Active Advantage-Aligned Online Reinforcement Learning with Offline Data

Xuefeng Liu[1]*, **Hung T. C. Le**[1], **Siyu Chen**[2]
Rick Stevens[1], **Zhuoran Yang**[2], **Matthew R. Walter**[3], **Yuxin Chen**[1]
[1]Department of Computer Science, University of Chicago
[2]Department of Statistics and Data Science, Yale University
[3]Toyota Technological Institute at Chicago

Abstract

Online reinforcement learning (RL) enhances policies through direct interactions with the environment, but faces challenges related to sample efficiency. In contrast, offline RL leverages extensive pre-collected data to learn policies, but often produces suboptimal results due to limited data coverage. Recent efforts integrate offline and online RL in order to harness the advantages of both approaches. However, effectively combining online and offline RL remains challenging due to issues that include catastrophic forgetting, lack of robustness to data quality and limited sample efficiency in data utilization. In an effort to address these challenges, we introduce A^3RL, which incorporates a novel confidence-aware Active Advantage-Aligned (A^3) sampling strategy that dynamically prioritizes data aligned with the policy's evolving needs from both online and offline sources, optimizing policy improvement. Moreover, we provide theoretical insights into the effectiveness of our active sampling strategy and conduct diverse empirical experiments and ablation studies, demonstrating that our method outperforms competing online RL techniques that leverage offline data. Our code will be publicly available at: https://github.com/xuefeng-cs/A3RL.

1 Introduction

Reinforcement learning (RL) has achieved notable success in many domains, such as robotics [24, 25], game play [40, 57], drug discovery [34, 37], and reasoning with Large Language Models (LLMs) [16]. Online RL algorithms such as Q-learning [67], SARSA [53], and PPO [56] learn and make decisions in an online, sequential manner, whereby an agent interacts with an environment and learns from its experience. However, due to the need for exploration that is fundamental to RL, online RL tends to be highly sample inefficient in high-dimensional or sparse reward environments. A complementary approach to improve the sample efficiency is imitation learning (IL) [51, 52], where an agent learns a policy by leveraging expert demonstrations [7, 35, 36].

However, in many cases, we do not have access to a live expert to query, but often have access to an abundance of logged data collected from experts. One approach to make use of this data is through offline reinforcement learning. Offline RL [30, 47] learns a policy solely from such a fixed dataset of pre-collected experiences, without the need to directly interact with the environment. Despite its advantages, offline RL often results in a suboptimal policy due to dataset limitations. This has motivated recent work that combines offline and online RL, whereby learning begins from a logged dataset before transitioning to online interactions for further improvement. While beneficial, contemporary offline-to-online RL methods suffer from catastrophic forgetting, where previously

[*] Correspondence to: Xuefeng Liu <xuefeng@uchicago.edu>.

learned knowledge is overwritten during online fine-tuning, leading to significant performance degradation [39, 69].

More recently, methods that integrate online RL with offline datasets utilize off-policy techniques to incorporate offline data while learning online [2, 59], mitigating catastrophic performance drops. These techniques do not require any preliminary offline RL training or incorporate specific imitation clauses that prioritize pre-existing offline data. Notably, RLPD [2] exhibits strong empirical performance, however it employs a uniform random sampling strategy for both offline and online learning, ignoring that different transitions contribute differently to the various stage of policy improvement. Furthermore, this uniform sampling strategy may result in data inefficiencies (e.g., sampling useless data while missing valuable data) and also make policy improvement highly sensitive to data quality.

Our contributions. In this work, we introduce Active Advantage-Aligned Reinforcement Learning (A^3RL), a novel method that operates in the realm of online RL with an offline dataset. Our approach dynamically prioritizes data (transitions) that have the highest potential to maximize policy improvement, aligning with the evolving quality and learning needs of the policy. More specifically, A^3RL considers not only the relevance of the data in facilitating the current policy's online exploration and exploitation but also its estimated contribution to policy improvement via confidence-aware advantage-based prioritization. A^3RL demonstrates robustness to data quality in a black-box manner and maintains resilience under varying environmental conditions. Notably, it also effectively accelerates policy improvement, even in a purely online environment.

In summary, our contributions are:

- We propose A^3RL, a novel algorithm for online RL with offline data. This algorithm surpasses current state-of-the-art (SOTA) methods by integrating a priority-based active sampling strategy based on the value of confidence-aware advantage function and coverage by offline dataset.
- In contrast to RLPD and other related works [29, 55], which lack theoretical support, this study provides theoretical insights of our confidence-aware active advantage-aligned sampling strategy, demonstrating superiority and its minimum improvement gap over random sampling.
- Through extensive empirical evaluations in various environments, we demonstrate that A^3RL achieves consistent and significant improvements over prior SOTA models.
- Given the black-box nature of offline datasets, we conduct comprehensive ablation studies across a range of dataset qualities and environmental settings, including purely online scenarios, to evaluate the robustness of A^3RL. These studies consistently confirm its stable performance across diverse conditions, regardless of environmental factors or data quality.

2 Related Work

Online RL with offline datasets Several methods exist that incorporate offline datasets in online RL to enhance sample efficiency. Many rely on high-quality expert demonstrations [19, 23, 42, 49, 64, 70]. Nair et al. [43] introduced the Advantage Weighted Actor Critic (AWAC), which utilizes regulated policy updates to maintain the policy's proximity to the observed data during both offline and online phases. On the other hand, Lee et al. [29] propose an initially pessimistic approach to avoid over-optimism and bootstrap errors in the early online phase, gradually reducing the level of pessimism as more online data becomes available. Most relevant to our work is RLPD [2], which adopts a sample-efficient off-policy approach to learning that does not require pre-training. Unlike RLPD, which utilizes symmetric sampling to randomly draw from both online and offline datasets for policy improvement, A^3RL adopts a Prioritized Experience Replay (PER)-style method, whereby it selectively uses data from both datasets to enhance policy performance.

Prioritized experience replay Experience replay [31] enhances data efficiency in online RL by reusing past experiences. Priority Experience Replay (PER) [55] introduces prioritization based on temporal difference (TD) error to ensure that impactful experiences are used more frequently, and has proven effective in a variety of settings [17, 21, 42, 45, 54, 61, 65, 68]. Alternative prioritization strategies have been explored, such as prioritizing transitions based on expected return [20] or adjusting sample importance based on recency [11]. Existing research predominantly focuses on either purely online or offline applications of PER. Our research distinctively integrates the advantages of both online and offline data in an innovative way. Eysenbach et al. [9] apply a density ratio to the reward instead of weighting the samples. The most relevant studies to ours include Sinha et al.

[58] that uses the density ratio between off-policy and near-on-policy state-action distributions as an importance weight for policy evaluation, and Lee et al. [29] that employs density ratios to select relevant samples from offline datasets. Our method differs by not only using the density ratio to assess the "on-policyness" of the data but also by considering the confidence-aware advantage value to determine how much the data can contribute to enhancing policy improvement.

Active learning in RL Active learning has been explored in RL for data-efficient exploration [8, 10, 27, 33, 35, 36, 38]. Unlike previous approaches that focus on oracle selection [35, 36], state exploration [8, 36] or reward estimation [38], A^3RL introduces an active transition sampling mechanism tailored to online RL with offline data, prioritizing transitions that maximize policy improvement. We defer more details of related work to App. B.

3 Preliminaries and Problem Statement

We consider a discounted Markov decision process (MDP) environment [3] characterized by a tuple $M = (\mathcal{S}, \mathcal{A}, \mathcal{P}, R, \gamma, d_0)$, where $\mathcal{S}$ represents a potentially infinite state space, $\mathcal{A}$ is the action space, $\mathcal{P} : \mathcal{S} \times \mathcal{A} \rightarrow \Delta(\mathcal{S})$ is the unknown transition kernel, $R : \mathcal{S} \times \mathcal{A} \rightarrow [0, 1]$ is the reward function, $\gamma \in (0, 1)$ is the discount factor and $d_0\,(s)$ is the initial state distribution. The learner's objective is to solve for the policy $\pi : \mathcal{S} \rightarrow \Delta(\mathcal{A})$ that maximizes the expected sum of discounted future rewards $\mathbb{E}_\pi[\sum_{t=1}^{\infty} \gamma^t r\,(s_t, a_t)]$, where the expectation is taken over the trajectory sampled from π.

Maximum entropy RL. In this work, we adopt off-policy soft actor-critic (SAC) [15] RL to train an agent with samples generated by any behavior policy. We use a general maximum entropy objective [2, 15, 71] as follows:

$$\max_{\pi} \mathbb{E}_{s\sim\rho^\pi, a\sim\pi}\left[\sum_{t=0}^{\infty} \gamma^t \left(r_t + \alpha\mathcal{H}\left(\pi\left(a|s\right)\right)\right)\right], \tag{1}$$

where α is a temperature parameter. This involves optimizing reward while encouraging exploration, making the learned policy more robust.

Q-value and advantage function. The Q-value function measures the expected return of executing action a in state s under policy π: $Q^\pi\,(s, a) = \mathcal{B}^\pi Q^\pi\,(s, a)$, where $\mathcal{B}^\pi$ is the Bellman operator: $\mathcal{B}^\pi Q\,(s, a) := r\,(s, a) + \gamma\mathbb{E}_{s'\sim P(\cdot|s,a)}[V^\pi\,(s')]$. The soft state value function is defined as: $V^\pi\,(s) := \mathbb{E}_{a\sim\pi(\cdot|s)}[Q^\pi\,(s, a) - \log\pi\,(a|s)]$. For a generator policy π, the advantage function [60] quantifies the relative benefit of selecting a over the policy's default behavior:

$$A^\pi\,(s, a) = Q^\pi\,(s, a) - V^\pi\,(s)\,. \tag{2}$$

Specifically, SAC learns a soft Q-Function, denoted as $Q_\theta\,(s, a)$ which parameterized by θ, and a stochastic policy π_ϕ parameterized by ϕ. The SAC method involves alternating between updates for the critic and the actor by minimizing their respective objectives [29] as follows

$$\begin{aligned}\mathcal{L}^{\text{SAC}}_{\text{critic}}\,(\theta) =&\mathbb{E}_{(s_t,a_t,s_{t+1})\sim\mathcal{R}}[(Q_\theta\,(s_t, a_t) - r\,(s_t, a_t)\\ &- \gamma\mathbb{E}_{a_{t+1}\sim\pi_\phi}\left[Q_{\overline{\theta}}\,(s_{t+1}, a_{t+1}) - \alpha\log\pi_\phi\,(a_{t+1}|s_{t+1})\right])^2],\\ \mathcal{L}^{\text{SAC}}_{\text{actor}}\,(\phi) =&\mathbb{E}_{s_t\sim\mathcal{R},a_t\sim\pi_\phi}[\alpha\log\pi_\phi\,(a_t|s_t) - Q_\theta\,(s_t, a_t)]\,,\end{aligned}$$

Here, $\mathcal{R}$ is an experience replay buffer of either on-policy experience [60] or through off-policy experience [41, 46], and $\overline{\theta}$ denotes the delayed parameters.

Prioritized experience replay. PER [55] serves as the basis of our sampling techniques, providing a framework for prioritizing experience replay based on transition importance. Instead of sampling uniformly from the replay buffer $\mathcal{R}$, PER assigns higher probability to more informative transitions, leading to improved sample efficiency [17]. Each transition $\mathcal{R}_i = (s_i, a_i, r_i, s_{i+1})$ is assigned a priority σ_i, typically based on the TD-error: $\delta = r + \gamma V\,(s_{t+1}) - V\,(s_t)$ [4, 17, 45, 55, 63]. Subsequently, the sampling approach of PER involves establishing an index set $\mathcal{I}$ within the range of $[|\mathcal{R}|]$ based on the probabilities p_i assigned by the priority set as follows: $p_i = \frac{\sigma_i^\zeta}{\sum_{k\in[|\mathcal{R}|]}\sigma_k^\zeta}$, with a hyper-parameter $\zeta > 0$. To correct for sampling bias, PER applies importance sampling weights:

$$u_i \propto \left(1/(|\mathcal{R}| \cdot p_i)\right)^\beta, \tag{3}$$

where β anneals from $\beta_0 \in (0, 1)$ to 1 during training to counteract bias in the learning updates, and the importance sampling weights are normalized to have maximum weight of 1 for stability. While standard PER prioritizes TD-error, our method extends this framework to prioritize transitions based on onlineness and contribution to policy improvement.

Online RL with offline datasets. In this work, we study online RL with offline datasets denoted as $\mathcal{D}$ [2]. These datasets consist of a set of tuples (s, a, r, s') generated from a specific MDP. A key characteristic of offline datasets is that they typically offer only partial coverage of state-action pairs. In other words, the set of states and actions in the dataset, denoted as $\{(s, a) \in \mathcal{D}\}$, represents a limited subset of the entire state space and action space, $\mathcal{S} \times \mathcal{A}$. Moreover, learning on the data with incomplete coverage of state-action pairs potentially results in excessive value extrapolation during the learning process for methods using function approximation [14]. Our model, based on SAC [15], incorporates several effective strategies for RL with offline data, as outlined in RLPD [2]. These strategies include:

Layer Normalization: Off-policy RL algorithms often query the learned Q–function with out-of-distribution actions, leading to overestimation errors due to function approximation. This can cause training instabilities and even divergence, particularly when the critic struggles to keep up with growing value estimates. To address this, prior research has employed Layer Normalization to ensure that the acquired functions do not extrapolate in an unconstrained manner. Layer Normalization acts to confine Q-values within the boundaries set by the norm of the weight layer, even for actions beyond the dataset. As a result, the impact of inaccurately extrapolated actions is substantially reduced, as their associated Q-values are unlikely to significantly exceed those already observed in the existing data. Consequently, Layer Normalization serves to alleviate issues such as critic divergence and the occurrence of catastrophic overestimation.

Update-to-Data: Enhancing sample efficiency in Bellman backups can be accomplished by elevating the frequency of updates conducted per environment step. This approach, often referred to as the update-to-data (UTD) ratio, expedites the process of backing up offline data.

Clipped Double Q-Learning: The maximization objective of Q-learning and the estimation uncertainty from value-based function approximation often leads to value overestimation [62]. To address this problem, Fujimoto et al. [13] introduced Clipped Double Q-Learning (CDQ) as a means of mitigation. CDQ involves taking the minimum from an ensemble of two Q-functions for computing TD-backups. The targets for updating the critics are given by the equation $y = r(s, a) + \gamma \min_{i=1,2} Q_{\theta_i}(s', a')$, where $a' \sim \pi(\cdot|s')$.

Maximum Entropy RL: Incorporating entropy into the learning objective (as defined in (1)) helps mitigate overconfidence in value estimates, particularly when training with offline datasets. In offline RL, policies may become overly conservative due to limited dataset coverage, leading to suboptimal exploration during fine-tuning. By preserving policy stochasticity, entropy regularization ensures that the agent remains adaptable when transitioning from offline training to online interactions. This controlled exploration has been shown to improve training stability and prevent premature convergence [2, 6, 15, 18].

Limitations of the prior state-of-the-art. A drawback of RLPD, as discussed by Ball et al. [2], lies in its symmetric random sampling method applied to both online and offline data, disregarding the significance of individual transitions for evolving quality of policy. This predefined approach to sampling can potentially lead to less than optimal policy improvements due to the omission of vital data and inefficiencies arising from the use of redundant data. Such inefficiencies fail to offer any positive contribution towards enhancing policy. To address the limitation, our research presents an innovative active data sampling technique, specifically designed to optimize the use of both online and offline data in the process of policy improvement.

4 Algorithm

4.1 Confidence-aware Active Advantage-Aligned Sampling Strategy

In this study, we theoretically derived from the performance difference lemma in §5 and presented active advantage-aligned strategy, a novel sampling approach for policy improvement. Here, 'advantage' measures the potential impact of the transition on policy improvement, while 'aligned' assesses how well the transition aligns with the states sampled online by the current policy. This method allows for the safe utilization of online and offline samples by harnessing relevant, near on-policy offline samples that also present the potential to enhance policy improvement. For the advantage term, to enhance robustness, we use the pessimistic CDQ Q estimation, while incorporating uncertainty estimation for the value function under the current policy. Specifically, we estimate both the value function $\widehat{V}$—which directly determines the estimated advantage of $\widehat{A}$—and the associated uncertainty $\widehat{C}_A\left(s,a\right)$, through Monte Carlo samples of the on-policy actions. Furthermore, we extend this approach to density ratio estimation, using an ensemble of density networks to predict the density ratio $\widehat{w}\left(s,a\right)$ and associated uncertainty $\widehat{C}_w\left(s,a\right)$. This approach broadens the distribution of samples used for updates, centering around on-policy examples, thereby facilitating immediate value. The active advantage-aligned priority σ and the probability p are as follows:

$$p_i = \frac{\sigma_i^{\zeta}}{\sum_{k\in[|\mathcal{R}|]}\sigma_k^{\zeta}},\ \sigma_i = \sigma\left(s_i,a_i\right) = \left(\mathbb{I}^{\text{off}}\underline{w}\left(s_i,a_i\right) + \mathbb{I}^{\text{on}}\right)\cdot\exp\left(\xi\cdot\underline{A}\left(s_i,a_i\right)\right), \tag{4}$$

$$\underline{w}\left(s_i,a_i\right) = \widehat{w}\left(s_i,a_i\right) - \widehat{C}_w(s_i,a_i),\ \underline{A}\left(s_i,a_i\right) = \widehat{A}\left(s_i,a_i\right) - \widehat{C}_A(s_i,a_i), \tag{5}$$

where $\mathbb{I}^{\text{off}}$ and $\mathbb{I}^{\text{on}}$ represents the indicator of offline and online respectively, density ratio $\underline{w}\left(s,a\right)$ is the LCB (Lower Confidence Bound) [35] of density ratio, which measures the onlineness of the transition (defined in Eq. (6)) in a conservative manner, $\underline{A}\left(s,a\right)$ is LCB of the advantage term, which assesses the potential of the transition in improving the policy and $\xi > 0$ representing a temperature hyperparameter associated with the advantage term, and another $\zeta > 0$ for the entire priority term, per the standard PER approach. This approach considers not only the on-policyness of the data but also measures how important the data contributes to the current policy improvement. The active advantage-aligned sampling strategy aims to assign greater weight to transitions that are either not well covered by the offline dataset—indicating that the state-action pair is novel to the offline policy (i.e., the density ratio is large)—or that represent good actions for maximizing cumulative reward (i.e., the advantage / Q function is large).

Density ratio. We evaluate the onlineness through the use of a density ratio

$$w\left(s,a\right) := d^{\text{on}}\left(s,a\right)/d^{\text{off}}\left(s,a\right) \tag{6}$$

for a given transition, where $d^{\text{on}}\left(s,a\right)$ denotes the state-action distribution of online samples in the online buffer $\mathcal{R}^{\text{on}}$ and the $d^{\text{off}}\left(s,a\right)$ denotes the offline samples in the offline buffer $\mathcal{R}^{\text{off}}$. By identifying a transition with a high density ratio $w\left(s,a\right)$, we can effectively select a near-on-policy sample $\left(s,a,s'\right)$ from the offline dataset $\mathcal{B}^{\text{off}}$. Consider the much larger volume of offline data compared to online data, this would greatly improve the amount of transition and diversity of coverage for policy improvement in each step.

Estimating the likelihoods $d^{\text{off}}\left(s,a\right)$ and $d^{\text{on}}\left(s,a\right)$ poses a challenge, as they could represent stationary distributions from mixture of complex policy. To address this issue, we employ a method studied by Lee et al. [29], Sinha et al. [58] for density ratio estimation that does not rely on likelihoods. This method approximates $w\left(s,a\right)$ by training a neural network $w_{\psi_i}\left(s,a\right)$, which is parameterized by $\psi_i, i\in[N_e]$, where N_e is the number of density networks in the ensemble. The training exclusively uses samples from $\mathcal{B}^{\text{off}}$ and $\mathcal{B}^{\text{on}}$. We use variational representation of f-divergences [44]. Consider P and Q as probability measures on a measurable space $\mathcal{X}$, with P being absolutely continuous $w.r.t\ Q$. We define the function $f\left(y\right) := y\log\frac{2y}{y+1} + \log\frac{2}{y+1}$. The Jensen-Shannon (JS) divergence is then defined as $D_{JS}\left(P||Q\right) = \int_{\mathcal{X}} f\left(dP\left(x\right)/dQ\left(x\right)\right)dQ\left(x\right)$. Then we use a parametric based model $w_\psi\left(x\right)$ to represent density ratio $\frac{dP}{dQ}$ and estimated the density ratio by maximizing the lower bound of $D_{JS}\left(P||Q\right)$:

$$\mathcal{L}^{\text{DR}}\left(\psi\right) = \mathbb{E}_{x\sim P}[f'\left(w_\psi\left(x\right)\right)] - \mathbb{E}_{x\sim Q}[f^*\left(f'\left(w_\psi\left(x\right)\right)\right)],$$

where $w_\psi\left(x\right) \geq 0$ is represented by a neural network, with parameters ensuring that the outputs remain non-negative through the use of activation function. Additionally, f^* represents convex conjugate and we sampled from $\mathcal{B}^{\text{on}}$ for $x\sim P$ and from $\mathcal{B}^{\text{off}}$ for $x\sim Q$.

Confidence-aware active advantage-aligned sampling. Relying solely on the density ratio is insufficient; even if a transition appears to be relevant in the online context, it may still fail to contribute meaningfully to policy improvement. For instance, consider a transition (s,a,s'). If the

Algorithm 1 A^3RL

```
Select LayerNorm, large ensemble Size $E$, gradient steps $G$, discount $\gamma$, temperature $\alpha$.
Randomly initialize Critic $\theta_i$ (set targets $\theta'_i = \theta_i$) for $i = 1, 2, \dots, E$, Actor $\phi$ parameters.
Select critic EMA weight $\rho$, batch size $N$, determine number of Critic targets to subset $Z \in \{1, 2\}$
Initialize buffer $\mathcal{D}$ with offline data, online replay buffer $\mathcal{R} \leftarrow \emptyset$
while True do
    Receive initial observation state $s_0$
    for $t = 0, \dots, T$ do
        Take action $a_t \sim \pi_\phi(\cdot|s_t)$, update buffer $\mathcal{R} \leftarrow \mathcal{R} \cup \{(s_t, a_t, r_t, s_{t+1})\}$.
        Randomly sample a subset of size $\frac{N}{2}$ from online buffer $\mathcal{R}$ and size $\frac{N}{2}$ from offline buffer $\mathcal{D}$ to form a learning dataset $\mathcal{R}_N$
        Update density ensemble using $\mathcal{R}_N$
        Calculate priority $P_\mathcal{R}$ of $\mathcal{R}_N$ via (4)
        for $g = 1, \dots, G$ do
            Sample batch $b_N$ of size $N$ according to $P_\mathcal{R}$ from $\mathcal{R}_N$
            Sample set $\mathcal{Z}$ of $Z$ indices from $\{1, \dots, E\}$
            With $b$, set $y = r + \gamma\Big(\min_{i\in\mathcal{Z}} Q_{\theta'_i}(s', a') + \alpha \log \pi_\phi(a'|s')\Big), a' \sim \pi_\phi(\cdot|s')$
            for $i = 1, \dots, E$ do
                Calculate importance weight $u_i$ via (3).
                Update $\theta_i$ minimizing loss: $\ell = \sum_i u_i \cdot (y - Q_{\theta_i}(s, a))^2$
            Update target networks: $\theta'_i \leftarrow \rho\theta'_i + (1 - \rho)\theta_i$
        With b, update $\phi$ maximizing objective:
            $\frac{1}{E}\sum_{i=1}^{E} Q_{\theta_i}(s, a) - \alpha \log \pi_\phi(a|s)$, where $a \sim \pi_\phi(\cdot|s)$, $(s, a) \sim b_N$.
```

policy has previously encountered this state and taken the same action, or if the action performed in this state could potentially lead to a negative reward, such a transition would not that helpful in contributing to policy improvement, regardless of how closely it aligns with on-policy data.

To address this, we incorporate an estimate of the advantage value $A(s, a)$ (Eq. (2)) into our sampling strategy. Specifically, we integrate a non-negative exponential advantage term, $\exp(\xi \cdot A(s, a))$, into the priority calculation. This term ensures that transitions are selected not only based on relevance but also on their contribution to policy improvement. The higher the advantage value, the greater the transition's impact on learning, making our sampling mechanism both adaptive and optimization-aware.

For transitions from the offline dataset, we prioritize samples based on both the estimated density ratio and advantage value, retrieving near-on-policy samples that also provide policy improvement benefits. Since the data source is known, we set the density ratio to 1 for transitions from the online dataset and prioritize them purely based on advantage values under the current policy. Additionally, there may be uncertainty and significant variance in estimating the advantage value and density ratio. To address this, we adopt LCB as a conservative estimate. Thus, we define the priority function for sampling as:

$$\mathbb{I}^{\text{off}} \underline{w}(s_i, a_i) \cdot \exp(\xi \cdot \underline{A}(s_i, a_i)) + \mathbb{I}^{\text{on}} \exp(\xi \cdot \underline{A}(s_i, a_i)).$$

Note that this advantage-aligned sampling strategy is not a heuristic-based approach but is theoretically derived in the performance difference lemma [22], providing insights into its effectiveness and superiority over the random sampling approach (see Section 5).

The active sampling process in our algorithm is highlighted in blue in Algorithm 1, while our approach to addressing sampling bias is highlighted in red.

5 Theoretical Analysis

In this section, we derive the priority term theoretically from the performance difference lemma [22] and show that our active advantage-aligned sampling strategy leads to improved policy performance. Furthermore, we establish a theoretical lower bound on the performance improvement gap under our sampling scheme.

Theorem 1 *Suppose the Q-function class is uniformly bounded, and for any Q-function, the corresponding optimal policy lies within the policy function class. Let ϵ^t denote the ℓ_2 error of the Q-function in the critic update step. Let π^t be the policy at iteration t in $A^3\mathrm{RL}$, updated using priority-weighted sampling with $w(s,a)\exp(\xi \cdot A(s,a))$. Then, the following lower bound holds:*

$$J_\alpha^{\pi^{t+1}} - J_\alpha^{\pi^t} \geq J_\alpha^{\pi^\star} - J_\alpha^{\pi^t} - C\sqrt{\epsilon^t}\sup_{s,a}\left|R^t(s,a;\xi)\right|,$$

where $J_\alpha^\pi = \mathbb{E}_{s\sim\rho^\pi, a\sim\pi}\left[\sum_{t=0}^\infty \gamma^t\left(r_t + \alpha\mathcal{H}(\pi(a|s))\right)\right]$ is the entropy-regularized objective, $J_\alpha^{\pi^\star} - J_\alpha^{\pi^t}$ represents the maximum possible improvement if the true Q-function were known, and the function $R^t(s,a;\xi)$ is given by:

$$R^t(s,a;\xi) = \left(\frac{\pi^{t+1}(a \mid s)}{d^{on}(a \mid s)}\right)^{1-\xi} \cdot \frac{\sum_{s',a'} d^{on}(a',s')\pi^{t+1}(a' \mid s')^\xi}{d^{on}(a \mid s)^\xi} \cdot \frac{d^{\pi^{t+1}}(s)}{d^{on}(s)}.$$

The proof is provided in Appendix A. We note that the coefficient $R^t(s,a;\xi)$ is not necessarily the tightest possible bound, since it is based on the supremum norm and therefore can be dominated by a single (s,a) pair. A sharper result could be obtained by measuring distribution shift in the ℓ_2 norm (or some other weaker norm). We nevertheless adopt the simpler supremum-norm bound here for clarity and to highlight the core intuition behind why advantage reweighting yields improvement, as will be detailed in the following.

Comparison to random sampling. The fundamental concept behind proving that our sampling technique surpasses random sampling and contributes to positive policy improvement involves initially applying the performance difference lemma. This approach yields the performance differential term $J\left(\pi^{t+1}\right) - J\left(\pi^t\right)$ between the updated policy and the current policy. Our goal is to demonstrate that this term is non-negative under our sampling priority. To do this, we prove that by a shift of distribution, this term is no less than the gap

$$J^{\pi^\star} - J^{\pi^t} - C\sqrt{\epsilon^t}\sup_{s,a}|d^{\pi^{t+1}}(s,a)/\rho(s,a)|. \tag{7}$$

When looking at the distribution shift

$$\frac{d^{\pi^{t+1}}(s,a)}{\rho(s,a)} = \left(\frac{\pi^{t+1}(a \mid s)}{d^{\mathrm{on}}(a \mid s)}\right)^{1-\xi} \cdot \frac{\sum_{s',a'} d^{\mathrm{on}}(a',s')\pi^{t+1}(a' \mid s')^\xi}{d^{\mathrm{on}}(a \mid s)^\xi} \cdot \frac{d^{\pi^{t+1}}(s)}{d^{\mathrm{on}}(s)},$$

we notice the shift between online/offline dataset is canceled, and the remaining terms comprise a shift term $d^{\pi^{t+1}}(s)/d^{\mathrm{on}}(s)$ that characterizes how well the online data cover the visitation measure induced by the next policy, and another term that characterizes the shift in policy. In the sequel, we will see through an example why using some proper ξ helps reduce the shift in policy.

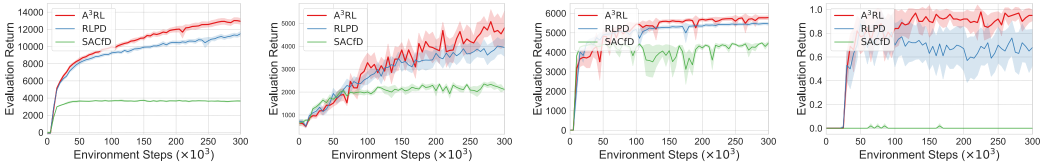
Why does advantage weighting help? We show that under certain conditions, the ratio $R^t(s,a)$ can decrease for increased value of ξ. Since ξ does not influcence the ration between the state distribution, let us just consider the bandit case with ratio

$$R^t(a;\xi) = \left(\frac{\pi^{t+1}(a)}{d^{\mathrm{on}}(a)}\right)^{1-\xi} \cdot \frac{\sum_{a'} d^{\mathrm{on}}(a')\pi^{t+1}(a')^\xi}{d^{\mathrm{on}}(a)^\xi}.$$

We illustrate the results of Theorem 1 on the bandit setting because its visitation measure reduces directly to the policy distribution—eliminating any dependence on a transition kernel—and note that the same argument carries over to MDPs with deterministic transitions. Moreover, the argument will still provide sufficient insight. Suppose the online data distribution $d^{\mathrm{on}}(a) \propto \exp(\beta_1 r(a))$ for some parameter β while the policy $\pi^{t+1}(a) \propto \exp(\beta_2 r(a))$ for some parameter $\beta_2 > \beta_1$. This is reasonable since the policy converges faster than the online buffer to the optimal policy. Then we have the following lemma.

Lemma 1 *For the bandit case with $d^{on}(a) \propto \exp(\beta_1 r(a))$ and $\pi^{t+1}(a) \propto \exp(\beta_2 r(a))$ for $\beta_2 > \beta_1$, the coefficient $\sup_a R^t(a;\xi)$ decreases as ξ increases within the range $\xi \in (0, 1-\beta_1/\beta_2)$.*

This lemma justifies that within a proper range of ξ, adding more advantage weighting would benefit learning by reducing the distributional shift.



(a) halfcheetah-random (b) ant-random (c) walker2d-medium (d) antmaze-medium

Figure 1: **Main results.** A comparison between A^3RL, the state-of-the-art baseline RLPD and SAC with offline data (SACfD) on various D4RL benchmark tasks. (a-c): dense reward, (d): sparse reward. Shaded areas represent one standard deviation based on ten seeds.

6 Experiments

Environments We evaluate A^3RL on both dense and sparse reward tasks from the D4RL benchmark [12]. These include halfcheetah, walker2d, and ant, which are dense reward locomotion tasks, and antmaze, which involves sparse rewards. Each environment offers offline datasets composed of trajectories ranging from completely random to expert. Appendix C provides additional details.

Setup. We employ the basic setup of the SAC networks as recommended by [2], i.e., with an ensemble of size 10 each for critic networks and target critic networks, as well as entropy regularization. A significant difference is that the MLP underlying these networks only has 2 layers of size 256 each, as we desired to see if the agent is able to learn with less complexity.

Baseline Methods For our main results, we compare A^3RL with two baselines: (1) RLPD [2], regarded as the SOTA baseline for addressing online RL with offline datasets, also attains state-of-the-art performance in this problem set, (2) SAC with offline data (SACfD), a canonical off-policy approach using offline data, as also studied in [43] and [64]. In the ablation studies, we evaluate A^3RL against five additional representative baselines: (3) Off2On [29], an offline-to-online RL method; (4) a variant of A^3RL using advantage estimation only; (5) an online version of A^3RL that excludes offline data; (6) SAC in an online setting without offline data; (7) TD (Temporal Difference) with a PER [55] sampling strategy; and (8) TD+Density, which combines PER with a density ratio sampling strategy.

6.1 Main results

Fig. 1 presents a comparative analysis of A^3RL's performance against the baseline SACfD and the current state-of-the-art method, RLPD. The results demonstrate that A^3RL consistently outperforms the baseline across the evaluated domains. This performance advantage can be attributed to a fundamental difference in sampling strategy: while RLPD relies on symmetric random sampling, A^3RL employs an active sampling approach based on advantage alignment.

Unlike RLPD, which treats all transitions uniformly, A^3RL dynamically reevaluates the relevance and on-policyness of each transition as the policy evolves, continuously adjusting its sampling priority to align with the current learning needs. This targeted sampling ensures that the most beneficial transitions are prioritized, directly contributing to faster and more effective policy improvement.

In scenarios involving nearly random offline datasets Fig.1a,1b, datasets containing trajectories from a poorly performing policy, or even medium datasets Fig.1d,1c, useful transitions are often sparse and scattered. Random sampling, as used by RLPD, is likely to miss these valuable data points, leading to suboptimal performance. In contrast, A^3RL 's active sampling strategy effectively identifies and emphasizes these critical transitions, resulting in substantial policy enhancements, as clearly illustrated in Fig. 1.

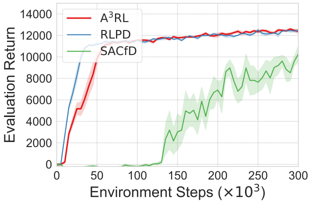
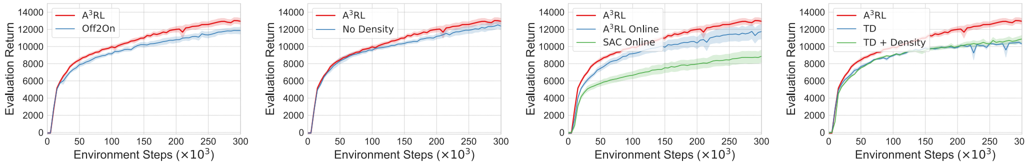


Figure 2: halfcheetah-expert

In expert environments Fig. 2, A^3RL consistently matches or exceeds the performance of RLPD. This superior performance can be attributed to the higher quality of transitions present in medium and expert datasets, compared to random datasets. Consequently, even with a random sampling strategy, RLPD is still likely to encounter useful transitions. However, most offline datasets are provided in a black-box format, where the specifics of the data are unknown. Despite this uncertainty, A^3RL achieves performance that is at least on par with RLPD, demonstrating robustness to varying data quality in these black-box conditions.



(a) advantage term (b) density term (c) purely online setting (d) TD-error priority

Figure 3: **Ablation Studies**: Results of ablation studies on the halfcheetah-random environment.

6.2 Ablation studies

Ablation on advantage term Fig. 3a illustrates the comparison between the performance of A^3RL using advantage-aligned sampling priority and Off2On utilizing solely density ratio ($\sigma = \mathbb{I}^{\text{off}} w(s_i, a_i) + \mathbb{I}^{\text{on}}$), a modified version of balanced experience replay [29]. The results show that A^3RL with the advantage term surpasses its counterpart that only considers online-ness in prioritizing samples in the halfcheetah-random environments. This superiority is attributed to the advantage term, which effectively screens out transitions that are either non-informative or harmful. For example, even if a transition indicates online-ness, it may not provide new information if the policy has already mastered the associated action for that state. By integrating the advantage term, such repetitive transitions are excluded, as the advantage value tends to zero for well-understood transitions.

Ablation on density term Fig. 3b compares the performance of A^3RL to A^3RL with only advantage in sampling priority ($\sigma = \exp(\xi \cdot A)$), without density term. The results consistently show that A^3RL, which incorporates onlineness through the density term $w = d^{\text{on}}/d^{\text{off}}$, outperforms the version that does not. Onlineness measures the likelihood that A^3RL will experience the given transition during the online exploration and exploitation of the current policy. Transitions experienced during online policy enhancement are more advantageous for policy development. In contrast, focusing on transitions that are unlikely to occur during live interactions with the environment can hinder the progression of policy improvement. This result demonstrates the effectiveness of onlineness term.

Ablation on purely online setting and offline data Fig. 3c presents an ablation study comparing regular A^3RL (in red), purely online A^3RL (in blue), and SAC (in green), with neither having access to offline data. A^3RL surpasses its purely online version when utilizing an offline dataset, as the offline data provides a more diverse range of transitions that the online policy might not encounter, effectively demonstrating A^3RL's ability to leverage offline datasets. Moreover, the purely online version of A^3RL outperforms SAC, highlighting A^3RL's robustness in environment setting. The results confirm A^3RL's effectiveness in a purely online environment and its superiority over SAC in online batch scenarios through active advantage-aligned sampling.

Ablation on priority term Fig. 3d presents an ablation study for A^3RL (in red), where we compare two different sampling strategies: PER as detailed in [55] (named as TD in blue), and a modified version incorporating a density ratio (named as TD+Density in green). The TD-error based sampling strategy prioritizes transitions with larger TD-errors. A^3RL significantly outperforms both strategies, illustrating that an active advantage-aligned sampling approach is more effective than prioritizing based on TD-error alone. The superior performance of A^3RL over TD+Density also indicates that prioritizing using the advantage term achieve the better performance compared to the TD-error term.

7 Conclusion

We present A^3RL, a novel algorithm for online RL with offline dataset through a confidence-aware active advantage-aligned sampling strategy. This algorithm is theoretically motivated by the objective of shifting the sampling distribution toward more beneficial transitions to maximize policy improvement. We provide theoretical insights for A^3RL and quantify its enhancement gap. Moreover, we conduct comprehensive experiments with various qualities of offline data, demonstrating that A^3RL outperforms the SOTA RLPD method with significance. We also conduct multiple ablation studies and confirm the importance of each component within the active advantage-aligned formula and its effectiveness to pure online setting as well. While our approach primarily aims to enhance performance, it may result in higher computational costs due to the calculations needed for determining advantage-aligned sampling priorities. Reducing computational demands will be a focus of our future work.

Acknowledgements

We thank Yicheng Luo for the initial discussion and helpful suggestions. This work is supported by the RadBio-AI project (DE-AC02-06CH11357), U.S. Department of Energy Office of Science, Office of Biological and Environment Research, the Improve project under contract (75N91019F00134, 75N91019D00024, 89233218CNA000001, DE-AC02-06-CH11357, DE-AC52-07NA27344, DE-AC05-00OR22725), Exascale Computing Project (17-SC-20-SC), a collaborative effort of the U.S. Department of Energy Office of Science and the National Nuclear Security Administration, the AI-Assisted Hybrid Renewable Energy, Nutrient, and Water Recovery project (DOE DE-EE0009505), and the National Science Foundation under Grant No. IIS 2313131, IIS 2332475, DMS 2413243 and HDR TRIPODS (2216899).

References

[1] András Antos, Csaba Szepesvári, and Rémi Munos. Fitted Q-iteration in continuous action-space mdps. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2007. 16

[2] Philip J Ball, Laura Smith, Ilya Kostrikov, and Sergey Levine. Efficient online reinforcement learning with offline data. *arXiv preprint arXiv:2302.02948*, 2023. 2, 3, 4, 8

[3] Richard Bellman. A Markovian decision process. *Journal of mathematics and mechanics*, pages 679–684, 1957. 3

[4] Marc Brittain, Josh Bertram, Xuxi Yang, and Peng Wei. Prioritized sequence experience replay. *arXiv preprint arXiv:1905.12726*, 2019. 3

[5] Xi Chen, Ali Ghadirzadeh, Tianhe Yu, Jianhao Wang, Alex Yuan Gao, Wenzhe Li, Liang Bin, Chelsea Finn, and Chongjie Zhang. Lapo: Latent-variable advantage-weighted policy optimization for offline reinforcement learning. *Advances in Neural Information Processing Systems*, 35:36902–36913, 2022. 16

[6] Xinyue Chen, Che Wang, Zijian Zhou, and Keith Ross. Randomized ensembled double q-learning: Learning fast without a model. *arXiv preprint arXiv:2101.05982*, 2021. 4

[7] Ching-An Cheng, Andrey Kolobov, and Alekh Agarwal. Policy improvement via imitation of multiple oracles. *Advances in Neural Information Processing Systems*, 33:5587–5598, 2020. 1

[8] Arkady Epshteyn, Adam Vogel, and Gerald DeJong. Active reinforcement learning. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 296–303, 2008. 3

[9] Benjamin Eysenbach, Swapnil Asawa, Shreyas Chaudhari, Sergey Levine, and Ruslan Salakhutdinov. Off-dynamics reinforcement learning: Training for transfer with domain classifiers. *arXiv preprint arXiv:2006.13916*, 2020. 2

[10] Meng Fang, Yuan Li, and Trevor Cohn. Learning how to active learn: A deep reinforcement learning approach. *arXiv preprint arXiv:1708.02383*, 2017. 3

[11] William Fedus, Prajit Ramachandran, Rishabh Agarwal, Yoshua Bengio, Hugo Larochelle, Mark Rowland, and Will Dabney. Revisiting fundamentals of experience replay. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 3061–3071, 2020. 2

[12] Justin Fu, Aviral Kumar, Ofir Nachum, George Tucker, and Sergey Levine. D4RL: Datasets for deep data-driven reinforcement learning. *arXiv preprint arXiv:2004.07219*, 2020. 8, 19

[13] Scott Fujimoto, Herke Hoof, and David Meger. Addressing function approximation error in actor-critic methods. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 1587–1596, 2018. 4

[14] Scott Fujimoto, David Meger, and Doina Precup. Off-policy deep reinforcement learning without exploration. In *International conference on machine learning*, pages 2052–2062, 2019. 4, 16

[15] Tuomas Haarnoja, Aurick Zhou, Kristian Hartikainen, George Tucker, Sehoon Ha, Jie Tan, Vikash Kumar, Henry Zhu, Abhishek Gupta, Pieter Abbeel, et al. Soft actor-critic algorithms and applications. *arXiv preprint arXiv:1812.05905*, 2018. 3, 4

[16] Alex Havrilla, Yuqing Du, Sharath Chandra Raparthy, Christoforos Nalmpantis, Jane Dwivedi-Yu, Maksym Zhuravinskyi, Eric Hambro, Sainbayar Sukhbaatar, and Roberta Raileanu. Teaching

large language models to reason with reinforcement learning. *arXiv preprint arXiv:2403.04642*, 2024. 1

[17] Matteo Hessel, Joseph Modayil, Hado Van Hasselt, Tom Schaul, Georg Ostrovski, Will Dabney, Dan Horgan, Bilal Piot, Mohammad Azar, and David Silver. Rainbow: Combining improvements in deep reinforcement learning. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, 2018. 2, 3

[18] Takuya Hiraoka, Takahisa Imagawa, Taisei Hashimoto, Takashi Onishi, and Yoshimasa Tsuruoka. Dropout q-functions for doubly efficient reinforcement learning. *arXiv preprint arXiv:2110.02034*, 2021. 4

[19] Auke Ijspeert, Jun Nakanishi, and Stefan Schaal. Learning attractor landscapes for learning motor primitives. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2002. 2

[20] David Isele and Akansel Cosgun. Selective experience replay for lifelong learning. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, 2018. 2

[21] Max Jaderberg, Volodymyr Mnih, Wojciech Marian Czarnecki, Tom Schaul, Joel Z Leibo, David Silver, and Koray Kavukcuoglu. Reinforcement learning with unsupervised auxiliary tasks. *arXiv preprint arXiv:1611.05397*, 2016. 2

[22] Sham Kakade and John Langford. Approximately optimal approximate reinforcement learning. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2002. 6

[23] Beomjoon Kim, Amir-massoud Farahmand, Joelle Pineau, and Doina Precup. Learning from limited demonstrations. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2013. 2

[24] Jens Kober, Erhan Oztop, and Jan Peters. Reinforcement learning to adjust robot movements to new situations. In *Twenty-Second International Joint Conference on Artificial Intelligence*, 2011. 1

[25] Jens Kober, J Andrew Bagnell, and Jan Peters. Reinforcement learning in robotics: A survey. *The International Journal of Robotics Research*, 32(11):1238–1274, 2013. 1

[26] Ilya Kostrikov, Ashvin Nair, and Sergey Levine. Offline reinforcement learning with implicit q-learning. *arXiv preprint arXiv:2110.06169*, 2021. 16

[27] David Krueger, Jan Leike, Owain Evans, and John Salvatier. Active reinforcement learning: Observing rewards at a cost. *arXiv preprint arXiv:2011.06709*, 2020. 3

[28] Aviral Kumar, Aurick Zhou, George Tucker, and Sergey Levine. Conservative Q-learning for offline reinforcement learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020. 16

[29] Seunghyun Lee, Younggyo Seo, Kimin Lee, Pieter Abbeel, and Jinwoo Shin. Offline-to-online reinforcement learning via balanced replay and pessimistic Q-ensemble. In *Proceedings of the Conference on Robot Learning (CoRL)*, pages 1702–1712, 2022. 2, 3, 5, 8, 9

[30] Sergey Levine, Aviral Kumar, George Tucker, and Justin Fu. Offline reinforcement learning: Tutorial, review, and perspectives on open problems. *arXiv preprint arXiv:2005.01643*, 2020. 1, 16

[31] Long-Ji Lin. Self-improving reactive agents based on reinforcement learning, planning and teaching. *Machine Learning*, 8:293–321, 1992. 2

[32] Tenglong Liu, Yang Li, Yixing Lan, Hao Gao, Wei Pan, and Xin Xu. Adaptive advantage-guided policy regularization for offline reinforcement learning. *arXiv preprint arXiv:2405.19909*, 2024. 16

[33] Xuefeng Liu, Fangfang Xia, Rick L Stevens, and Yuxin Chen. Cost-effective online contextual model selection. *arXiv preprint arXiv:2207.06030*, 2022. 3

[34] Xuefeng Liu, Songhao Jiang, Archit Vasan, Alexander Brace, Ozan Gokdemir, Thomas Brettin, and R Stevens. DRUGIMPROVER: Utilizing reinforcement learning for multi-objective alignment in drug optimization. In *NeurIPS 2023 Workshop on New Frontiers of AI for Drug Discovery and Development*, 2023. 1

[35] Xuefeng Liu, Takuma Yoneda, Rick L Stevens, Matthew R Walter, and Yuxin Chen. Blending imitation and reinforcement learning for robust policy improvement. *arXiv preprint arXiv:2310.01737*, 2023. 1, 3, 5

[36] Xuefeng Liu, Takuma Yoneda, Chaoqi Wang, Matthew Walter, and Yuxin Chen. Active policy improvement from multiple black-box oracles. In *International Conference on Machine Learning*, pages 22320–22337, 2023. 1, 3, 16

[37] Xuefeng Liu, Chih-chan Tien, Peng Ding, Songhao Jiang, and Rick L Stevens. Entropy-reinforced planning with large language models for drug discovery. *arXiv preprint arXiv:2406.07025*, 2024. 1

[38] Manuel Lopes, Francisco Melo, and Luis Montesano. Active learning for reward estimation in inverse reinforcement learning. In *Proceedings of the Joint European Conference on Machine Learning and Nnowledge Discovery in Databases (ECML PKDD)*, pages 31–46, 2009. 3

[39] Yicheng Luo, Jackie Kay, Edward Grefenstette, and Marc Peter Deisenroth. Finetuning from offline reinforcement learning: Challenges, trade-offs and practical solutions. *arXiv preprint arXiv:2303.17396*, 2023. 2, 16

[40] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, 2013. 1

[41] Rémi Munos, Tom Stepleton, Anna Harutyunyan, and Marc Bellemare. Safe and efficient off-policy reinforcement learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2016. 3

[42] Ashvin Nair, Bob McGrew, Marcin Andrychowicz, Wojciech Zaremba, and Pieter Abbeel. Overcoming exploration in reinforcement learning with demonstrations. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 6292–6299, 2018. 2

[43] Ashvin Nair, Abhishek Gupta, Murtaza Dalal, and Sergey Levine. Awac: Accelerating online reinforcement learning with offline datasets. *arXiv preprint arXiv:2006.09359*, 2020. 2, 8, 16

[44] XuanLong Nguyen, Martin J Wainwright, and Michael Jordan. Estimating divergence functionals and the likelihood ratio by penalized convex risk minimization. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2007. 5

[45] Youngmin Oh, Jinwoo Shin, Eunho Yang, and Sung Ju Hwang. Model-augmented prioritized experience replay. In *International Conference on Learning Representations*, 2021. 2, 3

[46] Doina Precup. Eligibility traces for off-policy policy evaluation. *Computer Science Department Faculty Publication Series*, page 80, 2000. 3

[47] Rafael Figueiredo Prudencio, Marcos ROA Maximo, and Esther Luna Colombini. A survey on offline reinforcement learning: Taxonomy, review, and open problems. *IEEE Transactions on Neural Networks and Learning Systems*, 2023. 1

[48] Yunpeng Qing, Shunyu Liu, Jingyuan Cong, Kaixuan Chen, Yihe Zhou, and Mingli Song. A2po: Towards effective offline reinforcement learning from an advantage-aware perspective. *Advances in Neural Information Processing Systems*, 37:29064–29090, 2024. 16

[49] Aravind Rajeswaran, Vikash Kumar, Abhishek Gupta, Giulia Vezzani, John Schulman, Emanuel Todorov, and Sergey Levine. Learning complex dexterous manipulation with deep reinforcement learning and demonstrations. *arXiv preprint arXiv:1709.10087*, 2017. 2

[50] Sarah Rathnam, Sonali Parbhoo, Weiwei Pan, Susan Murphy, and Finale Doshi-Velez. The unintended consequences of discount regularization: Improving regularization in certainty equivalence reinforcement learning. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 28746–28767, 2023. 16

[51] Stéphane Ross and Drew Bagnell. Efficient reductions for imitation learning. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 661–668, 2010. 1

[52] Stephane Ross and J Andrew Bagnell. Reinforcement and imitation learning via interactive no-regret learning. *arXiv preprint arXiv:1406.5979*, 2014. 1

[53] Gavin A Rummery and Mahesan Niranjan. *On-line Q-learning using connectionist systems*, volume 37. University of Cambridge, Department of Engineering Cambridge, UK, 1994. 1

[54] Baturay Saglam, Furkan B Mutlu, Dogan C Cicek, and Suleyman S Kozat. Actor prioritized experience replay. *arXiv preprint arXiv:2209.00532*, 2022. 2

[55] Tom Schaul, John Quan, Ioannis Antonoglou, and David Silver. Prioritized experience replay. *arXiv preprint arXiv:1511.05952*, 2015. 2, 3, 8, 9

[56] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017. 1

[57] David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, et al. Mastering the game of go without human knowledge. *nature*, 550(7676):354–359, 2017. 1

[58] Samarth Sinha, Jiaming Song, Animesh Garg, and Stefano Ermon. Experience replay with likelihood-free importance weights. In *Proceedings of the Annual Learning for Dynamics and Control Conference (L4DC)*, pages 110–123, 2022. 3, 5

[59] Yuda Song, Yifei Zhou, Ayush Sekhari, J Andrew Bagnell, Akshay Krishnamurthy, and Wen Sun. Hybrid rl: Using both offline and online data can make rl efficient. *arXiv preprint arXiv:2210.06718*, 2022. 2

[60] Richard S Sutton, David McAllester, Satinder Singh, and Yishay Mansour. Policy gradient methods for reinforcement learning with function approximation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 1999. 3

[61] Qi Tian, Kun Kuang, Furui Liu, and Baoxiang Wang. Learning from good trajectories in offline multi-agent reinforcement learning. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, pages 11672–11680, 2023. 2

[62] Hado Van Hasselt, Arthur Guez, and David Silver. Deep reinforcement learning with double Q-learning. In *Proceedings of the National Conference on Artificial Intelligence (AAAI)*, 2016. 4

[63] Hado P Van Hasselt, Matteo Hessel, and John Aslanides. When to use parametric models in reinforcement learning? In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019. 3

[64] Mel Vecerik, Todd Hester, Jonathan Scholz, Fumin Wang, Olivier Pietquin, Bilal Piot, Nicolas Heess, Thomas Rothörl, Thomas Lampe, and Martin Riedmiller. Leveraging demonstrations for deep reinforcement learning on robotics problems with sparse rewards. *arXiv preprint arXiv:1707.08817*, 2017. 2, 8

[65] Ziyu Wang, Tom Schaul, Matteo Hessel, Hado Hasselt, Marc Lanctot, and Nando Freitas. Dueling network architectures for deep reinforcement learning. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 1995–2003, 2016. 2

[66] Ziyu Wang, Alexander Novikov, Konrad Zolna, Josh S Merel, Jost Tobias Springenberg, Scott E Reed, Bobak Shahriari, Noah Siegel, Caglar Gulcehre, Nicolas Heess, et al. Critic regularized regression. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 7768–7778, 2020. 16

[67] Christopher John Cornish Hellaby Watkins. Learning from delayed rewards. 1989. 1

[68] Yang Yue, Bingyi Kang, Xiao Ma, Gao Huang, Shiji Song, and Shuicheng Yan. Offline prioritized experience replay. *arXiv preprint arXiv:2306.05412*, 2023. 2

[69] Han Zheng, Xufang Luo, Pengfei Wei, Xuan Song, Dongsheng Li, and Jing Jiang. Adaptive policy learning for offline-to-online reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 11372–11380, 2023. 2

[70] Henry Zhu, Abhishek Gupta, Aravind Rajeswaran, Sergey Levine, and Vikash Kumar. Dexterous manipulation with deep reinforcement learning: Efficient, general, and low-cost. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 3651–3657, 2019. 2

[71] Brian D Ziebart. *Modeling purposeful adaptive behavior with the principle of maximum causal entropy*. Carnegie Mellon University, 2010. 3

A Theoretical Motivation

In this section, we show that the active advantage-aligned sampling strategy helps mitigate the gap between offline data distribution, online data distribution and the current on-policy distribution, which serves as a main theoretical motivation for designing A^3RL.

Theorem 1 *Suppose the Q-function class is uniformly bounded, and for any Q-function, the corresponding optimal policy lies within the policy function class. Let ϵ^t denote the ℓ_2 error of the Q-function in the critic update step. Let π^t be the policy at iteration t in A^3RL, updated using priority-weighted sampling with $w(s,a)\exp(\xi \cdot A(s,a))$. Then, the following lower bound holds:*

$$J_\alpha^{\pi^{t+1}} - J_\alpha^{\pi^t} \geq J_\alpha^{\pi^\star} - J_\alpha^{\pi^t} - C\sqrt{\epsilon^t} \sup_{s,a} \left| R^t(s,a;\xi) \right|,$$

where $J_\alpha^\pi = \mathbb{E}_{s\sim\rho^\pi, a\sim\pi}\left[\sum_{t=0}^\infty \gamma^t \left(r_t + \alpha\mathcal{H}(\pi(a|s))\right)\right]$ is the entropy-regularized objective, $J_\alpha^{\pi^\star} - J_\alpha^{\pi^t}$ represents the maximum possible improvement if the true Q-function were known, and the function $R^t(s,a;\xi)$ is given by:

$$R^t(s,a;\xi) = \left(\frac{\pi^{t+1}(a \mid s)}{d^{on}(a \mid s)}\right)^{1-\xi} \cdot \frac{\sum_{s',a'} d^{on}(a',s')\pi^{t+1}(a' \mid s')^\xi}{d^{on}(a \mid s)^\xi} \cdot \frac{d^{\pi^{t+1}}(s)}{d^{on}(s)}.$$

Proof: (Proof of Theorem 1). Define visitation measures

$$d_h^\pi(s,a) = \mathbb{E}_{a\sim\pi(\cdot|s)}\left[\mathbb{1}(s_h = s, a_h = a)\right], \quad d^\pi(s,a) = \frac{1}{1-\gamma}\sum_{h=1}^\infty \gamma^h d_h^\pi(s,a).$$

Consider a sufficiently small one-step update in the policy network with step-size η. Define $J_\alpha^\pi = \mathbb{E}_{s\sim\rho^\pi, a\sim\pi}[\sum_{t=0}^\infty \gamma^t \left(r_t + \alpha\mathcal{H}\left(\pi\left(a|s\right)\right)\right)]$. Let $\widetilde{\pi}$ be the policy from the last iteration. In the following, we abbreviate $\mathbb{E}_\pi[\cdot]$ as $\mathbb{E}[\cdot]$.

$$\begin{aligned}
V^\pi - V^{\widetilde{\pi}} &= \mathbb{E}\left[\langle \pi, Q^\pi - \alpha\log\pi\rangle - \langle\widetilde{\pi}, Q^{\widetilde{\pi}} - \alpha\log\widetilde{\pi}\rangle_{\mathcal{A}}\right] \\
&= \mathbb{E}\left[\langle \pi, Q^\pi - Q^{\widetilde{\pi}}\rangle_{\mathcal{A}} + \langle \pi - \widetilde{\pi}, Q^{\widetilde{\pi}}\rangle_{\mathcal{A}} - \alpha\left\langle\pi, \log\pi\right\rangle + \alpha\left\langle\widetilde{\pi}, \log\widetilde{\pi}\right\rangle\right] \\
&= \mathbb{E}\left[\langle \pi, r + \gamma\mathbb{P}V^\pi - r + \gamma\mathbb{P}V^\pi\rangle + \langle \pi - \widetilde{\pi}, Q^{\widetilde{\pi}}\rangle_{\mathcal{A}} - \alpha\left\langle\pi, \log\pi\right\rangle + \alpha\left\langle\widetilde{\pi}, \log\widetilde{\pi}\right\rangle\right] \\
&= \mathbb{E}\left[\gamma\big\langle \pi, \mathbb{P}\big(V^\pi - V^{\widetilde{\pi}}\big)\big\rangle_{\mathcal{A}} + \big\langle \pi - \widetilde{\pi}, Q^{\widetilde{\pi}}\big\rangle_{\mathcal{A}} - \alpha\left\langle\pi, \log\pi\right\rangle + \alpha\left\langle\widetilde{\pi}, \log\widetilde{\pi}\right\rangle\right],
\end{aligned}$$

Using this iterative form, we conclude that

$$\begin{aligned}
J_\alpha^\pi - J_\alpha^{\widetilde{\pi}} &= \mathbb{E}\left[\sum_{h=1}^\infty \gamma^i \left(\big\langle \pi_i - \widetilde{\pi}_i, Q_i^{\widetilde{\pi}}\big\rangle_{\mathcal{A}} - \alpha\left\langle\pi_i, \log\pi_i\right\rangle + \alpha\left\langle\widetilde{\pi}_i, \log\widetilde{\pi}_i\right\rangle\right)\right] \\
&= \mathbb{E}_{d^\pi}\left[\big\langle \pi - \widetilde{\pi}, Q^{\widetilde{\pi}}\big\rangle_{\mathcal{A}} - \alpha\left\langle\pi, \log\pi\right\rangle + \alpha\left\langle\widetilde{\pi}, \log\widetilde{\pi}\right\rangle\right].
\end{aligned}$$

Recall our definition of $\sigma(s,a)$ that

$$\sigma(s,a) = \exp(\xi\widehat{A}^{\widetilde{\pi}}(s,a)) \cdot \frac{d^{\text{on}}(s,a)}{\mu(s,a)}, \tag{8}$$

where $\mu(\cdot,\cdot)$ is the distribution in the sampled batch and $d^{\text{on}}(\cdot,\cdot)$ is the online distribution. Note that the advantage function $\widehat{A}^{\widetilde{\pi}}(s,a) = \widehat{Q}^{\widetilde{\pi}}(s,a) - \alpha\log\sum_{a'}\exp(\alpha^{-1}\widehat{Q}(s,a'))$ is calculated using policy $\widetilde{\pi}$ and Q function $\widehat{Q}^{\widetilde{\pi}}$ obtained from the last iteration in the above formula. Let us define $\pi_{\phi^\star}$ as the optimal policy under the current Q function $\widetilde{Q}$:

$$\begin{aligned}
\pi^\star(\cdot \mid s) &= \arg\min_\pi \mathrm{KL}\left(\pi(\cdot \mid s) \,\middle\|\, \frac{\exp(\alpha^{-1}Q^{\widetilde{\pi}}(s,\cdot))}{\widetilde{Z}_\alpha(s)}\right) \\
&= \arg\max_\pi \big\langle \pi\left(\cdot|s\right), Q^{\widetilde{\pi}}\left(s,\cdot\right) - \alpha\log\pi\left(\cdot|s\right)\big\rangle_{\mathcal{A}} \propto \exp(\alpha^{-1}A^{\widetilde{\pi}}(s,\cdot)).
\end{aligned}$$

where $\widetilde{Z}_\alpha(s)$ is the normalization factor at state s for the exponential of the Q function, and $A^{\widetilde{\pi}}(s,\cdot)$ is the advantage function under policy $\widetilde{\pi}$. Recall by policy optimization:

$$\widehat{\pi} = \arg\max_{\pi} \mathbb{E}_\mu \left[\sigma\left(s,a\right)\left\langle \pi\left(\cdot|s\right), \widehat{Q}^{\widetilde{\pi}}\left(s,\cdot\right) - \alpha \log \pi\left(\cdot|s\right)\right\rangle_{\mathcal{A}}\right],$$

where $\widehat{Q}^{\widetilde{\pi}}$ is the estimated Q function at the current iteration. In the above formula, μ is the sampled data distribution and σ is the quantity calculated in (8). Suppose we take some function class π_ϕ which contains the optimal one-step policy improvement $\pi^\star$ and also the optimization target $\widehat{\pi}$. Using a shift of distribution, we have

$$\begin{aligned}
\mu(s,a)\sigma(s,a) = \mu(s,a) \cdot \frac{d^{\text{on}}(s,a)}{\mu(s,a)} \cdot \exp(\xi \widehat{A^{\widetilde{\pi}}}(s,a)) &= d^{\text{on}}(s,a) \cdot \widehat{\pi}(a\,|\,s)^{\xi} \\
&= d^{\widehat{\pi}}(s,a) \cdot \frac{d^{\text{on}}(s)}{d^{\widehat{\pi}}(s)} \cdot \frac{d^{\text{on}}(a\,|\,s)}{\widehat{\pi}(a\,|\,s)^{1-\xi}} \propto \rho(s,a),
\end{aligned}$$

where we define $\rho(s,a)$ as the probability density induced by the above distribution. Here, the first ratio $d^{\text{on}}(s)/d^{\pi^\star}(s)$ is the state-drift between the online data and the next-step optimal policy. Since the online batches are refreshing as the algorithm proceeds, the ratio will be close to 1. The second ratio term characterizes the drift caused by a mismatch in the policy. Intuitively, as we know the policy $\widetilde{\pi}$ from the last iteration, we can use this information to further boost the alignment between the online policy and the next-step policy. Suppose the Q function is learned up to ϵ error, that is

$$\mathbb{E}_\rho\left[(Q^{\widetilde{\pi}}(s,a) - \widehat{Q}^{\widetilde{\pi}}(s,a))^2\right] \le \epsilon.$$

Then, we have performance difference lemma that

$$\begin{aligned}
J_\alpha^{\widehat{\pi}} - J_\alpha^{\pi^\star} &= \mathbb{E}_{d^{\widehat{\pi}}}\left[\left\langle \widehat{\pi}, Q^{\widetilde{\pi}}\right\rangle_{\mathcal{A}} - \alpha\left\langle \widehat{\pi}, \log\widehat{\pi}\right\rangle - \left(\left\langle \pi^\star, Q^{\widetilde{\pi}}\right\rangle_{\mathcal{A}} - \alpha\left\langle \pi^\star, \log\pi^\star\right\rangle\right)\right] \\
&= \mathbb{E}_{d^{\widehat{\pi}}}\left[\left\langle \widehat{\pi}, Q^{\widetilde{\pi}}\right\rangle_{\mathcal{A}} - \alpha\left\langle \widehat{\pi}, \log\widehat{\pi}\right\rangle - \left(\left\langle \widehat{\pi}, \widehat{Q}^{\widetilde{\pi}}\right\rangle_{\mathcal{A}} - \alpha\left\langle \widehat{\pi}, \log\widehat{\pi}\right\rangle\right)\right] \\
&\quad + \mathbb{E}_{d^{\widehat{\pi}}}\left[\left\langle \widehat{\pi}, \widehat{Q}^{\widetilde{\pi}}\right\rangle_{\mathcal{A}} - \alpha\left\langle \widehat{\pi}, \log\widehat{\pi}\right\rangle - \left(\left\langle \pi^\star, \widehat{Q}^{\widetilde{\pi}}\right\rangle_{\mathcal{A}} - \alpha\left\langle \pi^\star, \log\pi^\star\right\rangle\right)\right] \\
&\quad + \mathbb{E}_{d^{\widehat{\pi}}}\left[\left\langle \pi^\star, \widehat{Q}^{\widetilde{\pi}}\right\rangle_{\mathcal{A}} - \alpha\left\langle \pi^\star, \log\pi^\star\right\rangle - \left(\left\langle \pi^\star, Q^{\widetilde{\pi}}\right\rangle_{\mathcal{A}} - \alpha\left\langle \pi^\star, \log\pi^\star\right\rangle\right)\right] \\
&\ge \mathbb{E}_{d^{\widehat{\pi}}}\left[\left\langle \widehat{\pi} - \pi^\star, Q^{\widetilde{\pi}} - \widehat{Q}^{\widetilde{\pi}}\right\rangle_{\mathcal{A}}\right] \\
&\ge -\sup_{s,a}\left|\frac{\pi^\star(a\,|\,s)}{\widehat{\pi}(a\,|\,s)} - 1\right| \cdot \mathbb{E}_{d^{\widehat{\pi}}}[|Q^{\widetilde{\pi}} - \widehat{Q}^{\widetilde{\pi}}|] \ge -C \cdot \mathbb{E}_{d^{\widehat{\pi}}}[|Q^{\widetilde{\pi}} - \widehat{Q}^{\widetilde{\pi}}|]
\end{aligned}$$

where C is an absolute constant given that both $Q^{\widetilde{\pi}}$ and $\widehat{Q}^{\widetilde{\pi}}$ are uniformly bounded. Here, the first inequality holds by the policy optimization step where we upper bound the second term by zero, and the last inequality holds by the assumption that the Q function class is uniformly bounded. Now, by a shift of distribution

$$\mathbb{E}_{d^{\widehat{\pi}}}[|Q^{\widetilde{\pi}} - \widehat{Q}^{\widetilde{\pi}}|] = \mathbb{E}_\rho\left[|Q^{\widetilde{\pi}} - \widehat{Q}^{\widetilde{\pi}}| \cdot \frac{d^{\widehat{\pi}}(s,a)}{\rho(s,a)}\right] \le \sqrt{\mathbb{E}_\rho[(Q^{\widetilde{\pi}} - \widehat{Q}^{\widetilde{\pi}})^2]} \cdot \sup_{s,a}\left|\frac{d^{\widehat{\pi}}(s,a)}{\rho(s,a)}\right|.$$

Let’s look at the distribution ratio

$$\begin{aligned}
\frac{d^{\widehat{\pi}}(s,a)}{\rho(s,a)} &= \frac{\widehat{\pi}(a\,|\,s)}{\widehat{\pi}(a\,|\,s)^{\xi} \cdot d^{\text{on}}(a\,|\,s)^{1-\xi}} \cdot \frac{\sum_{s',a'} d^{\text{on}}(a',s')\widehat{\pi}(a'\,|\,s')^{\xi}}{d^{\text{on}}(a\,|\,s)^{\xi}} \cdot \frac{d^{\widehat{\pi}}(s)}{d^{\text{on}}(s)} \\
&= \left(\frac{\widehat{\pi}(a\,|\,s)}{d^{\text{on}}(a\,|\,s)}\right)^{1-\xi} \cdot \frac{\sum_{s',a'} d^{\text{on}}(a',s')\widehat{\pi}(a'\,|\,s')^{\xi}}{d^{\text{on}}(a\,|\,s)^{\xi}} \cdot \frac{d^{\widehat{\pi}}(s)}{d^{\text{on}}(s)}.
\end{aligned}$$

Therefore, the policy improvement is guaranteed by

$$J_\alpha^{\widehat{\pi}} - J_\alpha^{\widetilde{\pi}} = J_\alpha^{\widehat{\pi}} - J_\alpha^{\pi^\star} + J_\alpha^{\pi^\star} - J_\alpha^{\widetilde{\pi}} \ge J_\alpha^{\pi^\star} - J_\alpha^{\widetilde{\pi}} - C \cdot \sqrt{\epsilon} \cdot \sup_{s,a}\left|\frac{d^{\widehat{\pi}}(s,a)}{\rho(s,a)}\right|.$$

This completes the proof.

□

Now we give a formal proof for Lemma 1.

Lemma 1 *For the bandit case with $d^{on}(a) \propto \exp(\beta_1 r(a))$ and $\pi^{t+1}(a) \propto \exp(\beta_2 r(a))$ for $\beta_2 > \beta_1$, the coefficient $\sup_a R^t(a;\xi)$ decreases as ξ increases within the range $\xi \in (0, 1 - \beta_1/\beta_2)$.*

Proof:(Proof of Lemma 1) Under the reparameterization $d^{\text{on}}(a) \propto \exp(\beta_1 r(a))$ and $\pi^{t+1}(a) \propto \exp(\beta_2 r(a))$, we have for the coefficient $R^t(a;\xi)$ that

$$\begin{aligned} R^t(a;\xi) &\propto \exp\big(\big((1-\xi)(\beta_2-\beta_1) - \xi\beta_1\big)\cdot r(a)\big) \\ &= \exp\big(\big((1-\xi)\beta_2 - \beta_1\big)\cdot r(a)\big)\,. \end{aligned}$$

Within the range $\xi \in (0, 1-\beta_1/\beta_2)$, we always have $(1-\xi)\beta_2 - \beta_1 > 0$. Hence, the largest coefficient always occurs on action $\widetilde{a} = \arg\max_{a'} r(a')$. In addition, we consider the following ratio

$$\begin{aligned} \log\left(\frac{R(a;\xi)}{R(a;0)}\right) &= -\xi\log(\pi^{t+1}(a)) + \log\left(\sum_{a'} d^{\text{on}}(a')\pi^{t+1}(a')^{\xi}\right) \\ &= -\xi\beta_2 r(a) + \log\left(\sum_{a'}\exp((\beta_1+\beta_2\xi)r(a'))\right)\,. \end{aligned}$$

Consider the gradient of $\log\left(\sum_{a'}\exp((\beta_1+\beta_2\xi)r(a'))\right)$ with respect to ξ:

$$\frac{\partial}{\partial\xi}\log\left(\sum_{a'}\exp((\beta_1+\beta_2\xi)r(a'))\right) = \frac{\sum_{a'}\beta_2 r(a')\exp((\beta_1+\beta_2\xi)r(a'))}{\sum_{a'}\exp((\beta_1+\beta_2\xi)r(a'))} - \beta_2 r(a).$$

Note that the largest probability ratio happens for $\widetilde{a} = \arg\max_{a'} r(a')$. Since the softmax is strictly less than the argmax when r has different values in each action, the above derivative for action $\widetilde{a}$ is negative, meaning that by increasing ξ, the value of $R(\widetilde{a};\xi)$ will decrease. As $\sup_a R(a;\xi) = R(\widetilde{a};\xi)$ by our previous discussion, we complete the proof. □

B Additional Related Work

Offline to online RL In an effort to mitigate the sample complexity of online RL [36], offline RL utilizes fixed datasets to train policies without online interaction, however it can be prone to extrapolation errors that lead to overestimation of state-action values. Recent off-policy actor-critic methods [14, 26, 28, 66] seek to mitigate these issues by limiting policy learning to the scope of the dataset, thereby minimizing extrapolation error. Strategies for reducing extrapolation error include value-constrained approaches [28] that aim for conservative value estimates and policy-constrained techniques [43] that ensure the policy remains close to the observed behavior in the data. There are several works that leverage advantage estimation to guide policy improvement in purely offline RL, such as LAPO [5], A2PR [32], and A2PO [48]. However, they are not well-suited for online settings because they fail to consider the importance of "onlineness," measured by the density ratio, to align with the needs of online RL exploration and exploitation. Additionally, they do not account for uncertainty in advantage estimation.

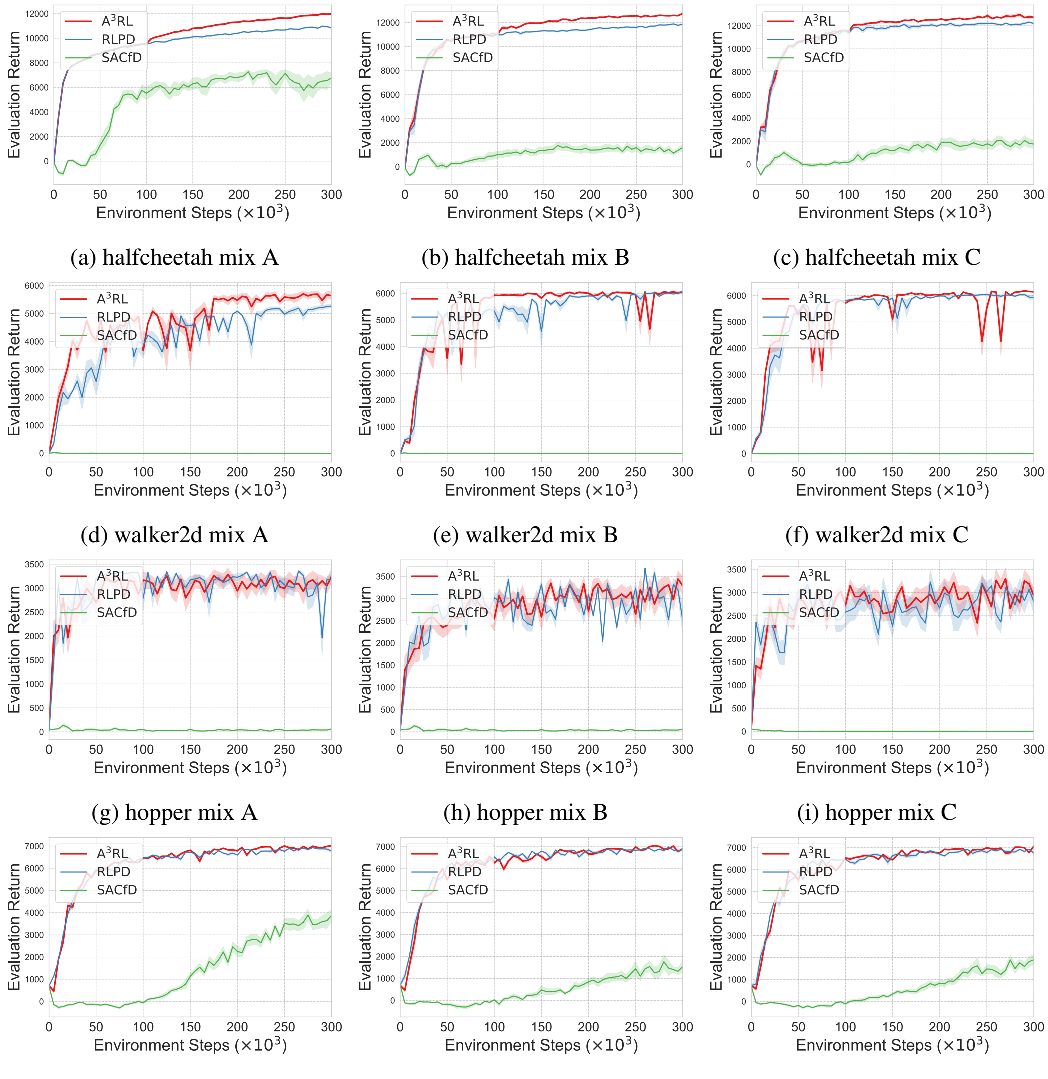
While offline RL methods can outperform the dataset's behavior policy, they rely entirely on static data [30]. When the dataset has comprehensive coverage, methods like FQI [1] or certainty-equivalence model learning [50] can efficiently find near-optimal policies. However, in practical scenarios with limited data coverage, policies tend to be suboptimal. One approach to addressing this suboptimality is to follow offline RL with online fine-tuning, however as discussed above, existing methods are prone to catastrophic forgetting and performance drops during fine-tuning [39]. In contrast, A^3RL begins with online RL while incorporating offline data to enhance the policy, selectively leveraging offline data to facilitate online policy improvement.

C Experimental Details

In order to ensure fair evaluation, all baselines and ablation studies are assessed using an equal number of environment interaction steps. We average results over 10 seeds to obtain the final result. One standard error of the mean is shaded for each graph.

C.1 Additional experimental results.

We explored whether different mixtures of offline datasets can be exploited by A^3RL. In particular, for the D4RL locomotion halfcheetah, walker2d, hopper and ant environments in Fig. 4, mix A corresponds to having the offline dataset consisting of 100% of the -simple Minari dataset, mix B corresponds to 100% -simple and 5% -medium, while mix C corresponds to 100% -simple and 10% -medium. Those proportions were chosen due to the recognizable difference in the performance of RLPD under these different settings. In particular, we observed that all RLPD runs with the offline dataset consisting of 100% of the -simple dataset and no less than 30-40% of the -medium dataset achieve similar performance. Meanwhile at lower percentages such as 5% and 10%, there is a difference between runs of RLPD, which implies that there is significant impact from the offline dataset quality to the bootstrapping from offline transitions.



(a) halfcheetah mix A

(b) halfcheetah mix B

(c) halfcheetah mix C

(d) walker2d mix A

(e) walker2d mix B

(f) walker2d mix C

(g) hopper mix A

(h) hopper mix B

(i) hopper mix C

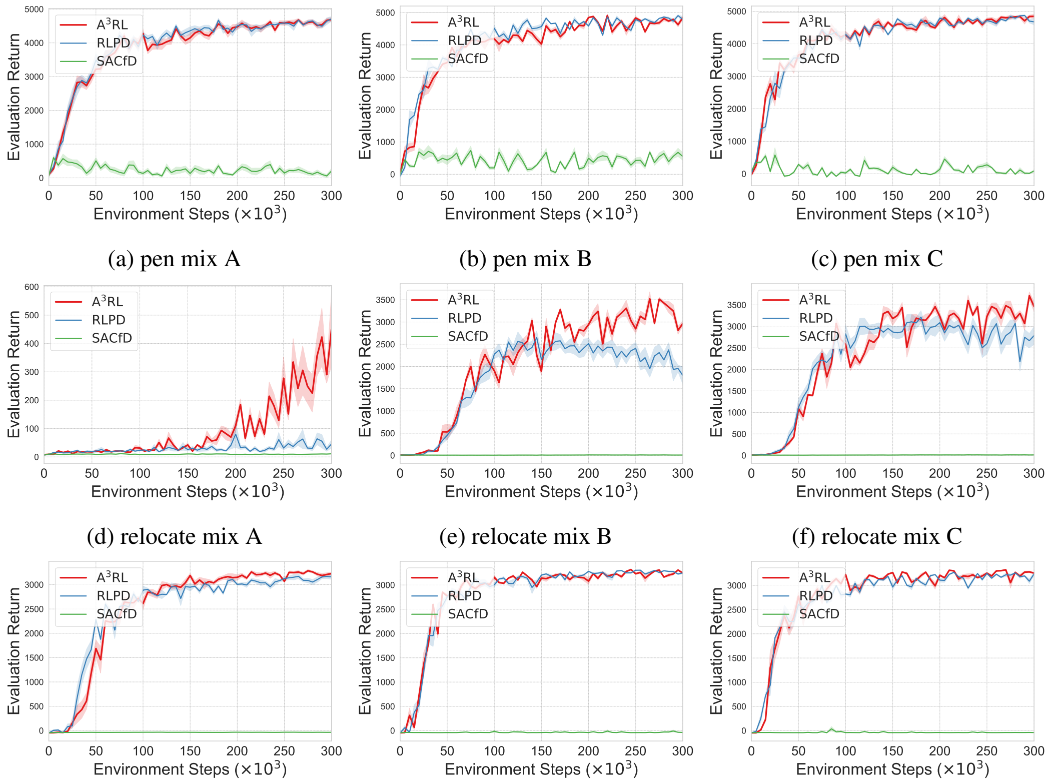
(j) ant mix A

(k) ant mix B

(l) ant mix C

Figure 4: A^3RL vs RLPD vs SACfD on halfcheetah, walker2d, hopper and ant with different offline dataset mixtures. A^3RL outperforms or performs comparably to RLPD across diverse black-box environments.

For the Adroit environments (Fig. 5) pen, relocate and door, the -cloned dataset plays the role of the -simple dataset above, while the -expert plays the role of the -medium dataset above. The mixtures were generated similarly. A^3RL robustly outperforms, or at least performs on par with, RLPD across diverse black-box environments.



(a) pen mix A

(b) pen mix B

(c) pen mix C

(d) relocate mix A

(e) relocate mix B

(f) relocate mix C

(g) door mix A

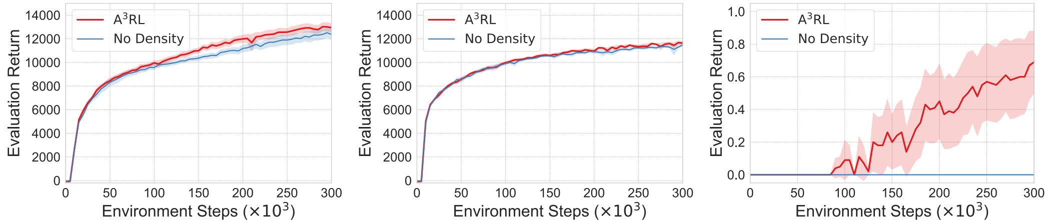
(h) door mix B

(i) door mix C

Figure 5: A^3RL vs RLPD vs SACfD on pen, relocate and door tasks with different offline dataset mixtures. A^3RL outperforms or performs comparably to RLPD across diverse black-box environments.

C.2 Additional ablation studies.

Ablation on density term. Fig. 6 presents further ablation studies on the density term for A^3RL. We see the distinction in the effectiveness of the density term is more significant over harder tasks like antmaze-medium-play.



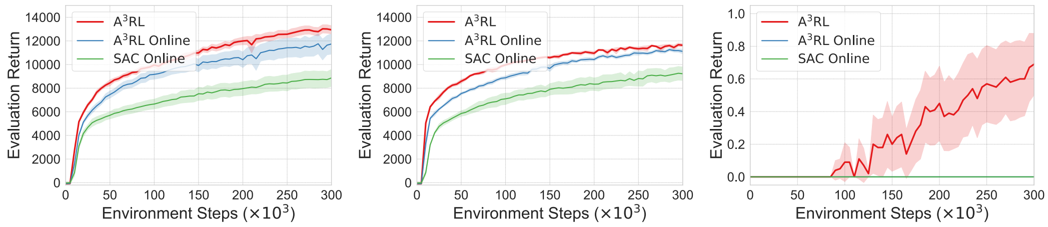
(a) halfcheetah-random

(b) halfcheetah-medium-replay

(c) antmaze-medium-play

Figure 6: **Ablation Studies**: A^3RL vs A^3RL without density term.

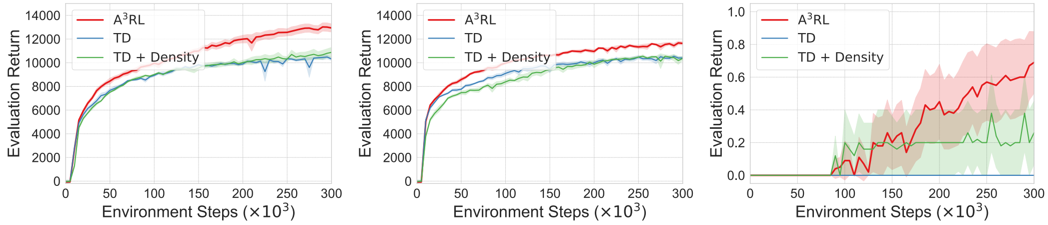
Ablation on purely online setting. Fig. 7 presents further ablation studies on A^3RL interacting with the environment in a purely online manner, i.e., the algorithm does not utilize access to offline data. It is consistent throughout tested environments that A^3RL is able to leverage offline data effectively, especially in harder tasks like antmaze-medium-play where purely online A^3RL fails to learn in the same number of steps.



(a) halfcheetah-random (b) halfcheetah-medium-replay (c) antmaze-medium-play

Figure 7: **Ablation Studies**: A^3RL vs purely online A^3RL vs purely online SAC.

Ablation on priority term. Fig. 8 presents further ablation studies on the priority term for A^3RL, where we compare it against the sampling strategy that solely uses TD-error as the priority term, and another that combines the density term with TD-error. The superior performance of A^3RL over TD+Density over tested environments indicates that prioritizing using the advantage term achieves better performance compared to the canonical TD-error term.



(a) halfcheetah-random (b) halfcheetah-medium-replay (c) antmaze-medium-play

Figure 8: **Ablation Studies**: A^3RL vs PER (TD) vs PER with Density (TD+Density).

Training and evaluation environments. Fig. 9 presents snapshots of tested D4RL locomotion tasks: halfcheetah, walker2d, hopper, ant and antmaze. halfcheetah, walker2d, hopper and ant have dense rewards, while antmaze has sparse rewards, and all environments are equipped with continuous state and action spaces.

In the halfcheetah environment, the 2D agent resembles a simplified cheetah model with a torso and lined legs, with the objective of forward locomotion and maintaining balance while maximizing speed. In the walker2d environment, the 2D humanoid agent has 2 legs and multiple joints, with the objective of stable bipedal walking without falling. In the ant environment, the agent is a 3D quadrupedal agent with multiple joints and degrees of freedom, with the objective of moving forward efficiently while maintaining balance. For all of these environments, rewards are given for velocity to encourage the agent to move forward efficiently while maintaining balance, and several offline datasets, per [12], with varying characteristics, as detailed below, were tested.

Fig. 10 presents snapshots of tested Adroit manipulation tasks: pen, relocate, and door. These environments involve a simulated 28-DoF robotic arm interacting with objects in a 3D space and are characterized by sparse rewards and continuous state and action spaces.

In the pen environment, the robotic arm is tasked with reorienting a pen to a target orientation in free space. In the relocate environment, the arm must pick up a ball and move it to a target position, requiring coordinated grasping and relocation of an object in 3D space. The door task involves the robotic arm grasping and pulling open a door, necessitating nuanced contact dynamics and manipulation under constraints. For all of these environments, rewards are sparse and typically only given upon task completion, increasing the exploration difficulty.

Offline dataset type	**Description**
-expert-v2	1M samples from policy trained to completion with SAC
-medium-v2	1M samples from policy trained to 1/3 of expert
-medium-replay-v2	Replay buffer of policy trained to medium
-random-v2	1M samples from randomly initialized policy

Table 1: Locomotion offline dataset.

In the antmaze environment, the aforementioned ant agent is placed in a maze environment and must navigate from a defined start point to a goal. Rewards are binary: 1 for reaching the goal and 0 otherwise. Varying sizes of the maze were tested: umaze (U-shaped), medium and large; which are naturally also of increasing difficulty.

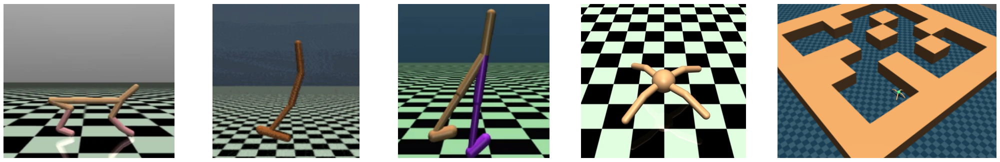


Figure 9: **Environments**: halfcheetah, walker2d, hopper, ant and antmaze respectively.

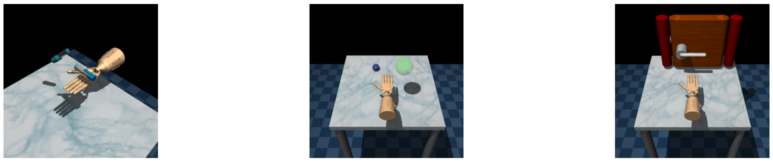


Figure 10: **Environments**: pen, relocate and door respectively.

C.3 Computing infrastructure and wall-time comparison.

We performed our experiments on a cluster that includes CPU nodes (approximately 280 cores) and GPU nodes (approximately 110 NVIDIA GPUs, ranging from Titan X to A6000, set up mostly in 4- and 8-GPU configurations). On the same cluster, the wall run time of A^3RL is approximately 1.5 times the run time of regular RLPD and is comparable to Off2On.

C.4 Hyperparameters and architectures.

We list the hyperparameters used for A^3RL in Table 2.

Parameter	**Value**
Batch size	256
Gradient steps G	20
MLP Architecture	2-Layer
Network width	256 Units
Discount	0.99
Learning rate	3×10^{-4}
Ensemble size E	10
ζ	0.3
ξ	0.03
Optimizer	Adam

Table 2: A^3RL hyperparameters.