Offline Reinforcement Learning via Tsallis Regularization

Anonymous authors
Paper under double-blind review

Abstract

Offline reinforcement learning (RL) focuses on learning a good policy from a fixed dataset. The dataset is generated by an unknown behavior policy through interactions with the environment and contains only a subset of the state-action spaces. Standard off-policy algorithms often perform poorly in this setting, suffering from errorneously optimistic values incurred by the out-of-distribution (OOD) actions not present in the dataset. The optimisim cannot be corrected as no further interaction with the environment is possible. Imposing divergence regularization and in-sample constraints are among the most popular methods to overcoming the issue by ensuring that the learned policy stays close to the behavior policy to minimize the occurrence of OOD actions. This paper proposes Tsallis regularization for offline RL, which aligns the induced *sparsemax* policies to the in-sample constraint. Sparsemax interpolates existing methods utilizing hard-max and softmax policies, in that only a subset of actions contributes non-zero action probability as compared to softmax (all actions) and hard-max (single action). We leverage this property to model the behavior policy and show that under several assumptions the learned sparsemax policies may have sparsity-conditional KL divergence to the behavior policy, making Tsallis regularization especially suitable for the Behavior Cloning methods. We propose two actor-critic algorithms, Tsallis In-sample Actor-Critic (Tsallis InAC) and Tsallis Advantage Weighted Actor-Critic (Tsallis AWAC), respectively generalizing InAC (Xiao et al., 2023) and AWAC (Nair et al., 2021) and analyze their performance in standard Mujoco baselines.

1 Introduction

Reinforcement learning (RL) has achieved impressive successes in various domains through learning from online interactions with the environment (Mnih et al., 2015; Silver et al., 2017; Andrychowicz et al., 2020). However, online RL is often less suited to real-world domains, especially when acting unconstrained in an environment can be expensive or dangerous. Offline RL instead addresses the problem of learning good policies completely from a given dataset generated following unknown policies. The goal of offline RL is to learn policies which outperform—or at least match—the policies used to generate the dataset.

However, many of the difficulties of offline RL stem from not using online interaction. Standard off-policy RL algorithms tend to perform poorly, due to the well-known extrapolation error or out-of-distribution (OOD) action problem: improving the learned policy beyond the level of behavior policy requires estimating values of state-action pairs not present in the dataset. Optimistic estimates will bias the agent into favoring the absent actions in the policy improvement stage, leading to a vicious loop (Fujimoto et al., 2019; Kostrikov et al., 2022). Since the environment cannot be sampled, the bias can never be alleviated by visiting the region and correcting the value estimate. It is worth noting the extrapolation problem does not occur in the tabular case but rather is the result of smoothness of function approximators (Gulcehre et al., 2021; Dadashi et al., 2021).

A popular branch of offline RL is behavior cloning (BC) (Pomerleau, 1988), referred to as imitation-based methods in (Xu et al., 2022). BC based methods enforce the learned policy to stay close to or reproduce the behavior policy (Dadashi et al., 2021; Ghasemipour et al., 2021; Nair et al., 2021; Siegel et al., 2020; Wang et al., 2020; Wu et al., 2022; 2020). This is often achieved in two ways: (1) by using **divergence regularization** $D(\pi_t(\cdot|s)||\pi_{\mathcal{D}}(\cdot|s))$ when updating policy to penalize large deviation from the learned policy π_t to the behavior policy $\pi_{\mathcal{D}}$ (Brandfonbrener et al., 2021; Jaques et al., 2020; Kostrikov et al., 2021; Wu et al., 2020; Osa et al., 2023), usually D is chosen to be the KL divergence but other divergences such as MMD (Kumar et al., 2019), Fisher's divergence (Kostrikov et al., 2021) or Shannon entropy (Xiao et al., 2023) have also been used; (2) imposing an **in-sample constraint** to the updates (Fujimoto et al., 2019; Kostrikov et al., 2022; Xiao et al., 2023) where the target hard max operator $\max_a Q(s,a)$ in Q-learning is replaced to the *in-sample* maximum $\max_{a:\pi_{\mathcal{D}}(a|s)>0} Q(s,a)$. This scheme has been recently extended to the in-sample softmax $\sum_{a:\pi_{\mathcal{D}}(a|s)>0} e^{Q(s,a)}$ (Xiao et al., 2023).

In this paper, we propose to use a general but less studied class of regularizers that interpolates softmax and hard-max−−the Tsallis regularizers, specifically, Tsallis entropy and Tsallis KL divergence−−as the choice of D. While Tsallis entropy and Tsallis KL divergence have recently been investigated in online RL (Lee et al., 2018; 2020; Pacchiano et al., 2021; Zhan et al., 2023; Zhu et al., 2023), they have never been utilized in the offline RL context. Tsallis entropy (resp. Tsallis KL) is a strict generalization of Shannon entropy (resp. KL). Different from the softmax policy induced by Shannon entropy and KL that has full support (Azar et al., 2012; Kozuno et al., 2019; Vieillard et al., 2020), Tsallis regularization induces the sparsemax policy that truncates actions with low values, i.e. setting their probability to zero. We link action truncation to the in-sample constraint $\pi_{\mathcal{D}}(a|s) > 0$, arriving at the assumption that the behavior policy is within the sparsemax policy class and collects in the offline dataset a subset of actions with high values. Intuitively, this assumption invites learning an improved sparsemax policy with support within the that of the behavior policy, which renders Tsallis regularization especially suited to BC or imitation-based methods. We formalize this intuition in Section 4 by showing that the KL divergence between $\pi_{\mathcal{D}}$ and the learned policy π_t, all within the sparsemax class, may be upper bounded depending on the Tsallis entropic index controlling the sparsity.

By combining in-sample constraint and Tsallis regularization, we propose two actor-critic algorithms: Tsallis In-sample Actor-Critic (Tsallis InAC), based on Tsallis entropy; and Tsallis advantage weighted actor-critic (Tsallis AWAC) based on Tsallis KL divergence. Tsallis InAC extends the In-sample Actor-Critic (InAC) (Xiao et al., 2023) to the $q > 1$ domain. Surprisingly, contrary to the stable behavior of InAC, Tsallis InAC behaves like a BC method: it is among the best performers on the expert-level datatsets but degrades for non-expert datasets. On the other hand, for non-expert datasets Tsallis AWAC is more competitive and outperforms the original AWAC by a large margin, by simply changing the exponential advantage function in AWAC (Nair et al., 2021) to the q-exponential.

2 Background

We model our problem as a markov decision process (MDP) expressed by the tuple $(\mathcal{S}, \mathcal{A}, P, r, \gamma)$. $\mathcal{S} \subseteq \mathbb{R}^n$ is the set of states, $\mathcal{A} \subseteq \mathbb{R}^m$ is the set of actions. $P(\cdot|s,a)$ denotes transition probability over the state space given state-action pair (s,a), and $r(s,a)$ defines the reward associated with that transition. $\gamma \in [0,1)$ is the discount factor. A policy $\pi(\cdot|s)$ is a mapping from the state space to distributions over actions. The state-action value function starting from (s,a) following policy π is defined as $Q_\pi(s,a) = \mathbb{E}_\pi\left[\sum_{t=0}^{\infty}\gamma^t r(s_t,a_t)|s_0=s,a_0=a\right]$. There exists a stationary optimal policy that maximizes the cumulative return with a fixed point $Q_*(s,a)$ satisfying the Bellman optimality equation $Q_*(s,a) = r(s,a) + \gamma\mathbb{E}_{s'\sim P(\cdot|s,a)}[\max_{a'} Q_*(s',a')]$. The optimal policy π_* can then be extracted by simply acting greedily with respect to the optimal action value function: in this case, $\pi_*(a|s) = \arg\max_\pi \mathbb{E}_\pi\left[Q_*(s,a)\right], \forall s$ is a deterministic policy.

Since deterministic policies are susceptible to errors and noises, stochastic policies are often employed in the literature by augmenting the arg max problem above with a regularizer: $\mathbb{E}_\pi\left[Q_*(s,a) - \Omega(\pi(\cdot|s))\right]$, where $\Omega(\pi(\cdot|s)) \in \mathbb{R}^{|\mathcal{S}|}$ is the regularizer convex in π. Popular choices for Ω include the negative Shannon entropy $-\tau\mathcal{H}\left(\pi(\cdot|s)\right) := \tau\sum_a \pi(a|s)\ln\pi(a|s)$, which encourages the policy to be uniform with τ weighting the effect. The maximum of the problem $\max_\pi \sum_a \pi(a|s)Q(s,a) + \tau\mathcal{H}\left(\pi(\cdot|s)\right)$ is attained at the well-known log-partition function $\tau\ln\sum_a \exp\left(\tau^{-1}Q(s,a)\right)$ when the policy is the Boltzmann softmax distribution $\pi(a|s) \propto \exp\left(\tau^{-1}Q(s,a)\right)$, where $\propto$ denotes *proportional to* up to a constant not depending on actions. KL divergence $D_{KL}(\pi(\cdot|s)\,||\,\mu(\cdot|s)) := \sum_a \pi(a|s)\ln\frac{\pi(a|s)}{\mu(a|s)}$ is another popular choice where μ is the reference policy (Azar et al., 2012; Rawlik et al., 2013; Vieillard et al., 2020). Using KL divergence as Ω penalizes large deviation from μ. By choosing μ to be the uniform distribution, KL divergence recovers the negative Shannon entropy case. The KL-regularized optimal policy takes the form $\pi(a|s) \propto \mu(a|s)\exp\left(\tau^{-1}Q(s,a)\right)$, where we overloaded the coefficient τ for Shannon entropy.

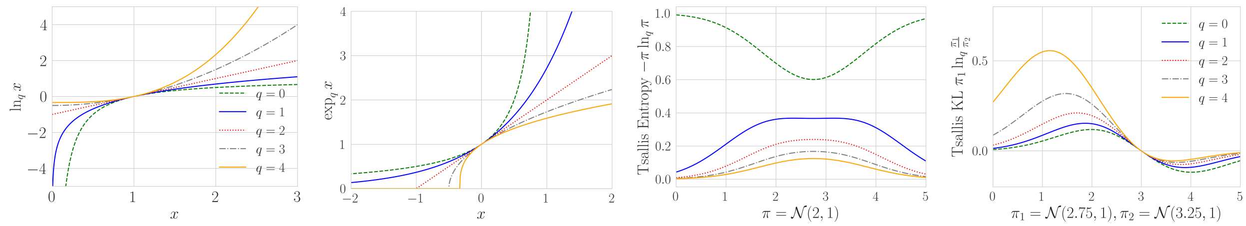

Figure 1: (From left to right) q-logarithm, q-exponential, Tsallis entropy and Tsallis KL divergence of Gaussian policies. Note that $q=0$ is shown here only for an illustration purpose. We consider $q>0$ in this paper for theoretically sound regularizers. When $q=1$, the above functions recover their standard counterparts.

2.1 Tsallis Regularization and Sparsemax Policies

In this paper, we consider a broad class of less studied entropic regularizers as Ω: Tsallis entropy and Tsallis KL divergence. We can define these regularizers using q-logarithm in a similar manner to the standard logarithm. For $q \in \mathbb{R}_+$, we define q-logarithm as $\ln_q x = \frac{x^{q-1}-1}{q-1}$ and its unique inverse function q-exponential $\exp_q x = [1+(q-1)x]_+^{\frac{1}{q-1}}$, where $\{\cdot\}_+ = \max\{\cdot, 0\}$, see Figure 1 for an illustration. As q gets larger, $\exp_q x$ becomes more flat (second plot). Notice that $\exp_q$ is only invertible when $x > -\frac{1}{q-1}$. On the other hand, $\ln_q x$ is more peaked than the standard logarithm for input $x>1$ and more flat otherwise (Ding & Vishwanathan, 2010). Therefore, for $0 \leq \pi \leq 1$, the Tsallis entropy component $-\pi \ln_q \pi$ is more flat than the Shannon entropy (third plot). Note that when $q \leq 0$, Tsallis entropy becomes a convex function and Tsallis KL a concave function, therefore they are not valid regularizers (Geist et al., 2019). We consider exclusively $q>0$ in this paper, and $q=0$ is shown in Figure 1 only for an illustration purpose. We can define the Tsallis entropy using q-logarithm (Tsallis, 2009): $S_q(\pi(\cdot|s)) = -k\sum_a \pi(a|s) \ln_q \pi(a|s), k \in \mathbb{R}$. When $q \to 1$, the q-logarithm (resp. q-exponential) recovers the standard logarithm (resp. exponential) and hence Tsallis entropy degenerates to Shannon entropy. When $q=\infty$, the regularizer vanishes. When $k=\frac{1}{2}, q=2$, we arrive at the most important non-trivial case: Tsallis sparse entropy $S_2(\pi(\cdot|s)) := \frac{1}{2}\sum_a \pi(a|s)\left(1-\pi(a|s)\right)$ (Chow et al., 2018; Lee et al., 2018). The name sparse entropy comes from the fact that the regularizer leads to sparse support of the resulting *sparsemax* policy (Blondel et al., 2020; Martins & Astudillo, 2016). We compare sparsemax against two other commonly used policies argmax and softmax in Figure 2.

For $q \neq 1, 2, \infty$, the resulting policy does not have closed-form solution. But we can resort to approximation (Zhu et al., 2023) for these cases and unify the policy expression by the following:

$$\pi(a|s) = \exp_q\left(\frac{Q(s,a)}{\tau} - \psi\left(\frac{Q(s,\cdot)}{\tau}\right)\right), \ \psi\left(\frac{Q(s,\cdot)}{\tau}\right) \doteq \frac{\sum_{a\in K(s)} \frac{Q(s,a)}{\tau} - 1}{|K(s)|} + \frac{1}{q-1}. \tag{1}$$

We call the policy Eq.(1) *sparsemax* for all $q \in \mathbb{R}_+ \setminus \{1\}$, since they truncate actions by the definition of q-exponential. $K(s)$ is the set of highest-value actions satisfying $1 + i\frac{Q(s,a_{(i)})}{\tau} > \sum_{j=1}^{i} \frac{Q(s,a_{(j)})}{\tau}$, with $a_{(j)}$ denotes the action with j-th largest value. Intuitively, the policy first sorts actions $a_{(1)}, \dots, a_{(|\mathcal{A}|)}$ and then compute the threshold ψ. Suppose $Q(s, a_{(j+1)}) < \psi < Q(s, a_{(j)})$, then $a_{(j+1)}, \dots, a_{(|\mathcal{A}|)}$ are truncated and have zero probability of being selected. The actions $a_{(1)}, \dots, a_{(j)}$ are called allowable actions and collected in the set $K(s)$. The degree of truncation can be controlled by either q or τ. As q gets larger, by definition of $\exp_q$ the truncation becomes stronger, as all $x < -\frac{1}{q-1}$ will be truncated; when τ becomes larger, more actions are collected in $K(s)$. Note our definition of the normalization ψ is different from $\tilde{\psi}$ used by prior work (Lee et al., 2018; Chow et al., 2018) which consider only $q=2$. However, we can link them by $\psi = \tilde{\psi} + \frac{1}{q-1}$, see (Zhu et al., 2023) for more details. When $q=2$, $\max_\pi \sum_a \pi(a|s)Q(s,a) + \tau S_2(\pi(\cdot|s))$ attains its maximum at $\frac{\tau}{2}\sum_{a\in K(s)}\left(\frac{Q(s,a)}{\tau}\right)^2 - \tilde{\psi}\left(\frac{Q(s,\cdot)}{\tau}\right)^2 + \frac{\tau}{2}$. We write the optimal Tsallis entropy regularized policy by $\pi_*(a|s) \propto \exp_q\left(\frac{Q(s,a)}{\tau}\right)$ to emphasize the similarity to the softmax policy.

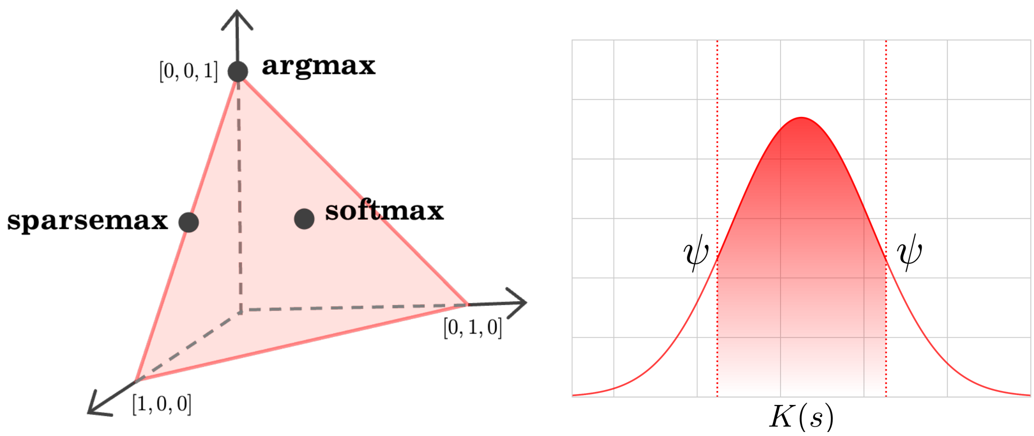

Figure 2: (Left) Comparison between argmax, softmax and sparsemax on the probability simplex. Argmax produces a deterministic policy residing on the vertices, while a softmax policy lies inside the simplex. By contrast, a sparsemax policy lives on the border. (Right) Sparsemax acting on a Gaussian policy by truncating actions with value below the threshold ψ. Actions with value larger than ψ are collected in the set $K(s)$.

Tsallis KL divergence $D^q_{KL}(\pi(\cdot|s)\,||\,\mu(\cdot|s)) = \sum_a \pi(a|s) \ln_q \frac{\pi(a|s)}{\mu(a|s)}$ is a generalization of KL divergence into the domain of q-logarithm (Furuichi et al., 2004). Its has been recently considered by Zhu et al. (2023) as $\Omega(\pi)$ in the online RL setting. Tsallis KL is more mode-seeking: the mismatch between π and μ will be penalized more with larger q, as can be seen from the rightmost subfigure in Figure 1. Note that the definition of Tsallis KL in this paper is different from (Furuichi et al., 2004; Zhu et al., 2023). However, they are the same by the $2-q$ duality, see (Zhu et al., 2023, Appendix A). Zhu et al. (2023) also proved that the regularized optimal policy takes a similar form to the optimal KL-regularized policy: $\pi(a|s) = \mu(a|s) \exp_q \left(\frac{Q(s,a)}{\tau} - \psi'\left(\frac{Q(s,\cdot)}{\tau}\right)\right)$. Note that the normalization function ψ' may not be the same as ψ, due to conditioning on the reference policy μ. We write the optimal regularized policy as $\pi(a|s) \propto \mu(a|s) \exp_q\left(\frac{Q(s,a)}{\tau}\right)$. This policy form can be interpreted as truncating actions (with $\exp_q$) but additionally conditioning on the support of μ, i.e., $\pi(a|s)$ has non-zero probability only when $\mu(a|s) > 0$. I.e., Tsallis KL regularized policies are also a member of the sparsemax policies.

2.2 Offline Reinforcement Learning

We consider the problem of offline RL, where the agent cannot interact with the environment and instead learn from a fixed dataset $\mathcal{D} = \{(s, a, r, s')_{1:N}\}$ collected by some unknown behavior policy $\pi_{\mathcal{D}}$. The dataset $\mathcal{D}$ typically contains only a small subset of the $\mathcal{S} \times \mathcal{A}$ space. Standard off-policy algorithms are known to suffer from extrapolation error referring to erroneously optimistic acation values for out-of-distribution actions due to generalization capibility of function approximators. Unlike online RL, where the OOD actions can lead to more sampling around the low sample region and eventually correction of the values, in offline RL the correction is impossible since no further interaction with the environment is allowed.

We position this paper in the popular BC/imitation-based literature, where the goal of learning is to reproduce the near-expert behavior policy $\pi_{\mathcal{D}}$ (Dadashi et al., 2021; Fujimoto et al., 2019; Fujimoto & Gu, 2021; Nair et al., 2021). In this conext, explicit or implicit constraints are usually used to enforce the proximity between learned policies and $\pi_{\mathcal{D}}$ to minimize the effect of OOD actions. Explicit constraints can be implemented via density models (Ghasemipour et al., 2021; Wu et al., 2022) or in-sample constraints (Fujimoto et al., 2019; Kostrikov et al., 2022; Xiao et al., 2023) to avoid querying OOD actions. Implicit constraints are often achieved via a divergence regularizer D that is added to the objective of policy improvement: $\max_\pi \mathbb{E}_{s\sim\mathcal{D}}\left[\mathbb{E}_{a\sim\pi(\cdot|s)}\left[Q(s,a)\right] - \tau D(\pi(\cdot|s)||\pi_{\mathcal{D}}(\cdot|s))\right]$, where D is typically chosen as KL divergence. The regularization leads to the policy form $\pi(a|s) \propto \pi_{\mathcal{D}}(a|s) \exp\left(\tau^{-1} Q(s,a)\right)$ where the learned policy conditions on the support of the behavior policy and weighted by exponential of action values (advantage) (Peng et al., 2020; Siegel et al., 2020; Nair et al., 2021). However, as shown by (Rudner et al., 2021), KL regularization can lead to pathologies of the learned policies such as vanishing variances and exploding gradients. In this paper, we propose to use the Tsallis KL regularization which generalizes KL divergence and offers more flexibility over the standard KL choice (e.g., AWAC). However, investigation of pathologies is beyond the scope and we leave it to future work.

Recently, there has seen a growing body of BC/imitation-based methods featuring a variety of divergences. Ke et al. (2019) discussed imitation learning via the lens of f-divergence minimization and proposed variational solutions. Xu et al. (2023) proposed a framework based on the α-divergence (Belousov & Peters, 2019) and showed that standard KL can be recovered as a special case. In the same context, they showed that Conservative Q-learning (Kumar et al., 2020) corresponds to the χ^2 regularization. In this paper, we consider regularization by Tsallis KL divergence and Tsallis entropy, inducing a general class of sparsemax policies. Offline RL with sparsity has also been discussed very recently: Xu et al. (2022; 2023) showed that α-divergence with $\alpha = -1$ also leads to sparsity. By contrast, we consider general sparsemax policies induced by Tsallis regularizers for all $q > 1$. Li et al. (2023) proposed to use q-Gaussian distribution (Suyari & Tsukada, 2005) which is a special case of the q-exponential policy. In this paper we do not specify the functional form of policies, but investigating the benefits of specific q-exponential policy parametrizations is an interesting future direction.

3 Matching In-sample constraint by Sparsemax Truncation

To alleviate the OOD error, Fujimoto et al. (2019) proposed the in-sample Bellman optimality equation that modifies the Q-learning update target to only for the actions present in the dataset:

$$Q_{*,\pi_{\mathcal{D}}}(s,a) = r(s,a) + \gamma \mathbb{E}_{s'\sim P(\cdot|s,a)}\left[\max_{a':\pi_{\mathcal{D}}(a'|s')>0} Q_{*,\pi_{\mathcal{D}}}(s',a')\right]. \tag{2}$$

Recently, Kostrikov et al. (2022) proposed Implicit Q-learning (IQL) that extended equation 2 to distributional RL implemented via expectile regression and showed promising performance. However, as noted by Xiao et al. (2023), IQL may be skewed towards suboptimal trajectories. Instead, they proposed in-sample softmax which is more straightforward to perform support-constrained sampling. Their in-sample softmax Bellman optimality equation is:

$$Q_{*,\pi_{\mathcal{D}}}(s,a) = r(s,a) + \gamma \mathbb{E}_{s'\sim P(\cdot|s,a)}\left[\tau \ln \sum_{a':\pi_{\mathcal{D}}(a'|s')>0} \exp\left(\tau^{-1} Q_{*,\pi_{\mathcal{D}}}(s',a')\right)\right]. \tag{3}$$

In-sample softmax policy takes the form $\pi^{\texttt{softmax}}_{*,\pi_{\mathcal{D}}}(a|s) \propto \pi_{\mathcal{D}}(a|s)\exp\left(\frac{Q_{*,\pi_{\mathcal{D}}}(s,a)}{\tau} - \ln \pi_{\mathcal{D}}(a|s)\right)$, the subtracted term $\ln \pi_{\mathcal{D}}$ is to make sure not tightly follow the behavior policy when it is not good.

On the other hand, the fact that the softmax policies always have full support indicates there is a persistent gap to the optimal action and the maximum Eq. (2). Since the goal is to improve beyond the behavior policy, we may desire that the action candidates gradually narrow down to a much smaller subset containing the maximizer. Moreover, by design in-sample softmax does not enforce proximity to the behavior policy by subtracting $\ln \pi_{\mathcal{D}}$, suggesting that InAC does not require the learned policy to be close to the behavior policy or narrow down the support set. While this may be desirable when the dataset is flooded with poor trajectories, in-sample softmax may learn slower in the BC context.

Let us consider replacing $\tau\mathcal{H}\left(\pi(\cdot|s)\right)$ to $\tau S_q(\pi(\cdot|s))$. The seemingly simple replacement, however, embodies a fundamentally different assumption: we can link the fact that the dataset contains only a subset of actions to sparsemax behavior policies. Indeed, if we assume the dataset was generated by the behavior policy in a manner such that an action is sampled with probability proportional to its value (Kostrikov et al., 2021), then we can model the behavior policy by a sparsemax policy. This assumption is mild since softmax policy is a member of sparsemax when $q = 1$. Sparsemax can be seen as interpolating hardmax and softmax, in the sense that sparsemax policies have non-zero probabilities only for high-valued actions as opposed to hardmax (resp. softmax) that assigns probability to only one action (resp. every action). By properly specifying q and/or τ, a sparsemax policy can either retain the same action support as softmax or narrow it down by truncating actions.

To formalize the idea, let us assume the behavior policy is within the sparsemax policy class induced by Tsallis entropy $\pi_{\mathcal{D}}(a|s) \propto \exp_q\left(\frac{Q_{\pi_{\mathcal{D}}}(s,a)}{\tau_{\mathcal{D}}}\right), \sum_{a\in K_{\mathcal{D}}(s)} \pi_{\mathcal{D}}(a|s) = 1$, where $\tau_{\mathcal{D}}$ is an unknown coefficient and $K_{\mathcal{D}}(s)$ denotes the set of actions present in the dataset. Therefore, we can replace the in-sample constraint $a : \pi_{\mathcal{D}}(a|s) > 0$ to the truncation criterion $a \in K_{\mathcal{D}}(s)$. This replacement hints at two potential benefits of applying in-sample Tsallis regularization: (i) suppose the action values are fixed (e.g., by training to convergence), then a sparsemax policy could extract a new subset of allowable actions from $K_{\mathcal{D}}$, and this procedure could continue until there is only the highest-valued action in the set. Note that this process can be stopped by properly setting τ (or $q = 1$) to maintain the current support. In this case, we may further deduce the second benefit: (ii) every newly updated sparsemax policy has its support within or equal to the last sparsemax policy (depending on q). Therefore, it may be possible to characterize the distance between consecutive sparsemax policies, and eventually the distance to the behavior policy.

We discuss (i) here and leave (ii) to the next section. To begin with, we write down the updating rule for sparsemax similar to Eqs. (2), (3):

$$Q_{*,\pi_{\mathcal{D}}}(s,a) = r(s,a) + \gamma \mathbb{E}_{s'\sim P(\cdot|s,a)}\left[\max_{\pi} \sum_{a'\in K_{\mathcal{D}}(s)} \pi(a'|s')\left(Q_{*,\pi_{\mathcal{D}}}(s',a') + \tau S_q(\pi(\cdot|s'))\right)\right].$$

$$\pi_{t+1,\pi_{\mathcal{D}}}(a|s) \propto \exp_q\left(\frac{Q_{*,\pi_{\mathcal{D}}}(s,a)}{\tau}\right), \quad \sum_{a\in K_{\mathcal{D}}(s)} \pi_{t+1,\pi_{\mathcal{D}}}(a|s) = 1. \tag{4}$$

Sparsemax interpolates softmax and hard-max. The max term inside the bracket is known as the q-maximum (Lee et al., 2020). Similar to softmax operators (Asadi & Littman, 2017; Azar et al., 2012), q-maximum is an approximation to the maximum but with the degree controlled by q. More importantly, with Eq. (4), a new set of allowable actions $K_{t,q}(s)$ is extracted from $K_{\mathcal{D}}(s)$ depending on q and iteration t. The set satisfies the condition $K_{t,q} \preceq K_{\mathcal{D}}$: i.e. $|K_{t,q}| \leq |K_{\mathcal{D}}|$ and support constraint $\pi_t \preceq \pi_{\mathcal{D}}$. Let us consider $q = 2$ where the q-maximum has an analytic solution. From Section 2.1 it is clear that:

$$Q_{*,\pi_{\mathcal{D}}}(s,a) = r(s,a) + \gamma \mathbb{E}_{s'\sim P(\cdot|s,a)}\left[\frac{\tau}{2}\sum_{a'\in K_{t,2}(s)}\left(\frac{Q_{*,\pi_{\mathcal{D}}}(s',a')}{\tau}\right)^2 - \tilde{\psi}\left(\frac{Q_{*,\pi_{\mathcal{D}}}(s',\cdot)}{\tau}\right)^2 + \frac{\tau}{2}\right]. \tag{5}$$

The constraint change is because $K_{t,2} \preceq K_{\mathcal{D}}$, i.e. $\pi_{t+1}(a|s) = 0, \forall a \in K_{\mathcal{D}}(s) \setminus K_{t,2}(s)$. When $K_{t,2}$ retains all the actions from $K_{\mathcal{D}}$, we have $K_{t,2} = K_{\mathcal{D}}$. However, it should be noted that we assumed the action values were fixed. Therefore, the relationship $K_{\mathcal{D}} \succeq K_{1,q} \succeq K_{2,q} \succeq \cdots \succeq K_{t,q}$ is generally not true since the ordering of action values can change when they are updated.

The above scheme regularizes both policy improvement and policy evaluation. Another possibility is to regularize only policy improvement by choosing Tsallis KL divergence as the regularizer, such that the policy is updated by $\max_\pi \mathbb{E}_{s\sim\mathcal{D}}\left[\mathbb{E}_{a\sim\pi(\cdot|s)}\left[Q(s,a)\right] - \tau D^q_{KL}(\pi(\cdot|s)\,||\,\pi_{\mathcal{D}}(\cdot|s))\right]$. The regularized optimal policy becomes $\pi_{t+1,\pi_{\mathcal{D}}}(a|s) \propto \pi_{\mathcal{D}}(a|s)\exp_q\left(\frac{Q_{t,\pi_{\mathcal{D}}}(s,a)}{\tau}\right)$. Therefore, the in-sample constraint $a \in K_{\mathcal{D}}(s)$ can be dropped since the optimal policy conditions on $\pi_{\mathcal{D}}(a|s)$. We introduce the implementation of both schemes in Section 5.

4 In-sample Sparsemax Policies Have q-bounded KL Divergence

A sparsemax policy can be expressed as a q-exponential policy. Therefore, existing theoretical results on q-statistics (Yamano, 2002; Ding & Vishwanathan, 2010) may provide a clue to characterizing the similarity between two sparsemax policies. We are especially interested in characterizing the distance between a learned policy and the behavior policy, which is particularly important for BC/imitation-based methods (Fujimoto & Gu, 2021; Fujimoto et al., 2019; Wu et al., 2022; 2020).

However, bounding the potentially unbounded KL divergence is in general a difficult task, if not impossible (McAllester & Stratos, 2020). Instead, we do not seek to provide a general upper bound but rather aim at quantifying the dependency of the similarity on some intermediate variable. To this end, we introduce q as the intermediate variable and investigate the KL distance conditional on q. We can then decompose KL as:

$$
\begin{aligned}
&D_{KL}(\pi_t(\cdot|s) \,||\, \pi_{\mathcal{D}}(\cdot|s)) = \mathbb{E}_{a\sim\pi_t(\cdot|s)}\left[\ln \pi_t(a|s) - \ln \pi_{\mathcal{D}}(a|s)\right] \\
&= \mathbb{E}_{a\sim\pi_t(\cdot|s)}\left[\underbrace{\ln \pi_t(a|s) - \ln_q \pi_t(a|s)}_{(i)} + \underbrace{\ln_q \pi_t(a|s) - \ln_q \pi_{\mathcal{D}}(a|s)}_{(ii)} + \underbrace{\ln_q \pi_{\mathcal{D}}(a|s) - \ln \pi_{\mathcal{D}}(a|s)}_{(iii)}\right].
\end{aligned} \tag{6}
$$

Therefore, bounding respectively the three terms allows us to provide a q-conditional bound on the KL divergence. To bound these log/q-log differences, we prove the following fact:

$$\ln x - \ln_q x = (q-1)\left[\frac{d}{dq}\ln_q x - \ln x \ln_q x\right].$$

We can verify this is true by working on the right-hand side:

$$
\begin{aligned}
&(q-1)\left[\frac{d}{dq}\ln_q x - \ln x \ln_q x\right] = (q-1)\left[\frac{d}{dq}\frac{x^{q-1}-1}{q-1} - \ln x \ln_q x\right] \\
&= (q-1)\left[\frac{(x^{q-1}-1)'(q-1) - (x^{q-1}-1)(q-1)'}{(q-1)^2} - \ln x \ln_q x\right] \\
&= (q-1)\left[\frac{(q-1)x^{q-1}\ln x - (x^{q-1}-1)}{(q-1)^2} - \ln x \ln_q x\right] \\
&= x^{q-1}\ln x - \ln_q x - (q-1)\ln x \ln_q x = ((q-1)\ln_q x + 1)\ln x - \ln_q x - (q-1)\ln x \ln_q x \\
&= \ln x - \ln_q x.
\end{aligned}
$$

Now to bound the log/q-log differences, we assume that both the learned policy and the behavior policy are sparsemax:

$$
\begin{aligned}
(i): \ln \pi_t(a|s) - \ln_q \pi_t(a|s) &= (q-1)\left[\frac{d}{dq}\ln_q \pi_t(a|s) - \ln_q \pi_t(a|s)\ln \pi_t(a|s)\right] \\
&= \pi_t^{q-1}(a|s)\ln \pi_t(a|s) - \ln_q \pi_t(a|s)\left(1 + (q-1)\ln \pi_t(a|s)\right) \\
&\leq \pi_t^{q-1}(a|s)\ln \pi_t(a|s) + \frac{1}{q-1} + \ln \pi_t(a|s) \\
&\leq \left(\pi_t^{q}(a|s) - \pi_t^{q-1}(a|s)\right) + \pi_t(a|s) - \frac{q-2}{q-1},
\end{aligned} \tag{7}
$$

where we leveraged the monotonicity of q-logarithm; and $\ln x \leq x - 1$, $\ln_q \exp_q(x) = x$ when $x > 0$. Considering the policy is a q-exponential policy $\pi_t(a|s) = \exp_q\left(\frac{Q_{t-1}(s,a)}{\tau} - \psi\left(\frac{Q_{t-1}(s,\cdot)}{\tau}\right)\right)$ and $\exp_q(x) = [1+(q-1)x]_+^{\frac{1}{q-1}}$, we must have $\pi_t(a|s) > 0 \Leftrightarrow a \in K_{t-1,q}(s) \Leftrightarrow -\frac{1}{q-1} \leq \frac{Q_{t-1}(s,a)}{\tau} - \psi\left(\frac{Q_{t-1}(s,\cdot)}{\tau}\right) \leq 0$ (Ding & Vishwanathan, 2010). **If $a \notin K_{t-1,q}(s)$, then $\ln \pi_t(a|s) = -\infty$ and the KL term is unbounded.** The same fact is exploited to yield an upper bound $\frac{1}{q-1}$ for (ii). Note that the same holds true for the Tsallis KL policy $\pi_t(a|s) = \mu(a|s)\exp_q\left(\frac{Q_{t-1}(s,a)}{\tau} - \psi\left(\frac{Q_{t-1}(s,\cdot)}{\tau}\right)\right)$. Since $\ln_q x$ is monotonically increasing, we have that $-\frac{1}{q-1} \leq \ln_q \mu(a|s)\exp_q\left(\frac{Q_{t-1}(s,a)}{\tau} - \psi\left(\frac{Q_{t-1}(s,\cdot)}{\tau}\right)\right) \leq \ln_q \exp_q\left(\frac{Q_{t-1}(s,a)}{\tau} - \psi\left(\frac{Q_{t-1}(s,\cdot)}{\tau}\right)\right)$. Same by monotonicity, we can bound (iii) depending on q by:

$$
\begin{aligned}
&(iii): \ln_q \pi_{\mathcal{D}}(a|s) - \ln \pi_{\mathcal{D}}(a|s) = -(q-1)\left[\frac{d}{dq}\ln_q \pi_{\mathcal{D}}(a|s) - \ln_q \pi_{\mathcal{D}}(a|s)\ln \pi_{\mathcal{D}}(a|s)\right] \\
&\quad \leq -\pi_{\mathcal{D}}^{q-1}(a|s)\ln \pi_{\mathcal{D}}(a|s) \leq -\pi_{\mathcal{D}}^{q-1}(a|s)\left(1 - \frac{1}{\pi_{\mathcal{D}}(a|s)}\right) = \pi_{\mathcal{D}}^{q-2}(a|s) - \pi_{\mathcal{D}}^{q-1}(a|s).
\end{aligned} \tag{8}
$$

Putting all terms together, we arrive at the following q-dependent upper bound:

$$D_{KL}(\pi_t(\cdot|s)\,||\,\pi_{\mathcal{D}}(\cdot|s)) \leq \begin{cases} |K_{t,q}(s)|\ \Pi_{\mathcal{D},t,q}, & \text{if } \pi_t \preceq \pi_{\mathcal{D}},\ 1 < q < \infty \\ \infty, & \text{otherwise} \end{cases} \tag{9}$$

where $\Pi_{\mathcal{D},t,q} := \pi_t^q(a|s) - \pi_t^{q-1}(a|s) + \pi_t(a|s) - \frac{q-3}{q-1} + \pi_{\mathcal{D}}^{q-2}(a|s) - \pi_{\mathcal{D}}^{q-1}(a|s)$, and the leading term $|K_{t,q}(s)|$ came from upper bounding the expectation $\mathbb{E}_{a\sim\pi_t(\cdot|s)}$ with all allowable actions in the set $K_{t,q}(s)$.

We may interpret the term $\pi_t(a|s)$ as a baseline having a fixed power $\pi_t^{q=1}(a|s)$. When $q = 1$, the upper bound may be ∞. On the other hand, when $q \to \infty$, the learned policy π_t approaches the $\arg\max$ (Lee et al., 2020), and the upper bound approaches zero. Intuitively, when $q = \infty$, $K_{t,q}(s)$ has only one action, which corresponds to assuming the behavior policy is an $\arg\max$ and it is identified by $\pi_t(a|s) = 1$, therefore the KL is zero when the their supports agree, or ∞ otherwise.

We can replace the reference policy $\pi_{\mathcal{D}}$ in equation 9 to a learned policy π_{t-1} if the assumption of fixed action values holds, since $\pi_t \preceq \pi_{t-1}$. Therefore, the bound provides a means to understanding the distance between sparsemax policies. For $q = 1$ (the in-sample softmax case) the bound is not useful and simply states the KL divergence may be unbounded. On the other hand, choosing any $q > 1$ brings an upper bound of at most $4|K_{t,q}(s)|$. When $q = 2$, the in-sample sparsemax has KL divergence to the behavior policy bounded by $|K_{t,q}(s)|\,(\pi_t(a|s) - \pi_{\mathcal{D}}(a|s) + 2)$. In general, there is a trade-off between the difference of power-policies π^q and the cardinality of $K_{t,q}(s)$: $K_{t,q}(s)$ tends to collect less actions when $q \to \infty$. On the down side, it should be noted that the bound is only applicable to discrete action spaces. Moreover, staying close to the behavior policy may not always be preferable, especially when the dataset contains too many poor trajectories.

5 In-sample Sparsemax Actor-Critic

We propose two actor-critic algorithms to implement Tsallis regularization: Tsallis In-Sample Actor-Critic (Tsallis InAC) based on Tsallis entropy; and Tsallis Advantage Weighted Actor-Critic (Tsallis AWAC) based on Tsallis KL divergence. To fulfill the in-sample constraint we leverage only those actions present in the offline dataset but not sampled actions to learn the actor. The architecture is similar to soft actor-critic (Haarnoja et al., 2018): let θ denote the parametrization of the critic Q_θ, ϕ the actor π_ϕ and ζ the state value function V_ζ. Another network ω is trained by maximum likelihood $-\mathbb{E}_{(s,a)\sim\mathcal{D}}\left[\log \pi_\omega(a|s)\right]$ to imitate the behavior policy $\pi_\omega \approx \pi_{\mathcal{D}}$.

Before deriving the Tsallis InAC/Tsallis AWAC loss functions, we recall the actor loss for AWAC:

$$\mathcal{L}^{\texttt{AWAC}}_{\text{actor}}(\phi) = -\mathbb{E}_{(s,a)\sim\mathcal{D}}\left[\exp\left(\frac{Q_\theta(s,a) - V_\zeta(s)}{\tau}\right)\ln \pi_\phi(a|s)\right], \tag{10}$$

which is a common actor loss function for advantage-weighted regression methods (Wang et al., 2020; Siegel et al., 2020; Nair et al., 2021; Xu et al., 2023; Garg et al., 2023). Intuitively, minimizing $\mathcal{L}^{\texttt{AWAC}}_{\text{actor}}(\phi)$ results in maximizing the log-likelihood of actions in the dataset $a \sim \mathcal{D}$ and implicitly minimizing the likelihood for these not in the dataset, as the probabilities of all actions should sum to one. The degree of maximization is controlled by the exponential advantage of an action $\exp\left(\frac{Q_\theta(s,a)-V_\zeta(s)}{\tau}\right)$: when an action has high $Q_\theta(s,a) - V_\zeta(s)$, the policy gets updated to increase its probability more, which may be problematic when the value estimates are poor. By contrast, in-sample softmax (Xiao et al., 2023) compensates for this fact by subtracting a $\ln \pi_{\mathcal{D}}$ term inside the exponential. For the proposed Tsallis methods, a crucial difference lies in how this exponential advantage function is modified.

Tsallis In-Sample Actor-Critic (Tsallis InAC). We follow in-sample softmax (Xiao et al., 2023) but replaces the Shannon entropy to Tsallis entropy. In order to fulfill the in-sample constraint, a step similar to

(Xiao et al., 2023) is made to extract $\pi_{\mathcal{D}}$ from the Tsallis entropy regularized policy for the actor loss:

$$\begin{aligned}
&\hat{\pi}^{\texttt{TInAC}}_{Q_\theta,\pi_{\mathcal{D}}}(a|s) \propto \mathbb{1}\{a : \pi_{\mathcal{D}}(a|s) > 0\} \cdot \exp_q\left(\frac{1}{\tau}Q_\theta(s,a)\right) \\
&= \pi_{\mathcal{D}}(a|s)\pi_{\mathcal{D}}(a|s)^{-1}\exp_q\left(\frac{1}{\tau}Q_\theta(s,a)\right) = \pi_{\mathcal{D}}(a|s)\exp_q\left(\ln_q \frac{1}{\pi_{\mathcal{D}}(a|s)}\right)\exp_q\left(\frac{1}{\tau}Q_\theta(s,a)\right) \\
&= \pi_{\mathcal{D}}(a|s)\left(\exp_q\left(\frac{1}{\tau}Q_\theta(s,a) + \ln_q \frac{1}{\pi_{\mathcal{D}}(a|s)}\right)^{q-1} + (q-1)^2\frac{1}{\tau}Q_\theta(s,a)\ln_q \frac{1}{\pi_{\mathcal{D}}(a|s)}\right)^{\frac{1}{q-1}}.
\end{aligned} \tag{11}$$

In the last step we made use of $\left(\exp_q x \cdot \exp_q y\right)^{q-1} = \exp_q(x+y)^{q-1} + (q-1)^2 xy$ (Yamano, 2002). Following (Haarnoja et al., 2018), we update towards this policy by minimizing the KL divergence between $\hat{\pi}^{\texttt{TInAC}}_{Q_\theta,\pi_{\mathcal{D}}}$ and π_ϕ. When used as the first argument of KL loss, the leading $\pi_{\mathcal{D}}$ allows us to compute the loss using only actions from the dataset:

$$\begin{aligned}
&D_{KL}\left(\hat{\pi}^{\texttt{TInAC}}_{Q_\theta,\pi_{\mathcal{D}}}(\cdot|s)\,||\,\pi_\phi(\cdot|s)\right) = \mathbb{E}_{a\sim\hat{\pi}^{\texttt{TInAC}}_{Q_\theta,\pi_{\mathcal{D}}}(\cdot|s)}\left[\ln \hat{\pi}^{\texttt{TInAC}}_{Q_\theta,\pi_{\mathcal{D}}}(a|s) - \ln \pi_\phi(a|s)\right] \\
&= \mathbb{E}_{a\sim\pi_{\mathcal{D}}(\cdot|s)}\left[\left(\exp_q\left(\frac{Q_\theta(s,a)}{\tau} + \ln_q \frac{1}{\pi_\omega(a|s)}\right)^{q-1} + (q-1)^2\frac{Q_\theta(s,a)\ln_q \frac{1}{\pi_\omega(a|s)}}{\tau}\right)^{\frac{1}{q-1}}\left(\ln \hat{\pi}^{\texttt{TInAC}}_{Q_\theta,\pi_{\mathcal{D}}}(a|s) - \ln \pi_\phi(a|s)\right)\right].
\end{aligned}$$

Notice the term $\ln \hat{\pi}^{\texttt{TInAC}}_{Q_\theta,\pi_{\mathcal{D}}}(a|s)$ does not depend on π_ϕ and hence can be removed from the actor loss. We observed that for Tsallis entropy regularization, adding normalization tends to underperform, therefore we propose to remove the normalization. We write out our losses for $q = 2$:

$$\begin{aligned}
&\mathcal{L}^{\texttt{TInAC}}_{\text{actor}}(\phi) = -\mathbb{E}_{s,a\sim\mathcal{D}}\left[\left(\exp_2\left(\frac{Q_\theta(s,a)}{\tau} + \ln_2 \frac{1}{\pi_\omega(a|s)}\right) + \frac{Q_\theta(s,a)\ln_2 \frac{1}{\pi_\omega(a|s)}}{\tau}\right)\ln \pi_\phi(a|s)\right], \\
&\mathcal{L}^{\texttt{TInAC}}_{\text{critic}}(\theta) = \mathbb{E}_{s,a,r,s'\sim\mathcal{D}}\left[\left(r + \gamma V_\zeta(s') - Q_\theta(s,a)\right)^2\right], \\
&\mathcal{L}^{\texttt{TInAC}}_{\text{baseline}}(\zeta) = \mathbb{E}_{s\sim\mathcal{D},a\sim\pi_\phi(\cdot|s)}\left[\left(V_\zeta(s) - \left(Q_\theta(s,a) - \tau\ln_2 \pi_\phi(a|s)\right)\right)^2\right].
\end{aligned} \tag{12}$$

The term $\frac{1}{\pi_\omega(a|s)}$ in $\mathcal{L}^{\texttt{TInAC}}_{\text{actor}}(\phi)$ is likely to cause numerical issues. To avoid it, we clip the range of π_ω by $\max\{\epsilon, \pi_\omega(a|s)\}$, with $\epsilon = 10^{-8}$. However, since $\ln_q x$ is proportional to the q-th power, we are unable to sweep over larger q due to numerical problems even with clipping. Therefore, in the experiments we sweep different q for the Tsallis KL implementation only.

Remark. Let us focus on $\mathcal{L}^{\texttt{TInAC}}_{\text{actor}}(\phi)$ and define the term in the bracket before $\ln \pi_\phi$ as C:

$$C := \exp_2\left(\frac{Q_\theta(s,a)}{\tau} + \ln_2 \frac{1}{\pi_\omega(a|s)}\right) + \frac{Q_\theta(s,a)\ln_2 \frac{1}{\pi_\omega(a|s)}}{\tau},$$

then $\mathcal{L}^{\texttt{TInAC}}_{\text{actor}}(\phi) = -\mathbb{E}_{s,a\sim\mathcal{D}}\left[C \ln \pi_\phi(a|s)\right]$. Notice that $\exp_2\left(\frac{Q_\theta(s,a)}{\tau} + \ln_2 \frac{1}{\pi_\omega(a|s)}\right)$ can be written as

$$\mathbb{1}\left\{1 + \frac{Q_\theta(s,a)}{\tau} + \ln_2 \frac{1}{\pi_\omega(a|s)} > 0\right\} \cdot \left(1 + \frac{Q_\theta(s,a)}{\tau} + \ln_2 \frac{1}{\pi_\omega(a|s)}\right),$$

therefore, it must be non-negative and the sign of C depends on $\frac{Q_\theta(s,a)\ln_2 \frac{1}{\pi_\omega(a|s)}}{\tau}$. However, since $\ln_2 \frac{1}{\pi_\omega(a|s)} > 0$, the sign actually depends on $Q_\theta(s,a)$ only. If this term is negative and $\left|\frac{Q_\theta(s,a)\ln_2 \frac{1}{\pi_\omega(a|s)}}{\tau}\right| > \exp_2\left(\frac{Q_\theta(s,a)}{\tau} + \ln_2 \frac{1}{\pi_\omega(a|s)}\right)$, then we will be minimizing the loss $\mathbb{E}_{s,a\sim\mathcal{D}}\left[|C| \ln \pi_\phi(a|s)\right]$, which corresponds to thinking an action a is bad (since $Q_\theta(s,a)$ is highly negative) and explicitly minimizing its log-likelihood; otherwise, we will be maximizing its likelihood. By contrast, all exponential-based methods (exponential advantage weighted regression algorithms like AWAC) do not minimize likelihood for actions in the dataset explicitly, since exp is always positive. Moreover, this suggests that the behavior of Tsallis InAC may be more extreme with larger q, since by equation 11, $(q-1)^2\frac{Q_\theta(s,a)\ln_2 \frac{1}{\pi_\omega(a|s)}}{\tau}$ will have a large negative value when q is large even with slightly negative $Q_\theta(s,a)$.

Tsallis AWAC (TAWAC). This scheme regularizes only the policy improvement step. Given $Q_\theta(s,a)$, we write the policy as $\hat{\pi}^{\texttt{TAWAC}}_{Q_\theta,\pi_\mathcal{D}}(a|s) \propto \pi_\mathcal{D}(a|s)\exp_q\left(\frac{1}{\tau}Q_\theta(s,a)\right)$. Repeating the KL loss step, we have:

$$\begin{aligned} D_{KL}\left(\hat{\pi}^{\texttt{TAWAC}}_{Q_\theta,\pi_\mathcal{D}}(\cdot|s)\,||\,\pi_\phi(\cdot|s)\right) &= \mathbb{E}_{a\sim\hat{\pi}^{\texttt{TAWAC}}_{Q_\theta,\pi_\mathcal{D}}(\cdot|s)}\left[\ln\hat{\pi}^{\texttt{TAWAC}}_{Q_\theta,\pi_\mathcal{D}}(a|s) - \ln\pi_\phi(a|s)\right] \\ &= \mathbb{E}_{a\sim\pi_\mathcal{D}}\left[\exp_q\left(\frac{Q_\theta(s,a)}{\tau} - \psi\left(\frac{Q_\theta(s,\cdot)}{\tau}\right)\right)\left(\ln\hat{\pi}^{\texttt{TAWAC}}_{Q_\theta,\pi_\mathcal{D}}(a|s) - \ln\pi_\phi(a|s)\right)\right]. \end{aligned}$$

The normalization function ψ poses a challenge to continuous-action problems. Inspired by (Xiao et al., 2023), we propose to replace it with the state value function V_ζ. We write the loss functions as:

$$\begin{aligned} &\mathcal{L}^{\texttt{TAWAC}}_{\text{actor}}(\phi) = -\mathbb{E}_{(s,a)\sim\mathcal{D}}\left[\exp_q\left(\frac{Q_\theta(s,a)-V_\zeta(s)}{\tau}\right)\ln\pi_\phi(a|s)\right], \\ &\mathcal{L}^{\texttt{TAWAC}}_{\text{critic}}(\theta) = \mathbb{E}_{(s,a,r,s')\sim\mathcal{D}}\left[\left(r+\gamma V_\zeta(s') - Q_\theta(s,a)\right)^2\right], \\ &\mathcal{L}^{\texttt{TAWAC}}_{\text{baseline}}(\zeta) = \mathbb{E}_{s\sim\mathcal{D},a\sim\pi_\phi(\cdot|s)}\left[\left(V_\zeta(s) - Q_\theta(s,a)\right)^2\right]. \end{aligned} \tag{13}$$

Remark. Comparing $\mathcal{L}^{\texttt{TAWAC}}_{\text{actor}}(\phi)$ and $\mathcal{L}^{\texttt{AWAC}}_{\text{actor}}(\phi)$, it is clear that even though Tsallis AWAC is only different to AWAC in the q-exponential function, it brings sparsity to the policy. To see this, we can write out $\mathcal{L}^{\texttt{TAWAC}}_{\text{actor}}(\phi)$ as the following:

$$-\mathbb{E}_{(s,a)\sim\mathcal{D}}\left[\mathbb{1}\left\{\left[1+(q-1)\left(\frac{Q_\theta(s,a)-V_\zeta}{\tau}\right)\right]^{\frac{1}{q-1}} > 0\right\}\cdot\left(\left[1+(q-1)\left(\frac{Q_\theta(s,a)-V_\zeta(s)}{\tau}\right)\right]^{\frac{1}{q-1}}\right)\ln\pi_\phi(a|s)\right].$$

Since the root does not affect the sign, it can be seen that actions with values $\frac{Q_\theta(s,a)-V_\zeta(s)}{\tau} < -\frac{1}{q-1}$ will be truncated: the actor loss for these actions becomes zero and TAWAC does not maximize its likelihood. Another interesting observation is that by setting $q=2$, $\mathcal{L}^{\texttt{TAWAC}}_{\text{actor}}(\phi)$ **recovers the sparse Q-learning objective in** (Xu et al., 2023), which was derived from the α-divergence perspective. However, our loss is more general since for all $q>0$ $\mathcal{L}^{\texttt{TAWAC}}_{\text{actor}}(\phi)$ has a truncation effect. As explained in the remark for Tsallis InAC, all exponential advantage weighted methods maximize likelihood for action in the dataset, but with different degrees controlled by $\exp\left(\frac{Q_\theta(s,a)-V_\zeta(s)}{\tau}\right)$. It is worth noting that InAC compensates for this fact by subtracting a $\ln\pi_\mathcal{D}$ term inside the exponential to behave more conservatively and achieves significantly better results on non-expert datasets. The difference between Tsallis AWAC and Tsallis InAC lies in that: Tsallis AWAC does not minimize action probability, but instead only sets the loss of bad actions to zero, which suggests that Tsallis AWAC may be less drastic and more amenable to non-expert datasets.

6 Experiments

Below, we evaluate the Tsallis InAC and Tsallis AWAC agents against several offline reinforcement learning baselines. The goal of the experiments is to find when are Tsallis InAC and Tsallis AWAC agents best applied in offline RL, as well as gain insight into the effect of the τ and q parameters (especially as we enter regimes where there aren't closed form expressions for the policies). Finally, we evaluate if the upper bound holds in the continuous action setting.

Domain Details. We evaluate Tsallis InAC and Tsallis AWAC against a number of baseline algorithms on standard benchmark D4RL environments (Fu et al., 2020). Specifically, we use three datasets from the Mujoco suite in D4RL. Trajectories in the offline datasets are collected by a SAC agent. The naming expert/medium expert/medium replay reflects the level of the trained agent used to collect the transitions. The expert dataset contains trajectories collected by a fully trained SAC agent; the medium dataset contains transitions of a SAC agent trained halfway; medium-expert combines the trajectories of the expert and the medium.

The agents train from the specified datasets, and are evaluated every 10k steps on the corresponding Mujoco environments. Normalized scores are calculated according to the min and max values provided as a part of the benchmarks (Fu et al., 2020). Finally, results are over 5 runs unless otherwise specified.

Name	Value
Number of steps	1000000
Logging interval	10000
Hidden Units	256
Batch Size	256
Target Network Update Rate	1
Polyak Constant	0.995
Discount (γ)	0.99
Learning Rate	swept
Regularization coefficient (τ)	swept

Name	Swetp Values
Learning Rate	[0.00003, 0.0001, 0.0003, 0.001]
τ	[0.01, 0.1, 0.33, 0.5, 1.0]

Figure 3: (Left) Shared hyperparameters. (Right) Swept hyperparameters.

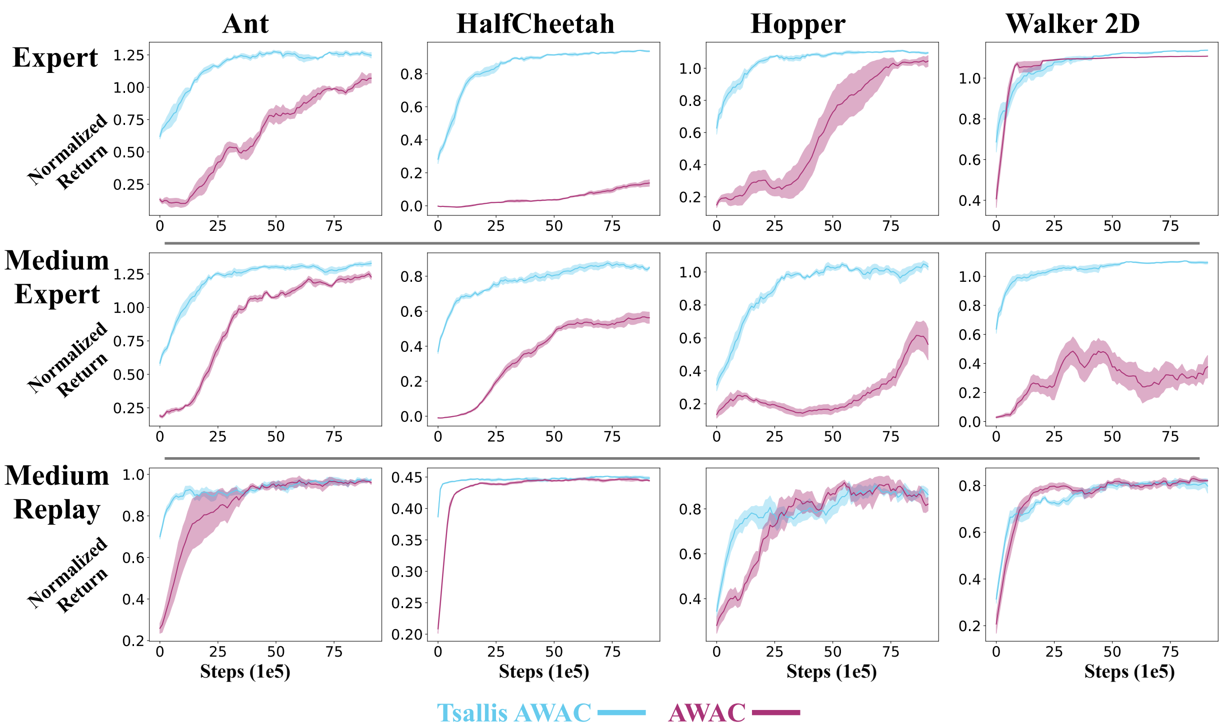

Figure 4: Comparison between Tsallis AWAC and AWAC. Normalized policy evaluation scores on Mujoco Medium Replay dataset over one million steps. Results are the average over 5 runs with ribbon denoting the standard error. y-axis: scores; x-axis: number of iterations.

Baseline algorithms. We compare against a number of baseline algorithms: InAC: in-sample softmax actor-critic (Xiao et al., 2023). It is worth noting InAC can be seen as the specical case $q = 1$ of Tsallis InAC. TD3 + BC (Fujimoto & Gu, 2021) augments the policy improvement step of TD3 with an additional behavior cloning (BC) term $(\pi(s) - a)^2$ as indicated by (Xiao et al., 2023). This term can be seen as a KL regularization under the Gaussian policy parametrization. IQL: implicit Q-learning (Kostrikov et al., 2022) employed in-sample hard max Eq. (2). AWAC: advantage-weighted actor-critic (Nair et al., 2021), it can be as a special case of Tsallis AWAC when $q = 1$. For all the baseline algorithms except InAC, we followed the published settings of the baseline agents. We fine-tune Tsallis InAC, Tsallis AWAC and InAC since they share a same set of hyperparameters. All the algorithms used a shared set of hyperparameters found in Table 3.

A grid search was done for Tsallis InAC, Tsallis AWAC, InAC according to the same protocol as (Xiao et al., 2023). In addition, we also added a larger learning rate (0.001), which seemed to improve InAC, Tsallis AWAC, and Tsallis InAC slightly on some domains. Our other baselines use performance data shared by (Xiao et al., 2023). A full list of the swept hyperparameters can be seen in the right hand side of Figure

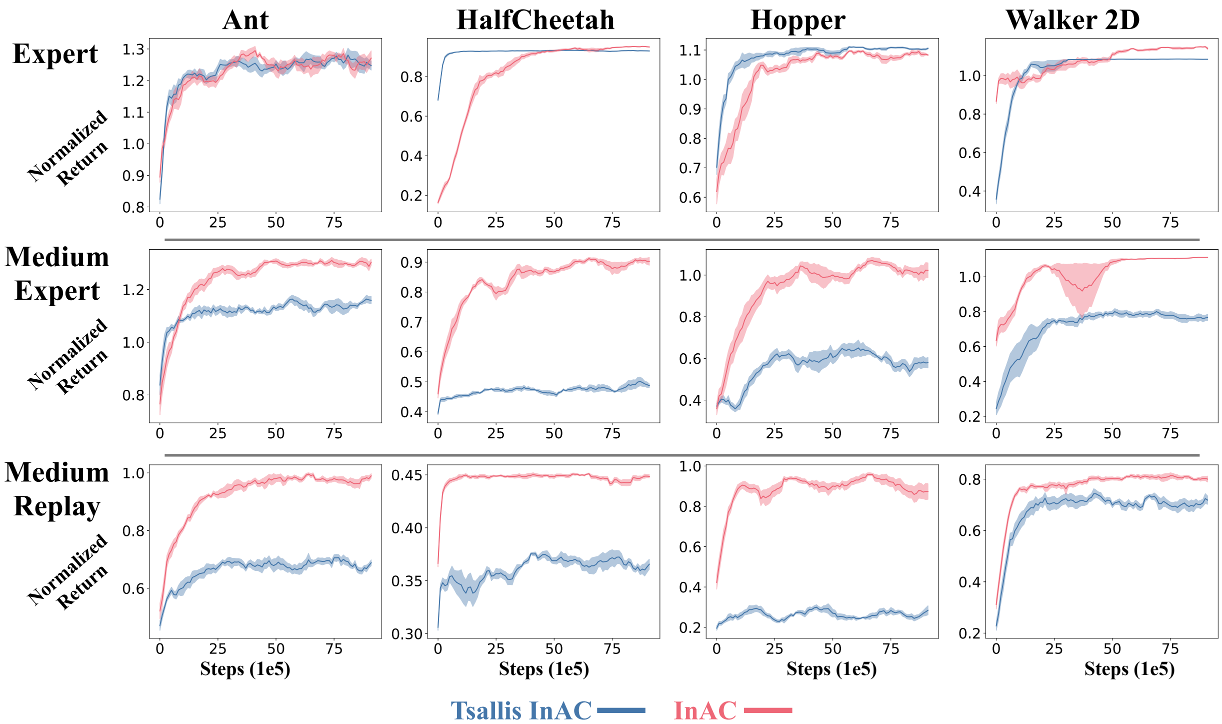

Figure 5: Comparison between Tsallis InAC and InAC. Normalized policy evaluation scores on Mujoco Expert and Medium Expert datasets over one million steps. Results are the average over 5 runs with ribbon denoting the standard error. y-axis: scores; x-axis: number of iterations.

3. In the grid search, we used the final 50% of evaluation tests of the normalized return to select the best hyperparameter shown. All hyperparameter settings were evaluated across 5 independent runs.

6.1 Comparison Against the Baselines

Since Tsallis AWAC (resp. Tsallis InAC) generalizes AWAC (resp. InAC), we first compare the generalizations in Figure 4 and Figure 5, and leave the comparison against all baselines to Figure 6.

Tsallis AWAC against AWAC. From Figure 4 it is visible Tsallis AWAC outperforms AWAC by a large margin on Expert and Medium-Expert datasets and the performance remains stable across all datasets. The poor performance of AWAC has been discussed a lot in the literature (Xiao et al., 2023; Xu et al., 2023): the exponential term in $\mathcal{L}^{\texttt{AWAC}}_{\text{actor}}$ can cause unstable gradients and is also more vulnerable to hyperparameter choices. On the other hand, the favorable performance of Tsallis AWAC confirms the discussion given by the remark after equation 10: the sparsity-inducing learning objective $\mathcal{L}^{\texttt{TAWAC}}_{\text{actor}}(\phi)$ is more robust against both transition/suboptimality noises and numerical errors than the exponential function (Xu et al., 2023).

Tsallis InAC against InAC. From Figure 5 it is clear that Tsallis InAC competes favorably against InAC only on the Expert dataset, and degrades significantly with the increase of non-expert trajectories in the dataset. In fact, Tsallis InAC behaves like a behavior cloning method and learns very fast on Expert `HalfCheetah` and `Hopper`. As analyzed in the remark after equation 12, the difference between InAC and AWAC policies lies in the term $-\ln \pi_{\mathcal{D}}$ in the policy $\pi^{\texttt{InAC}}(a|s) \propto \pi_{\mathcal{D}}(a|s) \exp\left(\frac{Q_\theta(s,a)}{\tau} - \ln \pi_{\mathcal{D}}(a|s)\right)$. This term can correct the bias introduced by suboptimal data distributions: a poor action with high $\pi_{\mathcal{D}}(a|s)$ does not necessarily lead to high $\pi^{\texttt{InAC}}(a|s)$ since it is weighted down inside the exponential by $-\ln \pi_{\mathcal{D}}(a|s)$. While Tsallis InAC follows this design choice, in general $\ln_q \frac{1}{\pi_{\mathcal{D}}(a|s)} \gg \ln \frac{1}{\pi_{\mathcal{D}}(a|s)}$. In fact, since $\ln_q \frac{1}{\pi_{\mathcal{D}}(a|s)}$ is proportional to the q-th power of $\frac{1}{\pi_{\mathcal{D}}(a|s)} \geq 1$, this term is likely to dominate the entire $\mathcal{L}^{\texttt{TInAC}}_{\text{actor}}$, suggesting Tsallis InAC is very sensitive to the level of behavior policy like a BC method.

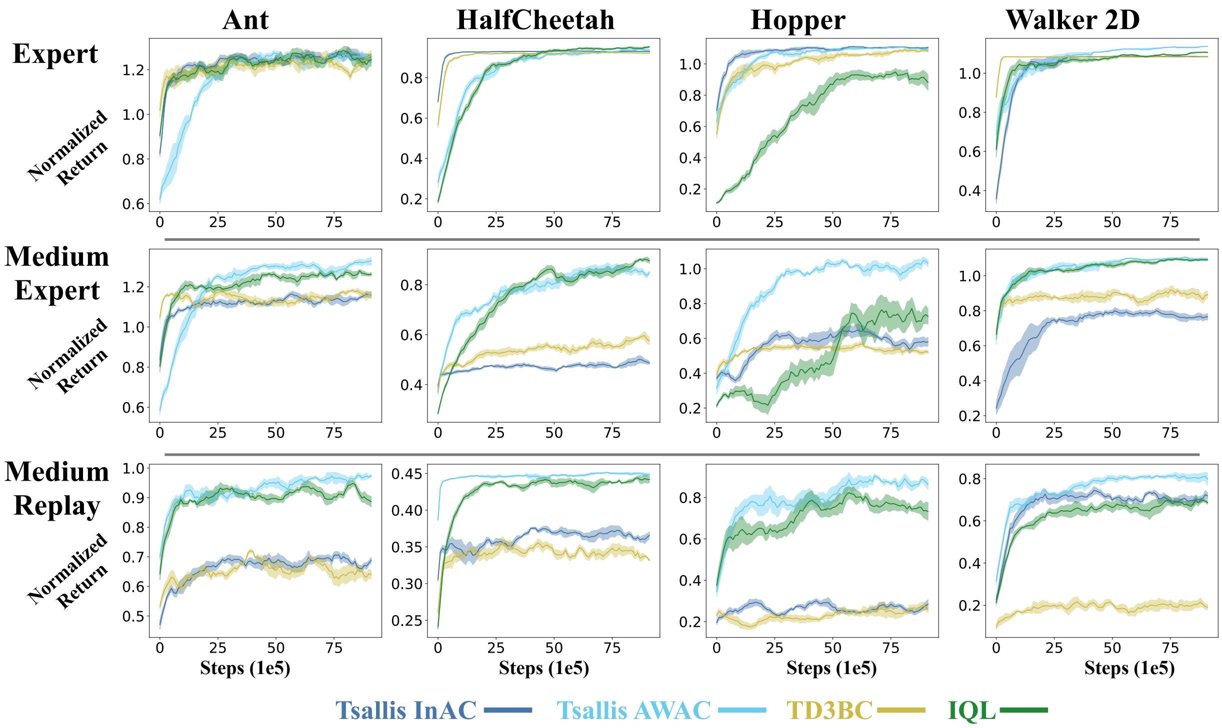

Figure 6: Additional comparison against TD3BC and Implicit Q-learning. Normalized policy evaluation scores over one million steps and are the average over 5 runs, with ribbon denoting the standard error. y-axis: scores; x-axis: number of iterations.

Against all baselines. From Figure 6 it is visible that Tsallis InAC and TD3BC are the best performers on expert level datasets—in terms of convergence speed and the final score. By the analysis for Tsallis InAC, the term $\ln_q \frac{1}{\pi_{\mathcal{D}}}$ may dominate the actor and render Tsallis InAC behave like a BC method even without an explicit BC term. This is especially apparent on `HalfCheetah-expert` and `Hopper-expert` where Tsallis InAC learned even faster than TD3BC. Both TD3BC and Tsallis InAC drastically degrade for Medium Expert and Medium Replay datasets. Tsallis AWAC outperforms IQL on almost all non-expert datasets, which matches the observation of (Xiao et al., 2023): IQL may perform poorly when the data distribution is skewed towards suboptimal actions in some states, pulling down the expectile regression targets.

6.2 The Importance of q and τ

Entropic index q and regularization coefficient τ determine the degree of sparsity. Therefore, they are related to performance and the q-conditional distance to the behavior policy as shown by equation 9. We evaluate different (q, τ) combinations of Tsallis AWAC in Figure 7. Performance is evaluated after 5×10^5 steps of training. It is visible that for both environments larger q tend to learn quicker and are relatively insensitive to τ; while smaller q such as $q = 2$ prefers larger τ. Consider fixed τ, by definition for larger q, the Tsallis AWAC policy $\pi_{\mathcal{D}}(a|s) \exp_q \left(\frac{Q_\theta(s,a) - V_\zeta(s)}{\tau}\right) = \pi_{\mathcal{D}}(a|s) \left[1 + (q-1)\frac{Q_\theta(s,a) - V_\zeta(s)}{\tau}\right]_+^{\frac{1}{q-1}}$ shrinks the gap $Q_\theta(s,a) - V_\zeta(s)$ (imagine softmax with a large τ) and the truncation threshold $-\frac{1}{q-1}$. Comparing to $q = 2$ where $\exp_q$ is linear in its argument, higher q truncates more, especially for actions that $Q_\theta(s,a) - V_\zeta(s) < 0$. Let us in turn consider fixed q. Larger τ results in larger set of allowable actions $K_{t,q}$. Similar to the softmax case (Haarnoja et al., 2017), it is advocated that choosing a larger τ at the beginning and gradually annealed towards zero. Therefore, when the dataset contains sufficient (near) expert trajectories, choosing a large (q, τ) may accelerate learning at the early stage. In Figure 9 it is visible that on the expert level environments all Tsallis agents seem to be robust and converged to policies of similar level.

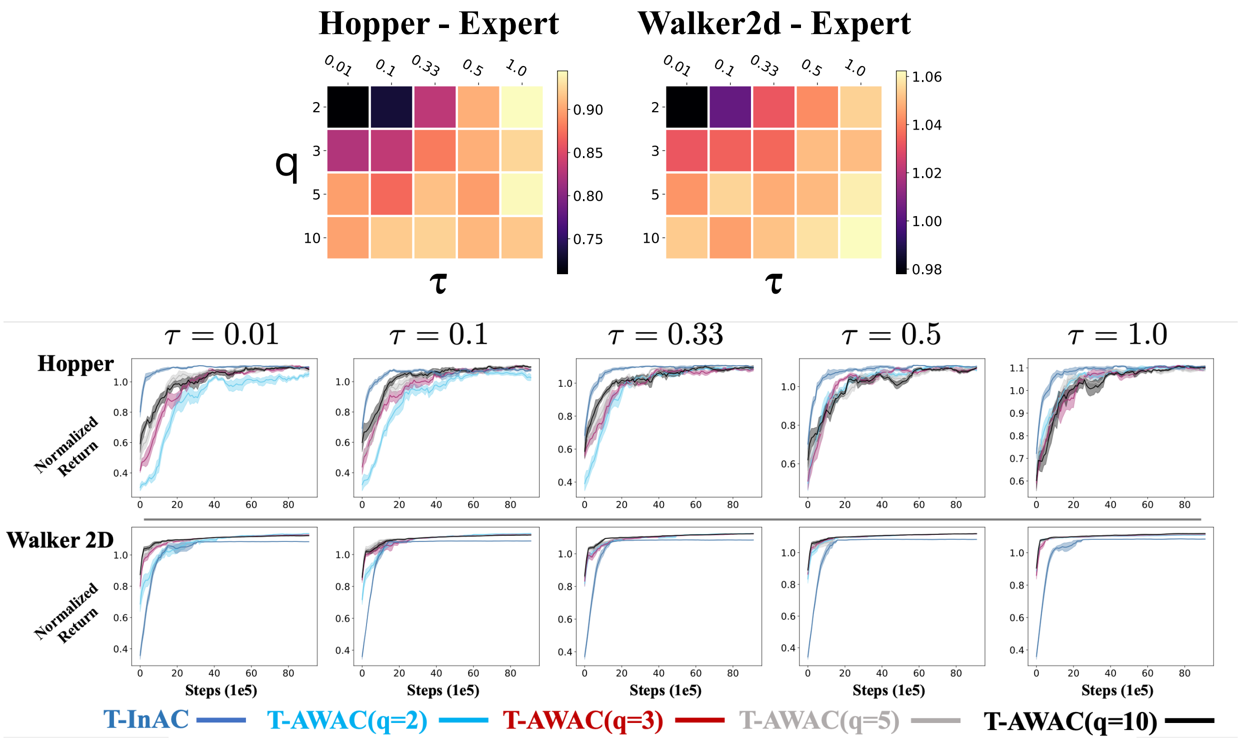

Figure 7: (Upper) Heatmap of average Normalized score of Tsallis AWAC with different q, evaluated over first 500K steps of training. x-axis: τ, y-axis: q. (Lower) Normalized score learning curves of Tsallis AWAC with different q.

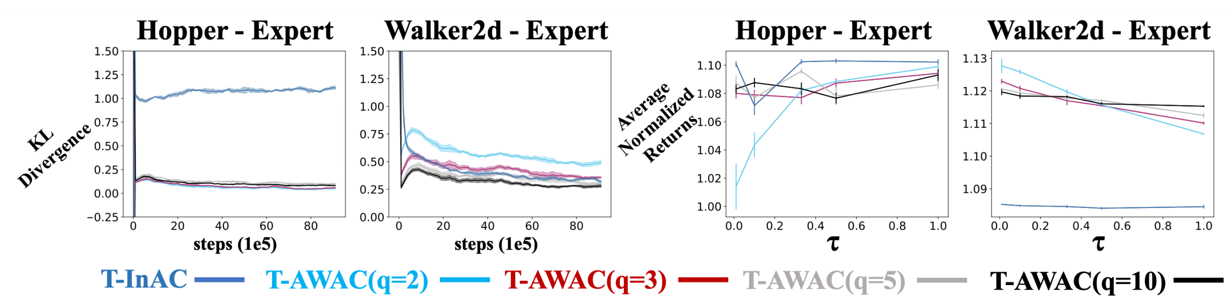

Figure 8: $D_{KL}(\pi_\omega(\cdot|s)\,||\,\pi_\phi(\cdot|s))$ throughout training for different q with best τ. Averaged over 5 runs with ribbons denoting standard error.

Figure 9: Sensitivity of Tsallis AWAC to τ. Reported values are average normalized scores over the final 500k steps of training. Averaged over 5 runs with standard error bars.

6.3 KL divergence to the Behavior Policy

Though higher similarity to the behavior policy does not necessarily imply better performance, we quantitatively evaluate it in support of our theory equation 9. We plot in Figure 8 $D_{KL}(\pi_\omega(\cdot|s)\,||\,\pi_\phi(\cdot|s))$ for Tsallis AWAC with different q and Tsallis InAC throughout training. Since the KL divergence can be written as $\mathbb{E}_{a\sim\pi_\omega(\cdot|s)}\left[\ln \pi_\omega(a|s) - \ln \pi_\phi(a|s)\right]$, and π_ω is imitating $\pi_\mathcal{D}$, we replace the sampling part to $a \sim \pi_\mathcal{D}(\cdot|s)$ to allow for random sampling actions from the dataset to compute the log-policy difference. Though in the continuous action setting $K_{t,q}$ is uncountable and the upper bound on KL is no longer valid, we can empirically investigate it. On `Hopper-expert` Tsallis AWAC for all q converged to 0, and Tsallis InAC remains stable around 1; different q did not seem to affect the KL divergence. On the other hand, on `Walker2d-expert` a clear stratification was displayed: larger q resulted in lower divergence value. Theoretically, the RHS of Eq. (9) approaches $|K_{t,q}(s)| \left(\pi_t(a|s) - \frac{q-3}{q-1}\right)$ as q gets larger, which is indeed tighter than $q = 2$. Therefore, the upperbound may still be in effect even in the continuous action setting.

7 Discussion and Conclusion

Tsallis regularizers have been less popular in RL due to the action truncation of its regularized optimal policy. The truncation often leads to underperformance in online problems resulting from limited exploration. This paper is the first work that introduces Tsallis regularizers to offline RL, where no exploration is required. Tsallis regularizers bring close two popular offline RL methods avoiding producing erronesouly optimistic out-of-distribution actions: divergence regularization and in-sample constraint. Tsallis regularization induces sparsemax policies that truncate actions with low action values, which we exploited to link to the fact that offline datasets contain only a subset of actions: we assumed the dataset was generated by a sparsemax behavior policy. As such, the in-sample constraint can be replaced by the truncation criterion. We showed two interesting facts given the assumption: (1) sparsemax policies interpolate hardmax and softmax, and when action values are fixed, consecutive sparsemax policies are within or equal to the support of the last sparsemax policy. (2) the KL divergence between a two sparsemax policies have a sparsity-conditional (q-conditional) upper bound.

We proposed two actor-critic algorithms, Tsallis In-sample Actor-Critic (Tsallis InAC) based on Tsallis entropy regularization, and Tsallis Advantage-Weighted Actor-Critic (Tsallis AWAC), that respectively generalize InAC and AWAC, two important offline RL algorithms based on Shannon entropy and KL divergence to the domain of q. This generalization is non-trivial since sparsity is introduced to the actor: unlike InAC and AWAC that only consider maximizing log-likelihood of actions in the dataset, Tsallis InAC can minimize likelihood for bad actions, while Tsallis AWAC can set the loss for bad actions to zero. Sparsity has been very recently investigated to be crucial for superior performance. We evaluated Tsallis InAC and Tsallis AWAC on the standard D4RL benchmark problem. We found that Tsallis InAC was sensitive to expert datasets on which it was among the best performer but quickly degraded on non-expert datasets. We attributed the Tsallis InAC behavior resemblant to TD3BC to the q-logarithm policy term. By contrast, Tsallis AWAC was more stable and outperformed AWAC by a large margin on almost every dataset, thanks to the sparsity introduced by the q-exponential policy.

Several interesting future directions concerning Tsallis regularization for offline RL exist. Theoretically, a probabilistic upper bound for the KL divergence of sparsemax policies and guarantees of policy improvement should be derived to deeply investigate the benefit of sparsity. Practically, it is important to improve Tsallis InAC on non-expert datasets can be potentially achieved by referring to Tsallis statistics literature to transform q-exponential. Moreover, examining the pattern of environment/dataset-specific optimal entropic index is another interesting topic.

References

OpenAI: Marcin Andrychowicz, Bowen Baker, Maciek Chociej, and Others. Learning dexterous in-hand manipulation. *The International Journal of Robotics Research*, 39(1):3–20, 2020.

Kavosh Asadi and Michael L. Littman. An alternative softmax operator for reinforcement learning. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, volume 70 of *Proceedings of Machine Learning Research*, pp. 243–252, International Convention Centre, Sydney, Australia, 06–11 Aug 2017. PMLR.

Mohammad Gheshlaghi Azar, Vicenç Gómez, and Hilbert J Kappen. Dynamic policy programming. *Journal of Machine Learning Research*, 13(1):3207–3245, 2012.

Boris Belousov and Jan Peters. Entropic regularization of markov decision processes. *Entropy*, 21(7), 2019.

Mathieu Blondel, AndrÃ© F.T. Martins, and Vlad Niculae. Learning with fenchel-young losses. *Journal of Machine Learning Research*, 21(35):1–69, 2020.

David Brandfonbrener, William F Whitney, Rajesh Ranganath, and Joan Bruna. Offline RL without off-policy evaluation. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, 2021.

Yinlam Chow, Ofir Nachum, and Mohammad Ghavamzadeh. Path consistency learning in Tsallis entropy regularized MDPs. In *International Conference on Machine Learning*, pp. 979–988, 2018.

Robert Dadashi, Shideh Rezaeifar, Nino Vieillard, Léonard Hussenot, Olivier Pietquin, and Matthieu Geist. Offline reinforcement learning with pseudometric learning. In *Proceedings of the 38th International Conference on Machine Learning*, pp. 2307–2318, 2021.

Nan Ding and S.v.n. Vishwanathan. t-logistic regression. In *Advances in Neural Information Processing Systems*, volume 23, 2010.

Justin Fu, Aviral Kumar, Ofir Nachum, George Tucker, and Sergey Levine. D4rl: Datasets for deep data-driven reinforcement learning, 2020.

Scott Fujimoto and Shixiang Shane Gu. A minimalist approach to offline reinforcement learning. In *Thirty-Fifth Conference on Neural Information Processing Systems*, 2021.

Scott Fujimoto, David Meger, and Doina Precup. Off-policy deep reinforcement learning without exploration. In *Proceedings of the 36th International Conference on Machine Learning*, pp. 2052–2062, 2019.

S. Furuichi, K. Yanagi, and K. Kuriyama. Fundamental properties of tsallis relative entropy. *Journal of Mathematical Physics*, 45(12):4868–4877, 2004.

Divyansh Garg, Joey Hejna, Matthieu Geist, and Stefano Ermon. Extreme q-learning: Maxent RL without entropy. In *The Eleventh International Conference on Learning Representations*, 2023.

Matthieu Geist, Bruno Scherrer, and Olivier Pietquin. A theory of regularized Markov decission processes. In *36th International Conference on Machine Learning*, volume 97, pp. 2160–2169, 2019.

Seyed Kamyar Seyed Ghasemipour, Dale Schuurmans, and Shixiang Shane Gu. Emaq: Expected-max q-learning operator for simple yet effective offline and online rl. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139, pp. 3682–3691, 2021.

Caglar Gulcehre, Sergio Gómez Colmenarejo, Ziyu Wang, Jakub Sygnowski, Thomas Paine, Konrad Zolna, Yutian Chen, Matthew Hoffman, Razvan Pascanu, and Nando de Freitas. Regularized behavior value estimation, 2021.

Tuomas Haarnoja, Haoran Tang, Pieter Abbeel, and Sergey Levine. Reinforcement learning with deep energy-based policies. In *34th International Conference on Machine Learning*, pp. 1352–1361, 2017.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *Proceedings of the 35th International Conference on Machine Learning*, pp. 1861–1870, 2018.

Natasha Jaques, Asma Ghandeharioun, Judy Hanwen Shen, Craig Ferguson, Agata Lapedriza, Noah Jones, Shixiang Gu, and Rosalind Picard. Way off-policy batch deep reinforcement learning of human preferences in dialog, 2020.

Liyiming Ke, Sanjiban Choudhury, Matt Barnes, Wen Sun, Gilwoo Lee, and Siddhartha Srinivasa. Imitation learning as f-divergence minimization. *arXiv*, 2019.

Ilya Kostrikov, Rob Fergus, Jonathan Tompson, and Ofir Nachum. Offline reinforcement learning with fisher divergence critic regularization. In *Proceedings of the 38th International Conference on Machine Learning*, pp. 5774–5783, 2021.

Ilya Kostrikov, Ashvin Nair, and Sergey Levine. Offline reinforcement learning with implicit q-learning. In *International Conference on Learning Representations*, 2022.

Tadashi Kozuno, Eiji Uchibe, and Kenji Doya. Theoretical analysis of efficiency and robustness of softmax and gap-increasing operators in reinforcement learning. In *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, volume 89, pp. 2995–3003, 2019.

Aviral Kumar, Justin Fu, George Tucker, and Sergey Levine. Stabilizing off-policy q-learning via bootstrapping error reduction. In *Advances on Neural Information Processing Systems*, 2019.

Aviral Kumar, Aurick Zhou, George Tucker, and Sergey Levine. Conservative q-learning for offline reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 33, pp. 1179–1191, 2020.

Kyungjae Lee, Sungjoon Choi, and Songhwai Oh. Sparse markov decision processes with causal sparse tsallis entropy regularization for reinforcement learning. *IEEE Robotics and Automation Letters*, 3:1466–1473, 2018.

Kyungjae Lee, Sungyub Kim, Sungbin Lim, Sungjoon Choi, Mineui Hong, Jae In Kim, Yong-Lae Park, and Songhwai Oh. Generalized tsallis entropy reinforcement learning and its application to soft mobile robots. In *Robotics: Science and Systems XVI*, pp. 1–10, 2020.

Yuhan Li, Wenzhuo Zhou, and Ruoqing Zhu. Quasi-optimal reinforcement learning with continuous actions. In *The Eleventh International Conference on Learning Representations*, 2023.

André F. T. Martins and Ramón F. Astudillo. From softmax to sparsemax: A sparse model of attention and multi-label classification. In *Proceedings of the 33rd International Conference on Machine Learning*, pp. 1614–1623, 2016.

David McAllester and Karl Stratos. Formal limitations on the measurement of mutual information. In *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108, pp. 875–884, 2020.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Joel Veness, Marc G. Bellemare, and Others. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015.

Ashvin Nair, Murtaza Dalal, Abhishek Gupta, and Sergey Levine. {AWAC}: Accelerating online reinforcement learning with offline datasets, 2021.

Takayuki Osa, Akinobu Hayashi, Pranav Deo, Naoki Morihira, and Takahide Yoshiike. Offline reinforcement learning with mixture of deterministic policies. *Transactions on Machine Learning Research*, 2023.

Aldo Pacchiano, Jonathan N Lee, Peter Bartlett, and Ofir Nachum. Near optimal policy optimization via reps. In *Advances in Neural Information Processing Systems*, volume 34, pp. 1100–1110, 2021.

Xue Bin Peng, Aviral Kumar, Grace Zhang, and Sergey Levine. Advantage weighted regression: Simple and scalable off-policy reinforcement learning, 2020.

Dean A. Pomerleau. Alvinn: An autonomous land vehicle in a neural network. In *Advances in Neural Information Processing Systems*, 1988.

Konrad Rawlik, Marc Toussaint, and Sethu Vijayakumar. On stochastic optimal control and reinforcement learning by approximate inference (extended abstract). In *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence*, IJCAI '13, pp. 3052–3056, 2013.

Tim G. J. Rudner, Cong Lu, Michael Osborne, Yarin Gal, and Yee Whye Teh. On pathologies in KL-regularized reinforcement learning from expert demonstrations. In *Advances in Neural Information Processing Systems*, 2021.

Noah Siegel, Jost Tobias Springenberg, Felix Berkenkamp, Abbas Abdolmaleki, Michael Neunert, Thomas Lampe, Roland Hafner, Nicolas Heess, and Martin Riedmiller. Keep doing what worked: Behavior modelling priors for offline reinforcement learning. In *International Conference on Learning Representations*, 2020.

David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, and Others. Mastering the game of Go without human knowledge. *Nature*, 550:354–359, 2017.

H. Suyari and M. Tsukada. Law of error in tsallis statistics. *IEEE Transactions on Information Theory*, 51 (2):753–757, 2005.

C. Tsallis. *Introduction to Nonextensive Statistical Mechanics: Approaching a Complex World.* Springer New York, 2009. ISBN 9780387853581.

Nino Vieillard, Tadashi Kozuno, Bruno Scherrer, Olivier Pietquin, Rémi Munos, and Matthieu Geist. Leverage the average: an analysis of regularization in rl. In *Advances in Neural Information Processing Systems 33*, pp. 1–12, 2020.

Ziyu Wang, Alexander Novikov, Konrad Żołna, Jost Tobias Springenberg, Scott Reed, Bobak Shahriari, Noah Siegel, Josh Merel, Caglar Gulcehre, Nicolas Heess, and Nando de Freitas. Critic regularized regression. In *34th Conference on Neural Information Processing Systems*, 2020.

Jialong Wu, Haixu Wu, Zihan Qiu, Jianmin Wang, and Mingsheng Long. Supported policy optimization for offline reinforcement learning. In *Advances in Neural Information Processing Systems*, 2022.

Yifan Wu, George Tucker, and Ofir Nachum. Behavior regularized offline reinforcement learning, 2020.

Chenjun Xiao, Han Wang, Yangchen Pan, Adam White, and Martha White. The in-sample softmax for offline reinforcement learning. In *The Eleventh International Conference on Learning Representations*, 2023.

Haoran Xu, Li Jiang, Jianxiong Li, and Xianyuan Zhan. A policy-guided imitation approach for offline reinforcement learning. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (eds.), *Advances in Neural Information Processing Systems*, 2022.

Haoran Xu, Li Jiang, Jianxiong Li, Zhuoran Yang, Zhaoran Wang, Victor Wai Kin Chan, and Xianyuan Zhan. Offline RL with no OOD actions: In-sample learning via implicit value regularization. In *The Eleventh International Conference on Learning Representations*, 2023.

Takuya Yamano. Some properties of q-logarithm and q-exponential functions in tsallis statistics. *Physica A: Statistical Mechanics and its Applications*, 305(3):486–496, 2002.

Wenhao Zhan, Shicong Cen, Baihe Huang, Yuxin Chen, Jason D. Lee, and Yuejie Chi. Policy mirror descent for regularized reinforcement learning: A generalized framework with linear convergence. *CoRR*, 2023.

Lingwei Zhu, Zheng Chen, Matthew Schlegel, and Martha White. Generalized munchausen reinforcement learning using tsallis kl divergence. In *Advances in Neural Information Processing Systems*, 2023.