDA 11.3, Pytorch 1.11.0.

表1　超参数设置

超参数	取值
网络隐藏层	2
网络隐藏层维度	256
优化器	Adam
激活函数	Mish
Critic和Actor网络学习率	3e-4
状态值网络学习率	3e-4
Critic和Actor目标网络学习率	3e-4
最大迭代数	1e+6
批大小	256
折扣率γ	0.99
软更新系数τ	0.005
最大扩散时间步T	5

2.2　bandit 实验结果与分析

对于 ACDP 而言, 扩散时间步直接影响生成行为策略的质量和计算成本. 为此, 首先以 bandit 任务为例来分析扩散时间步对 ACDP 性能的影响, 实验结果如图 1 所示. 如图 1(a) 所示, 本文采用 1 000 个带有噪声的样本点构成单位圆以生成离线 bandit 数据集, 其中噪声从高斯分布$\mathcal{N}\sim(0,0.05)$中采样得到. 噪声单位圆上的数据表示维度为 2 的动作, 其中$a\in[-1,1]$, 且所有动作都对应数值为 1 的正奖励. 在给定状态下, 有多个动作可以获得相同的正奖励, 因此该离线数据集表现出多模态性. 图 1 的实验结果表明: 1) 随着扩散时间步T的增大, 扩散模型对于行为策略的表达性越来越好; 2) 当扩散时间步$T=5$时, 行为策略数据能够基本被还原, 此时计算机耗时为 2.41 s; 3) 当扩散时间步T取值为 25 时, 由于模型可以更精细地调整和优化生成的样本, 行为策略数据被表达得更为准确, 但计算时间大为增加 (8.17 s). 综合考虑生成质量和计算效率两方面因素, 且鉴于扩散时间步为 5 时的生成行为策略数据带有小部分噪声, 这对于提升模型的泛化性和鲁棒性具有一定帮助, 为此在后续实验中将扩散时间步T设置为 5.

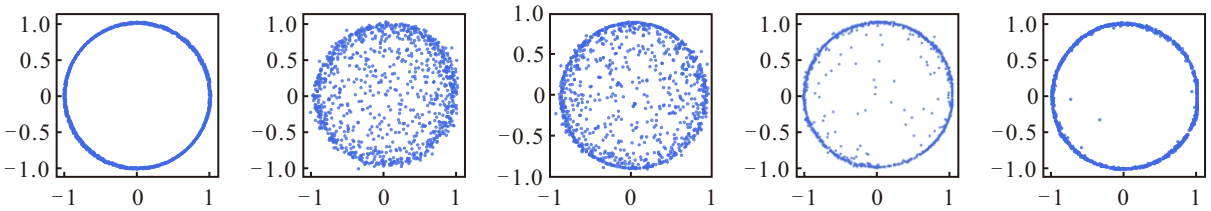


图1　扩散步数对 ACDP 性能的影响

此外, 对比了单峰高斯策略与扩散策略对于多模态数据的动作采样. 如果使用单峰高斯分布来表示策略, 则虽然噪声单位圆上的点均可以获得正奖励, 但由于单峰高斯分布的假设不准确, AWR 执行的动作会更逼近圆心, 如图 2(a) 所示. 也就是说, AWR 由于其单峰高斯的表达能力有限, 无法捕获离线数据集中的真实数据分布. 相比于 AWR, 由于 DiffusionQL 与 ACDP 均使用了扩散模型以表达行为策略, 能够更为准确地恢复行为策略的模态.

进一步, 通过绘制 bandit 任务的动作采样直方图来直观地展示 ACDP 能够指导智能体更好地进行策略提升. 在这个例子中, 用于演示的数据分布有两个突出的模态, 分别对应着低回报动作和高回报动作. 之所以将高回报动作的密度降低, 是因为在现实数据中, 高回报动作往往是珍贵且稀少的. 由图 3 可以看出: 1) 尽管 AWR 较好地拟合出了高斯分布, 但

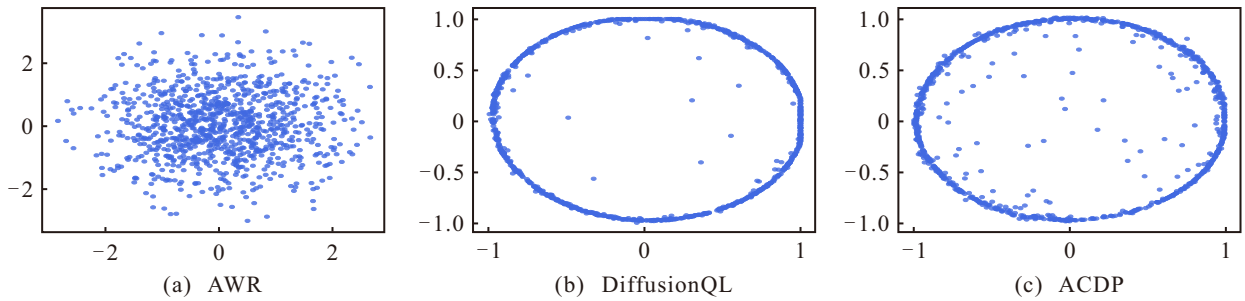


图2　不同算法的动作采样 (bandit 任务)

由于其表达能力有限, 仍然学习了很多分布外的动作; 2) DiffusionQL 可以有效缓解学习到分布外动作的问题, 准确地还原了多模态的数据分布, 但无法提升高回报动作的概率密度; 3) ACDP 通过使用优势约束扩散策略, 在还原数据分布的同时加大了高回报动作的密度, 从而使得习得策略能够更好地解决目标任务.

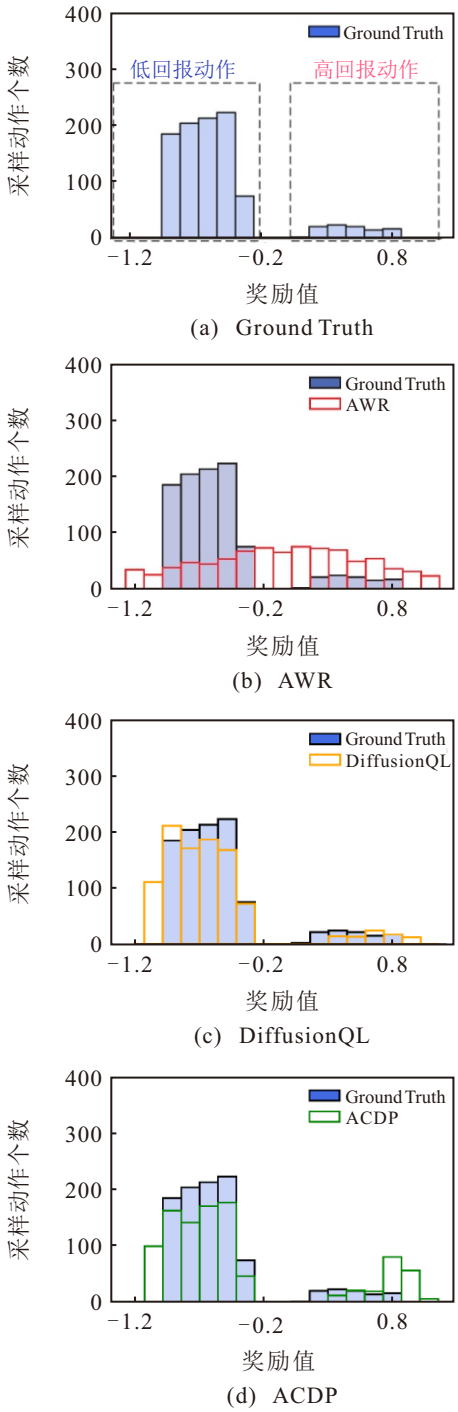


图3 动作采样直方图 (bandit 任务)

2.3 D4RL 基准测试结果与分析

本文使用 D4RL 基准测试中 Gym-Mujoco 连续控制任务和 Antmaze 稀疏奖励导航任务来评价方法性能. 对于连续控制任务, 使用 4 种不同质量的数据集进行实验: 中等 (medium)、中等专家 (medium-expert)、中等回放 (medium-replay) 和随机 (random), 分别简记为-m、-m-e、-m-r 和-r. 具体而言, 中等和随机数据集分别使用中等 SAC 策略和随机 SAC 策略收集了与环境 100 万次交互的经验. 中等回放数据集收集了从初始阶段到中等性能训练的 SAC 策略的交互经验. 中等专家数据集是来自中等数据集的 100 万个样本和来自专家数据集的 100 万个样本的混合物. 为方便跨任务比较, 此处采用归一化得分作为评估指标, 将每个环境的分数大致归一化到 0 至 100 之间. 归一化得分的计算方式为

$$\text{归一化得分} = 100 \times \frac{\text{平均回报} - \text{参考随机回报}}{\text{参考专家回报} - \text{参考随机回报}}. \tag{18}$$

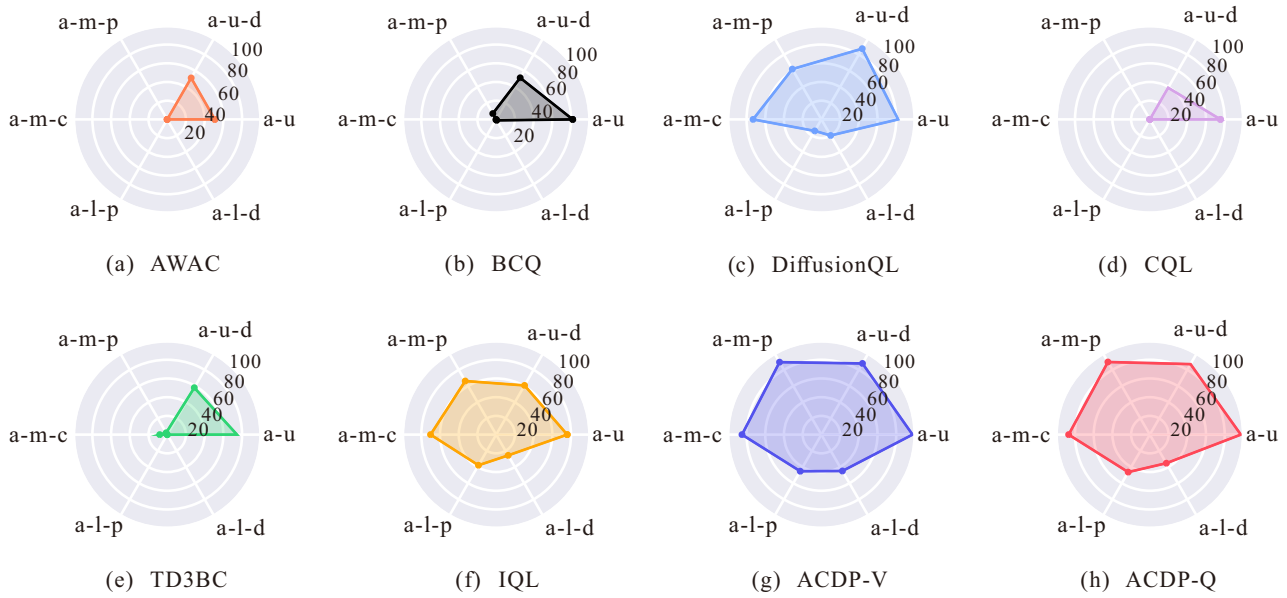
D4RL 基准测试任务的参考专家回报和参考随机回报如表 2 所示.

表2 D4RL 基准测试任务回报说明

任务	参考专家回报	参考随机回报
Halfcheetah	12 135.0	–280.1
Hopper	3 234.3	–20.2
Walker2d	4 592.3	1.63
Antmaze	1.0	0.0

对于稀疏奖励导航任务, 本文使用了 6 种不同质量的数据集进行实验. 小型迷宫 (umaze)、中型迷宫 (medium)、大型迷宫 (large) 分别表示不同复杂程度的迷宫: umaze 复杂度最低, large 复杂度最高. 与此同时, 根据四足机器人起点与终点不同, 将数据集分为演练和多样. 演练数据集包含一系列从不同指定起点到达一个特定终点的数据, 多样数据集包含从一个随机起点到达一个随机终点的数据. 与基于动作拼接的 Gym-Mujoco 连续控制任务不同的是, Antmaze 导航任务使用的奖励为 0-1 稀疏奖励. 当四足机器人到达终点时能够获得奖励 1, 其余奖励均为 0.

为方便表述, 分别将基于动作拼接和基于状态拼接的优势约束扩散策略的离线强化学习方法命名为 ACDP-Q、ACDP-V. 实验对比了 BC、AWAC[23]、BCQ[24]、DiffusionQL[19]、CQL[11]、TD3BC[8] 和 IQL[10]. 由于 Antmaze 环境的回报较为稀疏 (1 代表成功, 0 代表失败), 归一化得分呈现出较大的标准差. 为更直观地体现 ACDP 在稀疏奖励导航任务的优越性, 图 4 展示了所提方法与其余对比算法在 Antmaze 环境上的性能雷达图. 表 3 和图 5 分别展示了



(a) AWAC　(b) BCQ　(c) DiffusionQL　(d) CQL

(e) TD3BC　(f) IQL　(g) ACDP-V　(h) ACDP-Q

图4　Antmaze 稀疏奖励导航任务下的性能雷达图比较

表3　平均归一化得分对比

数据集	BC	AWAC	BCQ	DiffusionQL	CQL	TD3BC	IQL	ACDP-V	ACDP-Q
halfcheetah-m	42.40±0.19	49.62±0.20	46.57±0.49	49.40±0.17	46.82±0.15	48.28±0.15	48.39±0.13	50.26±0.46	**53.80±0.30**
hopper-m	53.51±1.76	69.87±4.10	55.75±1.83	75.25±4.19	61.83±1.64	58.64±1.68	62.19±4.53	81.64±4.75	**99.39±1.01**
walker2d-m	63.23±16.24	84.60±1.76	72.15±4.74	84.65±0.67	80.70±0.97	84.31±0.93	81.14±2.25	**85.93±0.56**	75.86±3.36
halfcheetah-m-e	55.95±7.35	95.06±0.96	92.86±1.50	95.13±0.40	91.62±2.58	92.06±2.47	93.53±1.09	95.71±0.35	**96.59±0.55**
hopper-m-e	52.30±4.01	**108.71±2.98**	106.41±2.96	100.75±3.32	97.30±5.93	97.27±5.81	91.63±27.23	105.84±2.09	101.02±3.46
walker2d-m-e	98.96±15.98	106.45±9.89	108.45±1.32	109.52±0.14	109.69±0.13	110.25±0.38	**111.42±0.41**	109.71±0.14	110.31±0.16
halfcheetah-m-r	35.66±2.33	45.78±0.24	41.17±0.55	46.84±0.21	45.13±0.31	44.54±0.25	43.71±0.70	47.33±0.15	**48.68±0.20**
hopper-m-r	29.81±2.07	97.25±2.57	40.33±10.46	99.12±1.48	87.14±6.67	67.59±12.41	80.98 ± 14.50	100.70±0.45	**100.78±0.48**
walker2d-m-r	21.80±10.15	81.41±2.58	54.37±5.38	90.46±2.24	81.11±2.30	81.08±4.23	77.46±7.84	92.34±2.58	**93.18±1.89**
halfcheetah-r	2.20±0.24	13.33±1.34	2.25±0	20.74±0.28	14.70±0.60	12.05±0.57	13.18±2.00	22.11±0.50	**22.96±0.24**
hopper-r	3.31±1.42	**12.75±6.20**	7.55±0.34	8.02±1.25	8.21±1.21	8.40±0.50	7.29±0.25	8.90±0.71	11.14±3.00
walker2d-r	0.98±0.10	5.30±2.78	4.39±1.31	1.84±1.39	3.67±1.30	1.64±1.24	3.08±0.70	**5.49±1.52**	3.25±1.19
locomotion total	460.11	770.13	632.25	781.72	727.92	706.11	714.00	**805.96**	**816.96**
antmaze-u	55.25±4.15	50.88±8.35	81.80±5.56	82.08±17.58	75.48±6.63	74.60±36.45	75.96±5.67	**97.44±2.26**	97.28±2.49
antmaze-u-d	47.25±4.09	51.20±11.06	51.36±6.72	87.32±4.28	39.12±21.66	57.92±31.85	60.72±10.27	**87.84±5.24**	87.00±4.12
antmaze-m-p	0	0	7.20±8.66	62.12±9.36	0	2.28±2.83	66.16±10.72	89.44±3.56	**89.64±4.01**
antmaze-m-d	0.75±0.83	0.08±0.56	0	73.04±12.75	0	8.04±9.26	70.16±7.00	84.88±6.12	**86.80±5.85**
antmaze-l-p	0	0	0.40±0.80	14.20±10.46	0	0.12±0.47	37.84±6.70	45.20±7.00	**46.28±8.94**
antmaze-l-d	0	0	1.16±1.60	19.84±10.27	0	0.08±0.39	25.64±8.40	**44.92±11.15**	35.28±13.13
antmaze total	103.25	102.16	141.92	338.60	114.6	143.04	336.48	**449.52**	**442.28**
total	563.36	872.29	774.17	1 120.32	842.52	849.15	1 050.48	**1 255.48**	**1 259.24**

ACDP-Q、ACDP-V 与对比方法在 D4RL 基准测试任务上的平均归一化得分和训练曲线. 实验过程中选取了 5 个随机种子, 并取后 10 步评估的均值计入表 3.

结合图 4、图 5 以及表 3 可以得出以下结论:

1) ACDP-Q 和 ACDP-V 在 Halfcheetah 环境中均展现出了很好的性能. 对于非专家数据集, ACDP-Q 和 ACDP-V 均领先于对比算法. 在收敛速度方面与对比算法持平, 但 ACDP-Q 和 ACDP-V 均获得了更高的归一化得分. 对于专家数据集, 由于专家数据的占比增大, ACDP-Q 和 ACDP-V 相较于对比算法没有体现出明显的竞争力, 但仍然获得了最高的归一化得分. 与此同时, 通过行为策略提取的 AWAC 和使用扩散模型表达行为策略的 DiffusionQL 获取了较高的归一化得分, 这是因为行为策略基本接近最优, 算法只需尽可能逼近行为策略就能达到较高的归一化得分. 尤其是 ACDP-Q, 在 hopper-m 数据集上获得了接近 100 的归一化得分, 这意味着

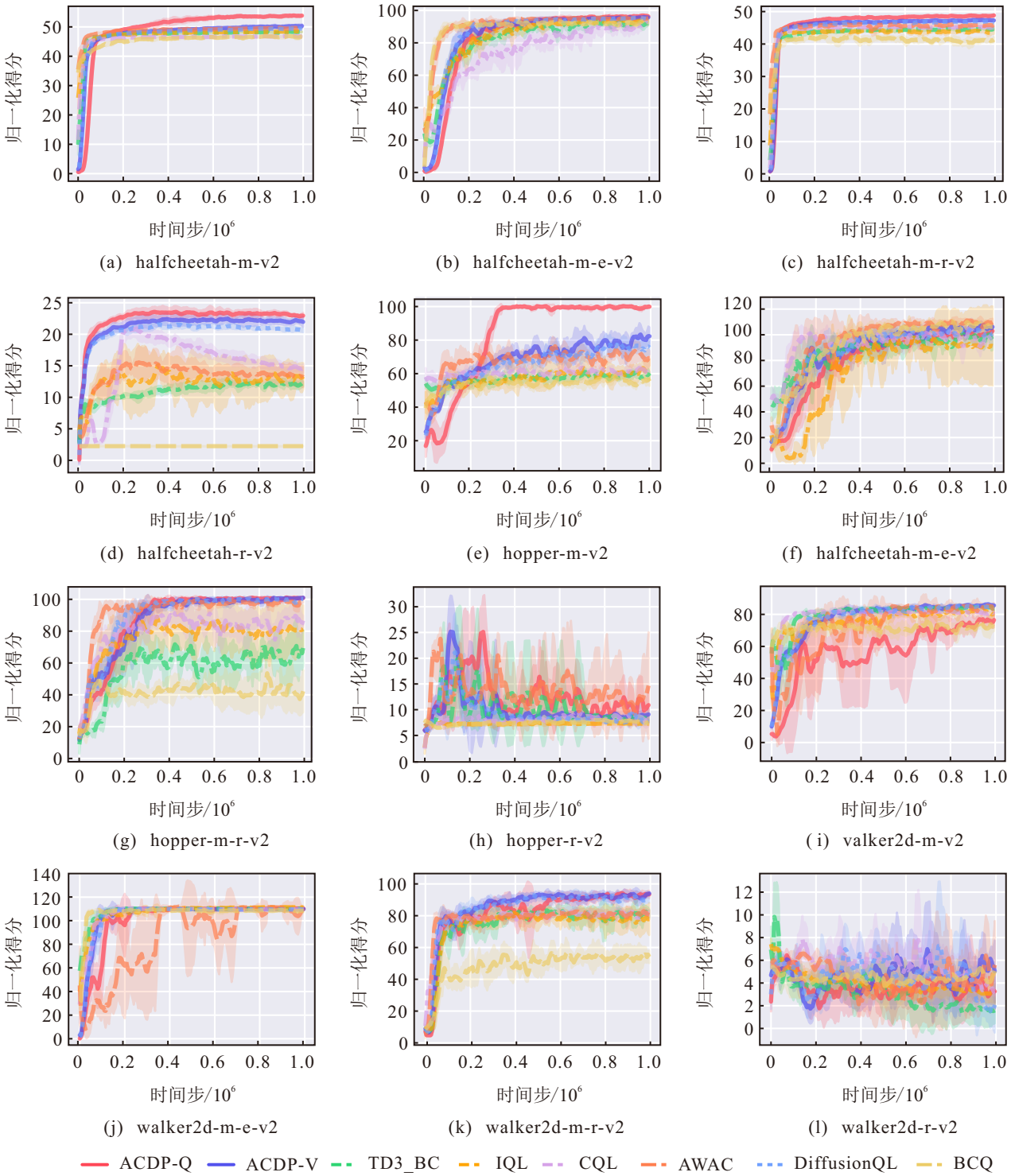


图5 不同控制任务上的归一化得分曲线比较

ACDP-Q 通过中等数据集学习的策略与专家策略基本持平. 对于 Walker2d 环境, ACDP-Q 在中等数据集上没有展现出最为先进的性能. 这是因为 walker2d-m 数据集在被制作时, 人为地给一些低质量动作添加高奖励, 因此对于考虑动作a的优势函数$A(s,a)$会被错误地估计, 因而导致约束部分失效, 从而得到相对次优的策略. 这点可以通过观察 BC 得到验证: 简单地进行行为克隆不仅会得到较低的归一化得分, 还会呈现较大的方差, 这说明 BC 的策略学习失效. 但 ACDP-V 通过使用状态优势弥补了这一缺陷, 获得了更高的归一化得分. 综上所述, ACDP-Q 和 ACDP-V 在非专家数据集中可以使策略更注重数量稀缺的高回报轨迹, 而在专家数据集中保持与对比算法相当的性能.

2) ACDP-Q 与 ACDP-V 在 antmaze-u、antmaze-u-d、antmaze-m-p 和 antmaze-m-d 数据集上已经达到较高水平. 在奖励更加稀疏的 antmaze-l-p 和 antmaze-l-d 数据集上也超越了当前稀疏奖励导航任务上最

为优越的 IQL. 无论是 ACDP-V 还是 ACDP-Q, 均实现了超越当前 SOTA 算法的性能.

3) 如表 3 所示, 在奖励连续的连续控制任务中, ACDP-Q 的性能优于 ACDP-V. 这是因为对于基于动作拼接的连续控制任务而言, 在设计优势函数时, ACDP-Q 相比于 ACDP-V 更注重于动作所产生的期望回报, 策略学习阶段更注重执行离线数据集中更容易得到高分的动作; 在奖励稀疏的导航任务中, 基于状态拼接的稀疏奖励导航任务的回报具有二值性. 因此, 通过动作引导使智能体逐渐向目标点靠近会更加困难. 如果智能体更希望达到目标点 (期望状态) 以获取最高的回报, 那么就更需要权衡当前状态和目标状态的差异, 这可以体现在状态值函数上.

通过实验对比发现: ACDP 在基于动作拼接的 Gym-Mujoco 连续控制任务和基于状态拼接的 Antmaze 稀疏奖励导航任务上都展现了较优的性能. 尤其是在 Antmaze 环境下的稀疏奖励导航任务上, ACDP 会使智能体更偏好那些稀缺的高回报轨迹, 这与本文的理论分析是一致的.

3 结 论

针对离线数据集中仅包含数量较少的高回报轨迹问题, 本文提出了一种优势约束扩散策略的离线强化学习方法. 通过对扩散策略进行优势加权, 使智能体能够在训练阶段更专注于那些具有高回报的轨迹. 针对连续控制任务和稀疏奖励导航任务, 分别设计了基于动作拼接和基于状态拼接的优势函数, 并使用指示函数直接对扩散策略进行加权. 最后, 通过 bandit 实验可视化了 ACDP 的效果, 并且 D4RL 基准测试上的结果表明, ACDP-Q 与 ACDP-V 对于大部分连续控制任务均获得了最高的归一化得分, 在稀疏奖励导航任务中达到最先进的性能.

参考文献 (References)

[1] 吴启宇, 谢非, 黄磊, 等. 基于深度/单目融合视觉及强化学习的机器人定位棋局与行棋策略[J]. 控制与决策, 2022, 37(12): 3278-3288.
(Wu Q Y, Xie F, Huang L, et al. Chess positioning and playing strategy of robot based on integrated depth/mono vision and reinforcement learning[J]. Control and Decision, 2022, 37(12): 3278-3288.)

[2] 闫超, 相晓嘉, 徐昕, 等. 多智能体深度强化学习及其可扩展性与可迁移性研究综述[J]. 控制与决策, 2022, 37(12): 3083-3102.
(Yan C, Xiang X J, Xu X, et al. A survey on scalability and transferability of multi-agent deep reinforcement learning[J]. Control and Decision, 2022, 37(12): 3083-3102.)

[3] Wang X S, Zhang J Z, Hou D Y, et al. Autonomous driving based on approximate safe action[J]. IEEE Transactions on Intelligent Transportation Systems, 2023, 24(12): 14320-14328.

[4] 户高铭, 蔡克卫, 王芳, 等. 基于深度强化学习的无地图移动机器人导航[J]. 控制与决策, 2024, 39(3): 985-993.
(Hu G M, Cai K W, Wang F, et al. Mapless navigation based on deep reinforcement learning for mobile robots[J]. Control and Decision, 2024, 39(3): 985-993.)

[5] 孙辉辉, 胡春鹤, 张军国. 基于主动风险防御机制的多机器人强化学习协同对抗策略[J]. 控制与决策, 2023, 38(5): 1420-1429.
(Sun H H, Hu C H, Zhang J G. Cooperative countermeasure strategy based on active risk defense multi-agent reinforcement learning[J]. Control and Decision, 2023, 38(5): 1420-1429.)

[6] 董豪, 杨静, 李少波, 等. 基于深度强化学习的机器人运动控制研究进展[J]. 控制与决策, 2022, 37(2): 278-292.
(Dong H, Yang J, Li S B, et al. Research progress of robot motion control based on deep reinforcement learning[J]. Control and Decision, 2022, 37(2): 278-292.)

[7] 顾扬, 程玉虎, 王雪松. 基于优先采样模型的离线强化学习[J]. 自动化学报, 2024, 50(1): 143-153.
(Gu Y, Cheng Y H, Wang X S. Offline reinforcement learning based on prioritized sampling model[J]. Acta Automatica Sinica, 2024, 50(1): 143-153.)

[8] Fujimoto S, Gu S S. A minimalist approach to offline reinforcement learning[J/OL]. 2021, arXiv: 2106.06860.

[9] 程玉虎, 黄龙阳, 侯棣元, 等. 广义行为正则化离线 Actor-Critic[J]. 计算机学报, 2023, 46(4): 843-855.
(Cheng Y H, Huang L Y, Hou D Y, et al. Generalized offline actor-critic with behavior regularization[J]. Chinese Journal of Computers, 2023, 46(4): 843-855.)

[10] Kostrikov I, Nair A, Levine S. Offline reinforcement learning with implicit Q-learning[J/OL]. 2021, arXiv: 2110.06169.

[11] Kumar A, Zhou A, Tucker G, et al. Conservative Q-learning for offline reinforcement learning[J/OL]. 2020, arXiv: 2006.04779.

[12] Huang L Y, Dong B T, Zhang W D. Efficient offline reinforcement learning with relaxed conservatism[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2024, 46(8): 5260-5272.

[13] Chen L L, Lu K, Rajeswaran A, et al. Decision transformer: Reinforcement learning via sequence modeling[J/OL]. 2021, arXiv: 2106.01345.

[14] Janner M, Du Y L, Tenenbaum J B, et al. Planning with diffusion for flexible behavior synthesis[J/OL]. 2022, arXiv: 2205.09991.

[15] Ajay A, Du Y L, Gupta A, et al. Is conditional generative modeling all you need for decision-making?[J/OL]. 2022, arXiv: 2211.15657.

[16] Kidambi R, Rajeswaran A, Netrapalli P, et al. MOReL: Model-based offline reinforcement learning[J/OL]. 2020, arXiv: 2005.05951.

[17] Yu T H, Kumar A, Rafailov R, et al. Combo: Conservative offline model-based policy optimization[C]. Proceedings of Advances in Neural Information Processing Systems. Virtual, 2021: 28954-28967.

[18] Yu T H, Thomas G, Yu L T, et al. MOPO: Model-based offline policy optimization[J/OL]. 2020, arXiv: 2005.13239.

[19] Wang Z D, Hunt J J, Zhou M Y. Diffusion policies as an expressive policy class for offline reinforcement learning[J/OL]. 2023, arXiv: 2208.06193.

[20] Zhang W H, Kumar A, Karnik S, et al. Beyond uniform sampling: Offline reinforcement learning with imbalanced datasets[C]. Proceedings of 37th Conference on Neural Information Processing Systems. New Orleans, 2023: 4985-5009.

[21] Ho J, Jain A, Abbeel P. Denoising diffusion probabilistic models[C]. Proceedings of Advances in Neural Information Processing Systems. Virtual, 2020: 6840-6851.

[22] Peng X B, Kumar A, Zhang G, et al. Advantage-weighted regression: Simple and scalable off-policy reinforcement learning[J/OL]. 2019, arXiv: 1910.00177.

[23] Nair A, Gupta A, Dalal M, et al. AWAC: Accelerating online reinforcement learning with offline datasets[J/OL]. 2020, arXiv: 2006.09359.

[24] Fujimoto S, Meger D, Precup D. Off-policy deep reinforcement learning without exploration[J/OL]. 2018, arXiv: 1812.02900.

作者简介

王雪松 (1974−), 女, 教授, 博士生导师, 主要研究方向为机器学习、人工智能, E-mail: wangxuesongcumt@163.com;

张恒瑞 (2001−), 男, 硕士生, 主要研究方向为强化学习, E-mail: hengruizhang@cumt.edu.cn;

张佳志 (1998−), 男, 博士生, 主要研究方向为强化学习, E-mail: zjzcumt@163.com;

程玉虎 (1973−), 男, 教授, 博士生导师, 主要研究方向为机器学习、智能系统, E-mail: chengyuhu@163.com.