Mutual Information Regularized Offline Reinforcement Learning

Xiao Ma* **Bingyi Kang*†** **Zhongwen Xu** **Min Lin** **Shuicheng Yan**
Sea AI Lab
{yusufma555, bingykang}@gmail.com

Abstract

The major challenge of offline RL is the distribution shift that appears when out-of-distribution actions are queried, which makes the policy improvement direction biased by extrapolation errors. Most existing methods address this problem by penalizing the policy or value for deviating from the behavior policy during policy improvement or evaluation. In this work, we propose a novel MISA framework to approach offline RL from the perspective of **M**utual **I**nformation between **S**tates and **A**ctions in the dataset by directly constraining the policy improvement direction. MISA constructs lower bounds of mutual information parameterized by the policy and Q-values. We show that optimizing this lower bound is equivalent to maximizing the likelihood of a one-step improved policy on the offline dataset. Hence, we constrain the policy improvement direction to lie in the data manifold. The resulting algorithm simultaneously augments the policy evaluation and improvement by adding mutual information regularizations. MISA is a general framework that unifies conservative Q-learning (CQL) and behavior regularization methods (*e.g.*, TD3+BC) as special cases. We introduce 3 different variants of MISA, and empirically demonstrate that tighter mutual information lower bound gives better offline RL performance. In addition, our extensive experiments show MISA significantly outperforms a wide range of baselines on various tasks of the D4RL benchmark, *e.g.*, achieving 742.9 total points on gym-locomotion tasks. Our code is attached and will be released upon publication.

1 Introduction

Reinforcement learning (RL) has made remarkable achievements in solving sequential decision-making problems, ranging from game playing [28, 39, 5] to robot control [26, 20, 36]. However, its success heavily relies on 1) an environment to interact with for data collection and 2) an online algorithm to improve the agent based only on its own trial-and-error experiences. These make RL algorithms incapable in real-world safety-sensitive scenarios where interactions with the environment are dangerous or prohibitively expensive, such as in autonomous driving and robot manipulation with human autonomy [27, 25]. Therefore, offline RL is proposed to study the problem of learning decision-making agents from experiences that are previously collected from other agents when interacting with the environment is costly or not allowed.

Though much demanded, extending RL algorithms to offline datasets is challenged by the distributional shift between the data-collecting policy and the learning policy. Specifically, a typical RL algorithm alternates between evaluating the Q values of a policy and improving the policy to have better cumulative returns under the current value estimation. When it comes to the offline setting, policy improvement often involves querying out-of-distribution (OOD) state-action pairs that

*equal contribution, † corresponding author.

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

have never appeared in the dataset, for which the Q values are over-estimated due to extrapolation error of neural networks. As a result, the policy improvement direction is erroneously affected, eventually leading to a catastrophic explosion of value estimations as well as policy collapse after error accumulation. Existing methods [25, 42, 14, 44] tackle this problem by either forcing the learned policy to stay close to the behavior policy [16, 43, 14] or generating low value estimations for OOD actions [29, 25, 44]. Though these methods are effective at alleviating the distributional shift problem of the learning policy, the improved policy is unconstrained and might still deviate from the data distribution. A natural question thus arises: can we directly constrain the policy improvement direction to lie in the data manifold?

In this paper, we step back and consider the offline dataset from a new perspective, *i.e.*, the **M**utual **I**nformation between **S**tates and **A**ctions (MISA). By viewing state and action as two random variables, the mutual information represents the reduction of uncertainty of actions given certain states, *a.k.a.*, information gain in information theory [31]. Therefore, mutual information is an appealing metric to sufficiently acquire knowledge from a dataset and characterize a behavior policy. We for the first time introduce it into offline RL as a regularization that directly constrains the policy improvement direction. Specifically, to allow practical optimizations of state-action mutual information estimation, we introduce the MISA lower bound of state-action pairs, which connects mutual information with RL by treating a parameterized policy as a variational distribution and the Q-values as the energy functions. We show that this lower bound can be interpreted as the likelihood of a non-parametric policy on the offline dataset, which represents the one-step improvement of the current policy based on the current value estimation. Maximizing MISA lower bound is equivalent to directly regularizing the policy improvement within the dataset manifold. However, the constructed lower bound involves integration over a self-normalized energy-based distribution, whose gradient estimation is intractable. To alleviate this dilemma, Markov Chain Monte Carlo (MCMC) estimation is adopted to produce an unbiased gradient estimation for MISA lower bound.

Theoretically, MISA is a general framework for offline RL that unifies several existing offline RL paradigms including behavior regularization and conservative learning. As examples, we show that TD3+BC [14] and CQL [25] are degenerated cases of MISA. Empirically, we verify that the tighter the mutual information lower bound, the better the final performance. We also demonstrate that MISA achieves significantly better performance on various environments of the D4RL [13] benchmark than a wide range of baselines, including CQL and IQL [24]. Additional visualizations are discussed to better understand the proposed method. Our code will be released upon publication.

2 Related Works

Offline Reinforcement Learning The most critical challenge for extending an off-policy RL algorithm to an offline setup is the distribution shift between the behavior policy, *i.e.*, the policy for data collection, and the learning policy [22]. To tackle this challenge, most of the offline RL algorithms consider a conservative learning framework. They either regularize the learning policy to stay close to the behavior policy [16, 43, 14, 38, 42, 45, 46], or force Q values to be low for OOD state-action pairs [29, 25, 44]. For example, TD3+BC [14] adds an additional behavior cloning (BC) signal along with the TD3 [15], which encourages the policy to stay in the data manifold; CQL [25], from the Q-value perspective, penalizes the OOD state-action pairs for generating high Q-value estimations and learns a lower bound of the true value function. However, their policy improvement direction is unconstrained and might deviate from the data distribution. On the other hand, SARSA-style updates [40] are considered to only query in-distribution state-action pairs [34, 24]. Nevertheless, without explicitly querying Bellman's optimality equation, they limit the policy from producing unseen actions. Our proposed MISA follows the conservative framework and directly regularizes the policy improvement direction to lie within the data manifold with mutual information, which more fully exploits the dataset information while learning a conservative policy. Different from the above discussed methods, a separate line works consider learning lower-bounded Q values by taking the minimum of multiple Q networks, e.g., SAC-n [2] uses 50 Q networks for hopper tasks, or using more powerful policy representations [41, 21]. Such methods achieve strong performance on existing benchmarks. However, it greatly increases the sample complexity and we do not directly compare with them in this work.

Mutual Information Estimation. Mutual information is a fundamental quantity in information theory, statistics, and machine learning. However, direct computation of mutual information is

intractable as it involves computing a log partition function of a high dimensional variable. Thus, how to estimate the mutual information $I(x, z)$ between random variables $\mathcal{X}$ and $\mathcal{Z}$, accurately and efficiently, is a critical issue. One straightforward lower bound for mutual information estimation is Barber-Agakov bound [3], which introduces an additional variational distribution $q(z \mid x)$ to approximate the unknown posterior $p(z \mid x)$. Instead of using an explicit "decoder" $q(z \mid x)$, we can use *unnormalized* distributions for the variational family $q(z \mid x)$ [12, 4, 33], i.e., approximate the distribution as $q(z \mid x) = \frac{p(z)e^{f(x,z)}}{\mathbb{E}_{p(z)}[e^{f(x,z)}]}$, where $f(x, z)$ is an arbitrary critic function. As an example, InfoNCE [33] has been widely used in representation learning literature [33, 18, 9]. To further improve the mutual information estimation, a combination of normalized and unnormalized variational distribution can be considered [8, 35]. Our MISA connects mutual information estimation with RL by parameterizing a tractable lower bound with a policy network as a variational distribution and the Q values as critics. In this way, MISA explicitly regularizes the policy improvement direction to lie in the data manifold and produces strong empirical performance.

3 Preliminaries

Reinforcement Learning We consider a Markov Decision Process (MDP) denoted as a tuple $\mathcal{M} = (\mathcal{S}, \mathcal{A}, p_0(s), p(s' \mid s, a), r(s, a), \gamma)$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, $p_0(s)$ is the initial state distribution, $p(s' \mid s, a)$ is the transition function, $r(s, a)$ is the reward function, and γ is the discount factor. The target of a learning agent is to find a policy $\pi^*(a \mid s)$ that maximizes the accumulative reward by interacting with the environment

$$\arg\max_{\pi} \mathbb{E}_{\pi}\left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) \mid s_0 \sim p_0(s), a_t \sim \pi(a \mid s_t)\right]. \tag{1}$$

Q-learning is a set of off-policy RL algorithms that utilize the optimal Bellman's optimality operator $\mathcal{B}^* Q(s, a) = r(s, a) + \gamma \mathbb{E}_{s' \sim p(s'|s,a)}[\max_{a'} Q(s', a')]$ to learn a Q function. Differently, Bellman's expectation operator $\mathcal{B}^\pi Q(s, a) = r(s, a) + \gamma \mathbb{E}_{s' \sim p(s'|s,a); a' \sim \pi(\cdot|s')}[Q(s', a')]$ gives an actor-critic framework that alternates between policy evaluation and policy improvement. Consider a value network $Q_\phi(s, a)$ parameterized by ϕ and a policy network $\pi_\theta(a|s)$ parameterzied by θ. Let $\mu_\pi(s)$ denote the stationary distribution induced with policy π, which is also called occupancy measure [37]. Given the current policy, the policy evaluation aims to learn a Q network that can accurately predict its values minimizing $\mathbb{E}_{\mu_{\pi_\theta}(s)\pi_\theta(a|s)}[(Q_\phi(s, a) - \mathcal{B}^{\pi_\theta} Q_\phi(s, a))^2]$. Policy improvement focuses on learning the optimal policy that maximizes $\mathbb{E}_{\mu_\pi(s)\pi(a|s)}[Q_\phi(s, a)]$. In practical implementations, the Bellman operator is replaced with its sample-based version $\hat{\mathcal{B}}$, and the expectation over $\mu_\pi(s)\pi(a|s)$ is approximated by an online replay buffer or an offline dataset $\mathcal{D}$.

Nevertheless, as it is unavoidable to query the OOD actions when performing the maximization over actions, an inaccurate over-estimation of the Q value will be selected and the error will accumulate during the Bellman's update. Conservative RL methods, in turn, aim to perform "conservative" updates of the value / policy function during optimization by constraining the updates on only the in-distribution samples, which minimizes the negative impact of OOD actions.

KL Divergence Given two probability distributions $p(x)$ and $q(x)$ on the same probability space, the KL divergence (*i.e.*, relative entropy) from q to p is given by $D_{\mathrm{KL}}(p||q) = \mathbb{E}_{p(x)}\left[\log \frac{p(x)}{q(x)}\right] \geq 0$. The minimum value is achieved when the two densities are identical. We consider two dual representations that result in tractable estimators for the KL divergence.

Lemma 3.1 (f-divergence representation [32]) *The KL divergence admits the following lower bound:*

$$D_{KL}(p||q) \geq \sup_{T \in \mathcal{F}} \mathbb{E}_{p(x)}[T(x)] - \mathbb{E}_{q(x)}[e^{T(x)-1}], \tag{2}$$

where the supremum is taken over a function family $\mathcal{F}$ satisfying the integrability constraints.

Lemma 3.2 (Donsker-Varadhan representation [30]) *The KL divergence has the lower bound:*

$$D_{KL}(p||q) \geq \sup_{T \in \mathcal{F}} \mathbb{E}_{p(x)}[T(x)] - \log(\mathbb{E}_{q(x)}[e^{T(x)}]), \tag{3}$$

where the supremum is taken over a function family $\mathcal{F}$ satisfying the integrability constraints.

The above two bounds are tight for large families $\mathcal{F}$.

4 Mutual Information Regularized Offline RL

We propose to tackle the offline RL problem from the perspective of mutual information estimation and develop a novel framework (MISA) by estimating the **M**utual **I**nformation between **S**tates and **A**ctions of a given offline dataset. MISA is a general framework that unifies multiple existing offline RL algorithms as special cases, including behavior cloning, TD3+BC [14], and CQL [25].

4.1 Mutual Information Regularization

Consider the state S and action A as two random variables. Let $p_{(S,A)}(s,a)$ denote the joint distribution of state-action pairs, and $p_S(s)$, $p_A(a)$ be the marginal distributions. The subscripts are omitted in the following for simplicity. The mutual information between S and A is defined with:

$$I(S;A) = \mathbb{E}_{p(s,a)}\left[\log \frac{p(s,a)}{p(s)p(a)}\right] = \mathbb{E}_{p(s,a)}\left[\log \frac{p(a \mid s)}{p(a)}\right] = H(A) - H(A \mid S), \tag{4}$$

where H is Shannon entropy, and $H(A|S)$ is conditional entropy of A given S. The higher mutual information between S and A means the lower uncertainty in A given the state S. This coincides with the observation that the actions selected by a well-performing agent are usually coupled with certain states. Therefore, given a joint distribution of state-action pairs induced by a (sub-optimal) behavior agent, it is natural to learn a policy that can recover the dependence between states and actions produced by the behavior agent. By regularizing the agent with $I(S;A)$ estimation, we encourage the agent to 1) perform policy update within the dataset distribution and 2) avoid being over-conservative and make sufficient use of the dataset information.

Let $\pi_\beta(a|s)$ represent a behavior policy and $p_\beta(s,a)$ be the joint distribution of state-action pairs induced by π_β. Calculating the mutual information is often intractable as accessing $p_\beta(s,a)$ is infeasible. Fortunately, in the problem of offline reinforcement learning, a dataset $\mathcal{D} = \{(s_t, a_t, r_t, s_{t+1})\}$ of transitions is given by drawing samples independently from $p_\beta(s,a)$. This dataset can thus be seen as a sample-based empirical joint distribution $p_\mathcal{D}(s,a)$ for p_β. Let $\mathcal{I}(\theta,\phi)$ denote a mutual information lower bound that relies on parameterized functions with parameters θ and ϕ[2], which are usually the policy network and Q network in the context of RL. We defer the derivation of such bounds in Sec. 4.2. Based on the above motivation, we aim at learning a policy that can approximate the mutual information of the dataset while being optimized to get the best possible cumulative return. We focus on the actor-critic framework, and formulate the offline RL problem with mutual information regularization as follows:

$$\min_{\phi} \quad \mathbb{E}_{s,a,s'\sim\mathcal{D}}\left[\frac{1}{2}\left(Q_\phi(s,a) - \mathcal{B}^{\pi_\theta} Q_\phi(s,a)\right)^2\right] - \alpha_1 \hat{\mathcal{I}}_\mathcal{D}(\theta,\phi), \qquad \text{(Policy Evaluation)} \tag{5}$$

$$\max_{\theta} \quad \mathbb{E}_{s\sim\mathcal{D}, a\sim\pi_\theta(a|s)}\left[Q_\phi(s,a)\right] + \alpha_2 \hat{\mathcal{I}}_\mathcal{D}(\theta,\phi), \qquad \text{(Policy Improvement)} \tag{6}$$

where α_1 and α_2 are the coefficients to balance RL objective and mutual information objective, and $\hat{\mathcal{I}}_\mathcal{D}(\theta,\phi)$ denotes the sample-based version of $\mathcal{I}(\theta,\phi)$ estimated from dataset $\mathcal{D}$.

4.2 State-Action Mutual Information Estimation

In this section, we develop practical solutions to approximate the mutual information $I(S;A)$ from samples of the joint distribution. We use the learning policy $\pi_\theta(a|s)$ as a variational variable and Eqn. 4 can be rewritten as:

$$I(S;A) = \mathbb{E}_{p(s,a)}\left[\log \frac{\pi_\theta(a|s)p(a|s)}{p(a)\pi_\theta(a|s)}\right] = \mathbb{E}_{p(s,a)}\left[\log \frac{\pi_\theta(a|s)}{p(a)}\right] + D_{\text{KL}}\left(p(s,a)||p(s)\pi_\theta(a|s)\right), \tag{7}$$

where $p(s)\pi_\theta(a|s)$ is an induced joint distribution. Let $\mathcal{I}_{\text{BA}} \triangleq \mathbb{E}_{p(s,a)}\left[\log \frac{\pi_\theta(a|s)}{p(a)}\right]$. We have $I(S;A) \geq \mathcal{I}_{\text{BA}}$ as the KL divergence is always non-negative. This is exactly the Barber-Agakov (BA) lower bound developed by [3].

[2]Note some lower bounds might only have one parameterized function.

To obtain tighter bounds, we turn to KL dual representations of $D_{\text{KL}}\left(p(s,a)||p(s)\pi_\theta(a|s)\right)$ in Eqn. 7. To this end, we choose $\mathcal{F}$ to be a set of parameterized functions $T_\phi : S \times A \to \mathbb{R}, \phi \in \Phi$, which can be seen as an energy function.

With the f-divergence dual representation, we derive MISA-f as

$$\mathcal{I}_{\text{MISA-}f} \triangleq \mathbb{E}_{p(s,a)}\left[\log \frac{\pi_\theta(a|s)}{p(a)}\right] + \mathbb{E}_{p(s,a)}\left[T_\phi(s,a)\right] - \mathbb{E}_{p(s)\pi_\theta(a|s)}\left[e^{T_\phi(s,a)-1}\right]. \tag{8}$$

The $\mathcal{I}_{\text{MISA-}f}$ bound is tight when $p(a|s) \propto \pi_\theta(a|s)e^{T_\phi(s,a)-1}$. Similarly, using the DV representation in Theorem 3.2, we can have another bound $\mathcal{I}_{\text{MISA-DV}} \leq I(S;A)$, as shown below:

$$\mathcal{I}_{\text{MISA-DV}} \triangleq \mathbb{E}_{p(s,a)}\left[\log \frac{\pi_\theta(a|s)}{p(a)}\right] + \mathbb{E}_{p(s,a)}\left[T_\phi(s,a)\right] - \log \mathbb{E}_{p(s)\pi_\theta(a|s)}\left[e^{T_\phi(s,a)}\right], \tag{9}$$

which is tight when $p(a|s) = \frac{1}{\mathcal{Z}}p(s)\pi_\theta(a|s)e^{T_\phi(s,a)}$, where $\mathcal{Z} = \mathbb{E}_{p(s)\pi_\theta(a|s)}\left[e^{T_\phi(s,a)}\right]$.

We observe that the KL term in Eqn. 7 can be rewritten as:

$$D_{\text{KL}}\left(p(s,a)||p(s)\pi_\theta(a|s)\right) = \mathbb{E}_{p(s)}\left[\mathbb{E}_{p(a|s)}\left[\log \frac{p(a|s)}{\pi_\theta(a|s)}\right]\right] = \mathbb{E}_{p(s)}\left[D_{\text{KL}}(p(a|s)||\pi_\theta(a|s))\right].$$

Applying the DV representation of $D_{\text{KL}}(p(a|s)||\pi_\theta(a|s))$, we can have a new lower bound $\mathcal{I}_{\text{MISA}}$:

$$\mathcal{I}_{\text{MISA}} \triangleq \mathbb{E}_{p(s,a)}\left[\log \frac{\pi_\theta(a|s)}{p(a)}\right] + \mathbb{E}_{p(s,a)}\left[T_\phi(s,a)\right] - \mathbb{E}_{p(s)} \log \mathbb{E}_{\pi_\theta(a|s)}\left[e^{T_\phi(s,a)}\right]. \tag{10}$$

The bound is tight when $p(a|s) = \frac{1}{\mathcal{Z}(s)}\pi_\theta(a|s)e^{T_\phi(s,a)}$, where $\mathcal{Z}(s) = \mathbb{E}_{\pi_\theta(a|s)}[e^{T_\phi(s,a)}]$.

Theorem 4.1 *Given the joint distribution of state s and action a, the lower bounds of mutual information $I(S;A)$ defined in Eqn. 8-10 have the following relations:*

$$I(S;A) \geq \mathcal{I}_{\textit{MISA}} \geq \mathcal{I}_{\textit{MISA-DV}} \geq \mathcal{I}_{\textit{MISA-}f}. \tag{11}$$

The proof is deferred to the appendix due to space limit.

4.3 Integration with Offline Reinforcement Learning

We now describe how our MISA lower bound is integrated into the above framework (Eqn. 12-13) to give a practical offline RL algorithm. Theoretically, $T_\psi(s,a)$ should be an arbitrary function optimized to accurately estimate the mutual information. As regularizing Q-functions is critical to offline RL, we propose to use a Q network $Q_\phi(s,a)$ as the energy function $T_\phi(s,a)$, and use $p_\mathcal{D}(s,a)$ as the joint distribution in Eqn. 10. Then we have the following objective to learn a Q-network during policy evaluation:

$$J_Q(\phi) = J_Q^\mathcal{B}(\phi) - \gamma_1 \mathbb{E}_{s,a\sim\mathcal{D}}\left[Q_\phi(s,a)\right] - \gamma_1 \mathbb{E}_{s\sim\mathcal{D}}\left[\log \mathbb{E}_{\pi_\theta(a|s)}\left[e^{Q_\phi(s,a)}\right]\right], \tag{12}$$

where $J_Q^\mathcal{B}(\phi) = \mathbb{E}_{s,a,s'\sim\mathcal{D}}\left[\frac{1}{2}\left(Q_\phi(s,a) - \mathcal{B}^{\pi_\theta}Q_\phi(s,a)\right)^2\right]$ represents the TD error. For policy improvement, note that the entropy term $H(a)$ in Eqn. 10 can be omitted as it is a constant given dataset $\mathcal{D}$. Thus, we have the below objective to maximize:

$$J_\pi(\theta) = \mathbb{E}_{s\sim\mathcal{D},a\sim\pi_\theta(a|s)}\left[Q_\phi(s,a)\right] + \gamma_2 \mathbb{E}_{s,a\sim\mathcal{D}}[\log \pi_\theta(a|s)] - \gamma_2 \mathbb{E}_{s\sim\mathcal{D}}\left[\log \mathbb{E}_{\pi_\theta(a|s)}\left[e^{Q_\phi(s,a)}\right]\right]. \tag{13}$$

The formulations for other regularizers (*e.g.*, $\mathcal{I}_{\text{MISA-DV}}$ and $\mathcal{I}_{\text{MISA-}f}$) can be derived similarly. A detailed description of the MISA algorithm for offline RL can be found in Algo. 1.

Intuitive Explanation on the Mutual Information Regularizer. By rearranging the terms in Eqn. 10, MISA can be written as:

$$\mathcal{I}_{\text{MISA}} = \mathbb{E}_{s,a\sim\mathcal{D}}\left[\log \frac{\pi_\theta(a \mid s)e^{Q_\phi(s,a)}}{\mathbb{E}_{\pi_\theta(a'|s)}\left[e^{Q_\phi(s,a')}\right]}\right], \tag{14}$$

Algorithm 1 Mutual Information Regularized Offline RL

Input: Initialize Q network Q_ϕ, policy network π_θ, dataset $\mathcal{D}$, hyperparameters α_1 and α_2.
for $t \in \{1, \ldots, \texttt{MAX_STEP}\}$ **do**
  Train the Q network by gradient descent with objective $J_Q(\phi)$ in Eqn. 12:
  $\phi := \phi - \eta_Q \nabla_\phi J_Q(\phi)$
  Improve policy network by gradient ascent with object $J_\pi(\theta)$ in Eqn. 13:
  $\theta := \theta + \eta_\pi \nabla_\theta \mathbb{E}_{s\sim\mathcal{D}, a\sim\pi_\theta(a|s)}[Q_\phi(s,a)] + \alpha_2 \nabla_\theta I_{\text{MISA}}$
end
Output: The well-trained π_θ.

where the log term can be seen as the log probability of a one-step improved policy. More specifically, for policy improvement with KL divergence regularization: $\max_\pi \mathbb{E}_{s\sim\mathcal{D}, a\sim\pi}[Q_\phi(s,a)] + D_{\text{KL}}(\pi||\pi_\theta)$, the optimal solution is given by $\pi^*_{\theta,\phi} \propto \pi_\theta(a|s)e^{Q_\phi(s,a)}$ [1, 34]. Therefore, $\mathcal{I}_{\text{MISA}}$ is rewritten with $\pi^*_{\theta,\phi}$ as $\mathcal{I}_{\text{MISA}} = \mathbb{E}_{s,a\sim\mathcal{D}}[\log \pi^*_{\theta,\phi}(a|s)]$, and maximizing it means maximizing the log-likelihood of the dataset using the improved policy. In other words, instead of directly fitting the policy on the dataset, which is short-sighted, this objective considers the optimization direction of the policy improvement step. Given the current policy and policy evaluation results, it first computes the analytic improved policy, and then forces the dataset likelihood to be maximized using the improved policy. In this way, even if an out-of-distribution state-action pair get an overestimated q value, $\mathcal{I}_{\text{MISA}}$ is going to suppress this value and make sure in-distribution data have relatively higher value estimation.

Unbiased Gradient Estimation For policy improvement with Eqn. 13, differentiating through a sampling distribution $\pi_\theta(a \mid s)$ is required for $\mathbb{E}_{s\sim D} \log \mathbb{E}_{\pi_\theta(a|s)}\left[e^{Q_\phi(s,a)}\right]$. For a Gaussian policy $\pi_\theta(a \mid s) = \mathcal{N}(\mu_\theta, \sigma_\theta)$, one could consider the reparameterization trick [23] and convert the objective as $\mathbb{E}_{s\sim D} \log \mathbb{E}_{\epsilon\sim\mathcal{N}(0,\mathbb{I})}\left[e^{Q_\phi(s,\mu_\theta+\epsilon*\sigma_\theta)}\right]$. However, this introduces high variance in offline reinforcement learning setups because we condition the policy improvement directly on the Q values of the out-of-distribution actions, which eventually gives a noisy policy. Hence, we aim to minimize the influence of Q values for policy improvement.

Differentiating Eqn. 10 with respect to policy parameters θ, we have

$$\frac{\partial \mathcal{I}_{\text{MISA}}}{\partial \theta} = \mathbb{E}_{s,a\sim D}\left[\frac{\log \pi_\theta(a \mid s)}{\partial \theta}\right] - \mathbb{E}_{s\sim D, a\sim p_{\theta,\phi}(a|s)}\left[\frac{\log \pi_\theta(a \mid s)}{\partial \theta}\right] \tag{15}$$

where $p_{\theta,\phi}(a \mid s) = \frac{\pi_\theta(a|s)e^{Q_\phi(s,a)})}{\mathbb{E}_{\pi_\theta(a|s)}\left[e^{Q_\phi(s,a)})\right]}$ is a self-normalized distribution. See appendix A.2 for a derivation. By optimizing Eqn. 15, we obtain an unbiased gradient estimation of the MISA objective with respect to the policy parameters, while minimizing the negative effects of the Q values of OOD actions. To sample from $p_{\theta,\phi}(a \mid s)$, one can consider Markov-Chain Monte-Carlo (MCMC) methods, e.g., Hamiltonian Monte Carlo [6].

4.4 Connections to Existing Offline RL Methods

We show that some existing offline RL methods can be viewed as special cases of MISA framework.

Behavior Cloning and BC Regularized RL We first show that behavior cloning is a form of mutual information regularizer. As shown by Eqn. 7, $\mathcal{I}_{\text{BA}} \triangleq \mathbb{E}_{s,a\sim\mathcal{D}}\left[\log \frac{\pi_\theta(a|s)}{p(a)}\right]$ gives a lower bound of mutual information. Since $H(a)$ is a consistent given datasets, maximizing $\mathcal{I}_{\text{BA}}$ is equivalent to maximizing $\mathbb{E}_{s,a\sim\mathcal{D}}\left[\log \pi_\theta(a|s)\right]$, which is exactly the objective for behavior cloning.

As for TD3+BC [14], the policy evaluation is unchanged, while the policy improvement objective is augmented by an MSE regularization term, *i.e.*, $\mathbb{E}_{s\sim\mathcal{D}}[Q(s,\pi_\theta(s))] - \gamma\mathbb{E}_{s,a\sim\mathcal{D}}\left[(\pi_\theta(s) - a)^2\right]$, where λ is a hyperparameter. Maximizing the negative MSE term is equivalent to maximizing $\mathbb{E}_{s,a\sim\mathcal{D}}\left[\log p_{\pi_\theta}(a|s)\right]$, where $p_{\pi_\theta} = Ce^{-\frac{1}{2}(\pi_\theta(s)-a)^2}$ is a Gaussian distribution, and C is a constant. This is a special case of Eqn. 13 when we remove the last log-mean-exp term.

Conservation Q Learning CQL [25] was proposed to alleviate the over-estimation issue of Q learning by making conservative updates to the Q values during policy evaluation. The policy

improvement is kept unchanged compared to standard Q learning. We focus on the entropy-regularized policy evaluation of CQL as below:

$$\min_{\phi} -\gamma_1 \mathbb{E}_{s\sim\mathcal{D}} \left[\mathbb{E}_{a\sim\pi_{\mathcal{D}}(a|s)}[Q_\phi(s,a)] - \log \sum_a e^{Q_\phi(s,a)} \right] + J_Q^{\mathcal{B}}(\phi) \tag{16}$$

where we highlight the main difference between it and our MISA policy evaluation (Eqn. 12) in blue. Let $\pi_{\mathrm{U}}(a|s)$ denote a uniform distribution of actions and $|A|$ is the number of actions. The log-sum-exp term can be written as $\log \mathbb{E}_{a\sim\pi_{\mathrm{U}}(a|s)}[Q_\phi(s,a)] + \log |A|$. Substituting it into Eqn. 16 and discarding the constant $\log |A|$, we recover the formulation in Eqn. 12. Therefore, CQL is actually doing mutual information regularization during policy evaluation. The key difference is that it is not using the current policy network as the variational distribution. Instead, a manually designed distribution is used in CQL. However, a uniform policy is usually suboptimal in environments with continuous actions. CQL thus constructs a mixed variational policy by drawing samples drawn from the current policy network, a uniform distribution, and the dataset. In our formulation, the variational distribution will be optimized to give a better mutual information estimation. This explains why MISA gives better performance than CQL.

5 Experiments

We first conduct ablation studies on MISA to better understand the influences of mutual information estimation on offline RL. Next, we compare MISA with a wide range of baseline algorithms to demonstrate its effectiveness on the D4RL dataset [13].

5.1 Experiment Setups

We follow the network architectures of CQL [25] and IQL [24], where a neural network of 3 encoding layers of size 256 is used for antmaze-v0 environments, and 2 encoding layers for other tasks, followed by an output layer. When approximating $\mathbb{E}_{\pi_\theta(a|s)}\left[e^{T_\psi(s,a)}\right]$, we use 50 Monte-Carlo samples. To sample from the non-parametric distribution $p_{\theta,\phi}(a \mid s) = \frac{\pi_\theta(a|s)e^{Q_\phi(s,a)}}{\mathbb{E}_{\pi_\theta(a|s)}\left[e^{Q_\phi(s,a)}\right]}$, we use the Hamiltonian Monte Carlo algorithm. In addition, for unbiased gradient estimation with MCMC samples, we use a burn-in step of 5, set the number of leapfrog steps to 2, and set the MCMC step size to 1. All these setups enable MISA to run at a relatively fast speed under careful implementation and do not suffer from the slow MCMC speed. For all tasks, we average the mean returns over 10 evaluation trajectories and 5 random seeds. In particular, following [24], we evaluate the antmaze-v0 environments for 100 episodes instead. To stabilize the training of our agents in antmaze-v0 environments, we follow [25] and normalize the reward by $r' = (r - 0.5) * 4$.

For practical implementations, we follow the CQL-Lagrange [25] implementation by constraining the Q-value update by a "budget" variable τ and rewrite Eqn. 12 as

$$\min_{Q} \max_{\gamma_1 \geq 0} \gamma_1 (\mathbb{E}_{s\sim\mathcal{D}} \left[\log \mathbb{E}_{\pi_\theta(a|s)} \left[e^{Q_\phi(s,a)} \right] \right] - \mathbb{E}_{s,a\sim\mathcal{D}} \left[Q_\phi(s,a) \right] - \tau) - J_Q^{\mathcal{B}}(\phi). \tag{17}$$

Eqn. 26 implies that if the expected value of the Q-value difference is less than the threshold τ, γ_1 will adjust to close to 0; if the Q-value difference is higher than the threshold τ, γ_1 will be larger and penalize Q-values harder. More implementation details are in the appendix.

5.2 Influences of Mutual Information Estimation on Offline RL

To begin with, we perform a thorough ablation study on MISA and its variants to better understand the influences of mutual information estimation on offline RL. Specifically, we aim to understand: 1) if tighter mutual information lower bound regularization helps to improve the performance, and 2) how accurately estimating the mutual information affects the final performance of offline RL.

Tighter mutual information lower bound helps to improve the performance of offline RL. In Table 1, BA stands for the Barber-Adakov Bound, and as discussed in Sect. 4.2, we have such inequality for mutual information estimation: BA $\leq$ MISA-f $\leq$ MISA-DV $\leq$ MISA. As shown in Table 1, we observe a performance trend consistent with our theoretical analysis, where MISA significantly improves over the three variants with lower mutual information estimation bound.

Table 1: Ablation studies on gym-locomotion-v2. k denotes the number of Monte-Carlo samples for estimating $\mathbb{E}_{\pi_\theta(a|s)}\left[e^{T_\psi(s,a)}\right]$, BI represents the burnin-steps for MCMC simulation, and BA denotes the use of Barber-Agakov Bound. In addition, MISA-x denotes different variants of MISA.

Dataset	k=5	k=20	BI=1	no BA	BA	MISA-f	MISA-DV	MISA-biased	MISA-T	MISA
halfcheetah-medium-v2	47.1	46.9	47.2	49.1	56.3	43.5	45.5	48.4	**60.8**	47.4
hopper-medium-v2	62.2	65.3	61.8	64.4	1.2	60.5	61.6	65.7	1.9	**67.1**
walker2d-medium-v2	83.3	83.9	81.8	83.8	7.5	73.2	82.8	**84.2**	2.8	84.1
halfcheetah-medium-replay-v2	45.4	45.3	45.2	46.5	52.4	39.8	43.8	46.9	**53.1**	45.6
hopper-medium-replay-v2	79.9	88.4	72.9	**100.3**	56.4	34.8	45.9	98.1	44.2	98.6
walker2d-medium-replay-v2	83.7	**86.9**	82.8	86.1	51.1	34.9	81.4	80.6	79.8	86.2
halfcheetah-medium-expert-v2	94.5	92.8	92.8	87.1	26.8	57.6	92.4	84.6	24.9	**94.7**
hopper-medium-expert-v2	105.7	102.7	93.4	89.6	1.3	57.7	**111.5**	103.2	0.7	109.8
walker2d-medium-expert-v2	109.2	**109.4**	109.3	108.1	1.4	102.7	108.8	109.2	2.8	**109.4**
gym-locomotion-v2 (total)	711	721.6	687.2	715	254.4	504.7	673.7	720.9	271.0	**742.9**

Accurately estimating the mutual information is critical to the performance of offline RL. In Table 1, no-BA stands for removing the Barber-Agakov term in Eqn. 10, which gives an inaccurate estimation to mutual information, neither an upper bound nor a lower bound. We can observe that the no-BA variant suffers from a performance drop compared with MISA. Similarly, we vary the number (k) of Monte-Carlo samples for approximating $\mathbb{E}_{\pi_\theta(a|s)}\left[e^{T_\psi(s,a)}\right]$ and reduce the burn-in steps of MCMC sampling process. Both operations increase the Monte-Carlo approximation errors to MISA. Comparing $k = 5$, $k = 20$, and MISA ($k = 50$), the performance increases monotonically; comparing MISA (BI=5) with BI=1, we can observe a sharp performance drop. We then conclude that MISA requires careful Monte-Carlo approximation for good performance.

Unbiased gradient estimations improve the performance of MISA. MISA-biased ignores the bias correction term in Eqn. 15. Although MISA-biased outperforms the baselines, it still performs worse than MISA. This suggests that by correcting the gradient estimation with additional MCMC samples, MISA achieves better regularized policy learning in offline RL. Note that by setting a minimal 5 MCMC burn-in steps, MISA only slightly sacrifices the training speed, but during inference, MISA directly takes the learned model-free policy which does not affect the inference speed.

Estimating mutual information with Q functions helps to stabilize MISA. MISA-T is a variant that learns to estimate the mutual information lower bound by learning an independent function $T_\psi(s, a)$, rather than directly using $Q_\phi(s, a)$. Theoretically, MISA-T would give a more accurate mutual information estimation than MISA. Although practically we observe MISA-T indeed achieves superior performance on tasks like halfcheetah-medium-v2 (60.8) and halfcheetah-medium-replay-v2 (53.1), it mostly fails on others. MISA, instead, balances the mutual information estimation with Q-value regularization, and achieves a better overall performance.

5.3 Offline Reinforcement Learning on D4RL Benchmarks

Gym Locomotion Tasks. We first evaluate MISA on the MuJoCo-style continuous control tasks, reported as gym-locomotion-v2 in Table 2. We observe that MISA improves the performance of baselines by a large margin. Specifically, MISA is less sensitive to the characteristics of data distributions. The medium datasets include trajectories collected by a SAC agent trained to reach 1/3 of the performance of an expert; the medium-replay datasets contain all data samples of the replay buffer during the training of the medium SAC agent, which covers the noisy exploration process of the medium agent. We can observe that prior methods are generally sensitive to noisy sub-optimal data in medium and medium-replay environments, while MISA outperforms them by a large margin. In particular, MISA achieves near-expert performance on walker2d-medium-replay with only sub-optimal trajectories. This indicates that by regularizing the policy and Q-values within the mutual information of the dataset, we can fully exploit the data and perform safe and accurate policy improvement during RL. Moreover, on medium-expert environments, where the datasets are mixtures of medium agents and experts, MISA successfully captures the multi-modality of the datasets and allows further improvements of the policy over baselines.

Adroit Tasks. According to [14], adroit tasks require strong policy regularization to overcome the extrapolation error, because the datasets are either generated by humans (adroit-human-v0), which would show a narrow policy distribution, or a mixture of human demonstrations and a behavior

Table 2: Average normalized score on the D4RL benchmark. Results of baselines are taken directly from [24]. For the missing results of 10%, AWAC, and OneStep RL, we re-implement the baselines and report the results.

Dataset	BC	10%BC	DT	AWAC	OneStep RL	TD3+BC	CQL	IQL	MISA
halfcheetah-medium-v2	42.6	42.5	42.6	43.5	**48.4**	48.3	44	47.4	47.4±0.2
hopper-medium-v2	52.9	56.9	**67.6**	57	59.6	59.3	58.5	66.3	67.1±2.7
walker2d-medium-v2	75.3	75	74	72.4	81.8	83.7	72.5	78.3	**84.1**±0.3
halfcheetah-medium-replay-v2	36.6	40.6	36.6	40.5	38.1	44.6	45.5	44.2	**45.6**±0.2
hopper-medium-replay-v2	18.1	75.9	82.7	37.2	**97.5**	60.9	95	94.7	**98.6**±2.5
walker2d-medium-replay-v2	26	62.5	66.6	27	49.5	81.8	77.2	73.9	**86.2**±0.5
halfcheetah-medium-expert-v2	55.2	92.9	86.8	42.8	93.4	90.7	91.6	86.7	**94.7**±1.9
hopper-medium-expert-v2	52.5	**110.9**	107.6	55.8	103.3	98	105.4	91.5	109.8±1.8
walker2d-medium-expert-v2	107.5	109	108.1	74.5	**113**	110.1	108.8	109.6	109.4±0.3
gym-locomotion-v2 (total)	466.7	666.2	672.6	450.7	684.6	677.4	698.5	692.6	**742.9**±4.6
kitchen-complete-v0	**65**	7.2	-	39.3	57	16.8	43.8	62.5	**70.2**±6.8
kitchen-partial-v0	38	**66.8**	-	36.6	53.1	22.4	49.8	46.3	45.7±6.2
kitchen-mixed-v0	51.5	50.9	-	22	47.6	46.2	51	51	**56.6**±4.3
kitchen-v0 (total)	154.5	124.9	-	97.9	157.7	85.4	144.6	159.8	**172.5**±10.2
pen-human-v0	63.9	-2	-	15.6	71.8	64.8	37.5	71.5	**88.1**±9.7
hammer-human-v0	1.2	0	-	0.1	1.2	1.8	4.4	1.4	**8.1**±1.3
door-human-v0	2	0	-	0.1	5.4	0	**9.9**	4.3	5.2±2.4
relocate-human-v0	0.1	0	-	0.1	1.9	0.1	0.2	0.1	0.1±0.1
pen-cloned-v0	37	0	-	24.7	**60**	49	39.2	37.3	58.6±4.4
hammer-cloned-v0	0.6	0	-	0.3	2.1	0.2	2.1	2.1	**2.2**±0.4
door-cloned-v0	0	0	-	0.1	0.4	0	0.4	**1.6**	0.5±0.2
relocate-cloned-v0	-0.3	0	-	-0.1	-0.1	-0.2	-0.1	-0.2	-0.1±0.1
adroit-v0 (human+cloned)	104.5	-2	0	40.9	142.7	115.7	93.6	118.1	**162.7**±11.0
antmaze-umaze-v0	54.6	62.8	59.2	56.7	64.3	78.6	74	87.5	**92.3**±5.6
antmaze-umaze-diverse-v0	45.6	50.2	53	49.3	60.7	71.4	84	62.2	**89.1**±4.7
antmaze-medium-play-v0	0	5.4	0	0	0.3	10.6	61.2	**71.2**	63±6.9
antmaze-medium-diverse-v0	0	9.8	0	0.7	0	3	53.7	**70**	62.8±7.2
antmaze-large-play-v0	0	0	0	0	**0**	0.2	15.8	**39.6**	17.5±7.8
antmaze-large-diverse-v0	0	6	0	1	0	0	14.9	**47.5**	23.4±8.1
antmaze-v0 (total)	100.2	134.2	112.2	107.7	125.3	163.8	303.6	**378**	348.1±16.7

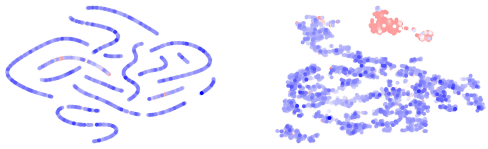
cloning policy (adroit-cloned-v0). We observe that MISA provides a stronger regularization and significantly outperforms the baselines on adroit.

Kitchen Tasks. An episode of Kitchen environment consists of multiple sub-tasks that can be mixed in an arbitrary order. We observe that MISA outperforms baselines on both kitchen-complete-v0 and kitchen-mixed-v0, while achieving slightly worse performance on kitchen-partial-v0. Specifically, on kitchen-mixed, the result is consistent with our assumption that by better regularizing the policy, MISA guarantees a safer and in-distribution policy improvement step in offline RL.

Antmaze Tasks. On the challenging AntMaze domain with sparse delayed reward, we observe that MISA generally outperforms CQL and achieves the best performance on umaze environments. However, MISA performs worse than IQL on challenging large environments. Multi-step value update is often necessary for learning a robust value estimation in these scenarios [24] while MISA adopts a single-step SAC for the base RL algorithm.

5.4 Visualization of Embedding

In Fig. 1, we visualize the embeddings before the output layer of Q-value networks, given different mutual information bounds (BA and MISA). We select a subset from walker2d-medium-v2 dataset to study the division of low reward (blue) and high reward (red) (s, a) pairs. We color each point by the reward $r(s, a)$. As discussed in Sect. 4.2, BA gives a lowest bound for mutual information estimation and MISA produces the tightest bound. In Fig. 1, we observe a consistent result. The embeddings of BA converge to a set of regular curves and fail to cluster the high $r(s, a)$, because Q-values have



(a) BA (b) MISA

Figure 1: tSNE of the Q-value network embeddings of walker2d-medium-v2 dataset, where red color denote high reward and blue color denote low reward.

converged to indistinguishably high values (3×10^{12}) for all (s, a) pairs. In contrast, MISA successfully learns to cluster the (s, a) pairs with a high reward into a cluster. From this perspective, we claim that regularizing the mutual information encourages learning a robust representation in offline RL scenarios.

6 Conclusions

We present the MISA framework for offline reinforcement learning by directly regularizing policy improvement and policy evaluation with the mutual information between state-action pairs of the dataset. MISA connects mutual information estimation with RL by constructing tractable lower bounds, treating the learning policy as a variational distribution and Q values as energy functions. The resulting tractable lower bound resembles a non-parametric energy-based distribution, which can be interpreted as the likelihood of a one-step improved policy given the current value estimation. In our experiments, we show how accurate mutual information estimation affects offline RL and the proposed method, MISA, has outperformed a wide range of baselines on D4RL benchmark by large margins, reaching a total point of 742.9 on gym-locomotion tasks of D4RL datasets. However, as a preliminary exploration in this direction, MISA does not sufficiently exploit the most advanced mutual information estimation methods. Future works could consider further optimizing the mutual information estimation to achieve higher performances.

References

[1] Abbas Abdolmaleki, Jost Tobias Springenberg, Yuval Tassa, Rémi Munos, Nicolas Heess, and Martin A. Riedmiller. Maximum a posteriori policy optimisation. In *International Conference on Learning Representations*, 2018.

[2] Gaon An, Seungyong Moon, Jang-Hyun Kim, and Hyun Oh Song. Uncertainty-based offline reinforcement learning with diversified q-ensemble. *Advances in neural information processing systems*, 34:7436–7447, 2021.

[3] David Barber and Felix Agakov. The IM algorithm: a variational approach to information maximization. *Advances in neural information processing systems*, 16(320):201, 2004.

[4] Mohamed Ishmael Belghazi, Aristide Baratin, Sai Rajeswar, Sherjil Ozair, Yoshua Bengio, R. Devon Hjelm, and Aaron C. Courville. Mutual information neural estimation. In Jennifer G. Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, 2018.

[5] Christopher Berner, Greg Brockman, Brooke Chan, Vicki Cheung, Przemysław Debiak, Christy Dennison, David Farhi, Quirin Fischer, Shariq Hashme, Chris Hesse, et al. Dota 2 with large scale deep reinforcement learning. *arXiv preprint arXiv:1912.06680*, 2019.

[6] Michael Betancourt. A conceptual introduction to hamiltonian monte carlo. *arXiv preprint arXiv:1701.02434*, 2017.

[7] James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, George Necula, Adam Paszke, Jake VanderPlas, Skye Wanderman-Milne, and Qiao Zhang. JAX: composable transformations of Python+NumPy programs, 2018.

[8] Rob Brekelmans, Sicong Huang, Marzyeh Ghassemi, Greg Ver Steeg, Roger Baker Grosse, and Alireza Makhzani. Improving mutual information estimation with annealed and energy-based bounds. In *International Conference on Learning Representations*, 2022.

[9] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning*, 2020.

[10] Andrzej Cichocki and Shun-ichi Amari. Families of alpha-beta-and gamma-divergences: Flexible and robust measures of similarities. *Entropy*, 12(6):1532–1568, 2010.

[11] Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. Fast and accurate deep network learning by exponential linear units (elus). *arXiv preprint arXiv:1511.07289*, 2015.

[12] Monroe D Donsker and SR Srinivasa Varadhan. Asymptotic evaluation of certain markov process expectations for large time, i. *Communications on Pure and Applied Mathematics*, 28(1):1–47, 1975.

[13] 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.

[14] Scott Fujimoto and Shixiang Shane Gu. A minimalist approach to offline reinforcement learning. *Advances in Neural Information Processing Systems*, 2021.

[15] Scott Fujimoto, Herke Hoof, and David Meger. Addressing function approximation error in actor-critic methods. In *International conference on machine learning*, 2018.

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

[17] Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International conference on machine learning*, 2018.

[18] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Conference on Computer Vision and Pattern Recognition*, 2020.

[19] Jonathan Heek, Anselm Levskaya, Avital Oliver, Marvin Ritter, Bertrand Rondepierre, Andreas Steiner, and Marc van Zee. Flax: A neural network library and ecosystem for JAX, 2020.

[20] Gregory Kahn, Adam Villaflor, Bosen Ding, Pieter Abbeel, and Sergey Levine. Self-supervised deep reinforcement learning with generalized computation graphs for robot navigation. In *International Conference on Robotics and Automation*, 2018.

[21] Bingyi Kang, Xiao Ma, Chao Du, Tianyu Pang, and Shuicheng Yan. Efficient diffusion policies for offline reinforcement learning. *arXiv preprint arXiv:2305.20081*, 2023.

[22] Bingyi Kang, Xiao Ma, Yirui Wang, Yang Yue, and Shuicheng Yan. Improving and benchmarking offline reinforcement learning algorithms. *arXiv preprint arXiv:2306.00972*, 2023.

[23] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In Yoshua Bengio and Yann LeCun, editors, *International Conference on Learning Representations*, 2014.

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

[25] Aviral Kumar, Aurick Zhou, George Tucker, and Sergey Levine. Conservative q-learning for offline reinforcement learning. *Advances in Neural Information Processing Systems*, 2020.

[26] Sergey Levine, Chelsea Finn, Trevor Darrell, and Pieter Abbeel. End-to-end training of deep visuomotor policies. *The Journal of Machine Learning Research*, 2016.

[27] 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.

[28] 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.

[29] Ofir Nachum, Mohammad Norouzi, Kelvin Xu, and Dale Schuurmans. Bridging the gap between value and policy based reinforcement learning. *Advances in neural information processing systems*, 2017.

[30] XuanLong Nguyen, Martin J Wainwright, and Michael I Jordan. Estimating divergence functionals and the likelihood ratio by convex risk minimization. *IEEE Transactions on Information Theory*, 2010.

[31] Sebastian Nowozin. Improved information gain estimates for decision tree induction. *arXiv preprint arXiv:1206.4620*, 2012.

[32] Sebastian Nowozin, Botond Cseke, and Ryota Tomioka. f-gan: Training generative neural samplers using variational divergence minimization. *Advances in neural information processing systems*, 29, 2016.

[33] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[34] Xue Bin Peng, Aviral Kumar, Grace Zhang, and Sergey Levine. Advantage-weighted regression: Simple and scalable off-policy reinforcement learning. *arXiv preprint arXiv:1910.00177*, 2019.

[35] Ben Poole, Sherjil Ozair, Aaron Van Den Oord, Alex Alemi, and George Tucker. On variational bounds of mutual information. In *International Conference on Machine Learning*, pages 5171–5180. PMLR, 2019.

[36] Manolis Savva, Jitendra Malik, Devi Parikh, Dhruv Batra, Abhishek Kadian, Oleksandr Maksymets, Yili Zhao, Erik Wijmans, Bhavana Jain, Julian Straub, Jia Liu, and Vladlen Koltun. Habitat: A platform for embodied ai research. In *International Conference on Computer Vision, ICCV*, 2019.

[37] John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. Trust region policy optimization. In *International conference on machine learning*, 2015.

[38] 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.

[39] 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*, 2017.

[40] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

[41] Zhendong Wang, Jonathan J Hunt, and Mingyuan Zhou. Diffusion policies as an expressive policy class for offline reinforcement learning. In *The Eleventh International Conference on Learning Representations*, 2023.

[42] 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. *Advances in Neural Information Processing Systems*, 2020.

[43] Yifan Wu, George Tucker, and Ofir Nachum. Behavior regularized offline reinforcement learning. *arXiv preprint arXiv:1911.11361*, 2019.

[44] Tianhe Yu, Aviral Kumar, Rafael Rafailov, Aravind Rajeswaran, Sergey Levine, and Chelsea Finn. Combo: Conservative offline model-based policy optimization. *Advances in Neural Information Processing Systems*, 2021.

[45] Yang Yue, Bingyi Kang, Xiao Ma, Gao Huang, Shiji Song, and Shuicheng Yan. Offline prioritized experience replay, 2023.

[46] Yang Yue, Bingyi Kang, Xiao Ma, Zhongwen Xu, Gao Huang, and Shuicheng Yan. Boosting offline reinforcement learning via data rebalancing, 2022.

Broader Impact

MISA framework provides a simple yet effective approach to policy learning from offline datasets. Although the results presented in this paper only consider simulated environments, given the generality of MISA, it could be potentially effective on learning real-robot policies in more complex environments. We should be cautious about the misuse of the method proposed. Depending on the specific application scenarios, it might be harmful to democratic privacy and safety.

A Proofs and Derivations

A.1 Proof for Theorem 4.1

We first show $\mathcal{I}_{\text{MISA}}$, $\mathcal{I}_{\text{MISA-DV}}$ and $\mathcal{I}_{\text{MISA-}f}$ are lower bounds for mutual information $I(S, A)$.

Let $\mu_{\theta,\phi}(a|s) \triangleq \frac{1}{\mathcal{Z}(s)}\pi_\theta(a|s)e^{T_\phi(s,a)}$, where $\mathcal{Z}(s) = \mathbb{E}_{\pi_\theta(a|s)}[e^{T_\phi(s,a)}]$, $\mathcal{I}_{\text{MISA}}$ can be written as:

$$\begin{aligned}
\mathcal{I}_{\text{MISA}} &\triangleq \mathbb{E}_{p(s,a)}\left[\log \frac{\pi_\theta(a|s)}{p(a)}\right] + \mathbb{E}_{p(s,a)}\left[T_\phi(s,a)\right] - \mathbb{E}_{p(s)} \log \mathbb{E}_{\pi_\theta(a|s)}\left[e^{T_\phi(s,a)}\right] \\
&= \mathbb{E}_{p(s,a)}\left[\log \frac{p(a|s)}{p(a)}\right] - \mathbb{E}_{p(s,a)}[\log p(a|s)] \\
&\quad + \mathbb{E}_{p(s,a)}[\log \pi_\theta(a|s)] + \mathbb{E}_{p(s,a)}\left[T_\phi(s,a)\right] - \mathbb{E}_{p(s)}[\log \mathcal{Z}(s)] \\
&= I(S,A) - \mathbb{E}_{p(s)}\left[D_{\text{KL}}(p(a|s)||\mu_{\theta,\phi}(a|s))\right] \leq I(S,A).
\end{aligned} \tag{18}$$

The above inequality holds as the KL divergence is always non-negative.

Similarly, let $\mu_{\theta,\phi}(s,a) \triangleq \frac{1}{\mathcal{Z}}p(s)\pi_\theta(a|s)e^{T_\phi(s,a)}$, where $\mathcal{Z}(s) = \mathbb{E}_{p(s)\pi_\theta(a|s)}[e^{T_\phi(s,a)}]$, $\mathcal{I}_{\text{MISA-DV}}$ can be written as:

$$\begin{aligned}
\mathcal{I}_{\text{MISA-DV}} &\triangleq \mathbb{E}_{p(s,a)}\left[\log \frac{\pi_\theta(a|s)}{p(a)}\right] + \mathbb{E}_{p(s,a)}\left[T_\phi(s,a)\right] - \log \mathbb{E}_{p(s)\pi_\theta(a|s)}\left[e^{T_\phi(s,a)}\right] \\
&= \mathbb{E}_{p(s,a)}\left[\log \frac{p(a|s)}{p(a)}\right] - \mathbb{E}_{p(s,a)}[\log p(a|s)] \\
&\quad + \mathbb{E}_{p(s,a)}[\log \pi_\theta(a|s)] + \mathbb{E}_{p(s,a)}\left[T_\phi(s,a)\right] - \log \mathcal{Z} \\
&= I(S,A) - D_{\text{KL}}(p(s,a)||\mu_{\theta,\phi}(s,a)) \leq I(S,A).
\end{aligned} \tag{19}$$

The above inequality holds as the KL divergence is always non-negative.

Consider the generalized KL-divergence [10, 8] between two un-normalized distributions $\tilde{p}(x)$ and $\tilde{q}(x)$ defined by

$$D_{\text{GKL}}(\tilde{p}(x)||\tilde{q}(x)) = \int \tilde{p}(x) \log \frac{\tilde{p}(x)}{\tilde{q}(x)} - \tilde{p}(x) + \tilde{q}(x)dx, \tag{20}$$

which is always non-negative and reduces to KL divergence when $\tilde{p}$ and $\tilde{q}$ are normalized. Let $\tilde{\mu}_{\theta,\phi}(a|s) \triangleq \pi_\theta(a|s)e^{T_\phi(s,a)-1}$ denote an un-normalized policy. We can rewrite $\mathcal{I}_{\text{MISA-}f}$ as

$$\begin{aligned}
\mathcal{I}_{\text{MISA-}f} &\triangleq \mathbb{E}_{p(s,a)}\left[\log \frac{\pi_\theta(a|s)}{p(a)}\right] + \mathbb{E}_{p(s,a)}\left[T_\phi(s,a)\right] - \mathbb{E}_{p(s)\pi_\theta(a|s)}\left[e^{T_\phi(s,a)-1}\right] \\
&= \mathbb{E}_{p(s,a)}\left[\log \frac{p(a|s)}{p(a)}\right] - \mathbb{E}_{p(s,a)}[\log p(a|s)] \\
&\quad + \mathbb{E}_{p(s,a)}[\log \pi_\theta(a|s)] + \mathbb{E}_{p(s,a)}\left[T_\phi(s,a) - 1\right] + 1 - \mathbb{E}_{p(s)\pi_\theta(a|s)}\left[e^{T_\phi(s,a)-1}\right] \\
&= I(S,A) - \mathbb{E}_{p(s)}\left[D_{\text{GKL}}(p(a|s)||\tilde{\mu}_{\theta,\phi}(a|s))\right] \leq I(S,A).
\end{aligned} \tag{21}$$

So far, we have proven that $\mathcal{I}_{\text{MISA}}$, $\mathcal{I}_{\text{MISA-DV}}$ and $\mathcal{I}_{\text{MISA-}f}$ mutual information lower bounds. Then we are going to prove their relations by starting fromt he relation between $\mathcal{I}_{\text{MISA}}$ and $\mathcal{I}_{\text{MISA-DV}}$.

$$\begin{aligned}
\mathcal{I}_{\text{MISA}} - \mathcal{I}_{\text{MISA-DV}} &= D_{\text{KL}}(p(s,a)||\mu_{\theta,\phi}(s,a)) - \mathbb{E}_{p(s)}\left[D_{\text{KL}}(p(a|s)||\mu_{\theta,\phi}(a|s))\right] \\
&= \mathbb{E}_{p(s)}\mathbb{E}_{p(a|s)}\left[\log\frac{p(s,a)}{p(a|s)} - \log\frac{\mu_{\theta,\phi}(s,a)}{\mu_{\theta,\phi}(a|s)}\right] \\
&= \mathbb{E}_{p(s)}\mathbb{E}_{p(a|s)}\left[\log p(s) - \log\frac{1}{\mathcal{Z}}p(s)\mathcal{Z}(s)\right] \\
&= \mathbb{E}_{p(s)}\left[\log p(s) - \log\frac{1}{\mathcal{Z}}p(s)\mathcal{Z}(s)\right] \\
&= D_{\text{KL}}\left(p(s)||\frac{1}{\mathcal{Z}}p(s)\mathcal{Z}(s)\right) \geq 0,
\end{aligned} \tag{22}$$

where $\frac{1}{\mathcal{Z}}p(s)\mathcal{Z}(s)$ is a self-normalized distribution as $\mathcal{Z} = \mathbb{E}_{p(s)[\mathcal{Z}(s)]}$. Therefore, we have $\mathcal{I}_{\text{MISA}} \geq \mathcal{I}_{\text{MISA-DV}}$.

Similarly, the relation between $\mathcal{I}_{\text{MISA-DV}}$ and $\mathcal{I}_{\text{MISA-}f}$ is given by:

$$\begin{aligned}
\mathcal{I}_{\text{MISA-DV}} - \mathcal{I}_{\text{MISA-}f} &= \mathbb{E}_{p(s)}\left[D_{\text{GKL}}(p(a|s)||\tilde{\mu}_{\theta,\phi}(a|s))\right] - D_{\text{KL}}(p(s,a)||\mu_{\theta,\phi}(s,a)) \\
&= \mathbb{E}_{p(s)}\mathbb{E}_{p(a|s)}\left[\log\frac{p(a|s)}{p(s,a)} - \log\frac{\tilde{\mu}_{\theta,\phi}(a|s)}{\mu_{\theta,\phi}(s,a)}\right] - 1 + \mathbb{E}_{p(s)}\mathbb{E}_{\pi_\theta(a|s)}\left[e^{T_\phi(s,a)-1}\right] \\
&= \mathbb{E}_{p(s)}\mathbb{E}_{p(a|s)}\left[-\log p(s) - \log\frac{\tilde{\mu}_{\theta,\phi}(a|s)}{\mu_{\theta,\phi}(s,a)}\right] - 1 + \mathbb{E}_{p(s)}\mathbb{E}_{\pi_\theta(a|s)}\left[e^{T_\phi(s,a)-1}\right] \\
&= \mathbb{E}_{p(s)}\mathbb{E}_{p(a|s)}\left[\log\frac{\mu_{\theta,\phi}(s,a)}{p(s)\tilde{\mu}_{\theta,\phi}(a|s)}\right] - \mathbb{E}_{\mu_{\theta,\phi}(s,a)}[1] + \mathbb{E}_{p(s)}\mathbb{E}_{\pi_\theta(a|s)}\left[e^{T_\phi(s,a)-1}\right] \\
&= \mathbb{E}_{p(s,a)}\left[\log\frac{e}{\mathcal{Z}}\right] - \mathbb{E}_{\mu_{\theta,\phi}(s,a)}[1] + \mathbb{E}_{p(s)}\mathbb{E}_{\pi_\theta(a|s)}\left[e^{T_\phi(s,a)-1}\right] \\
&= \mathbb{E}_{\mu_{\theta,\phi}(s,a)}\left[\log\frac{e}{\mathcal{Z}}\right] - \mathbb{E}_{\mu_{\theta,\phi}(s,a)}[1] + \mathbb{E}_{p(s)}\mathbb{E}_{\pi_\theta(a|s)}\left[e^{T_\phi(s,a)-1}\right] \\
&= \mathbb{E}_{\mu_{\theta,\phi}(s,a)}\left[\log\frac{\mu_{\theta,\phi}(s,a)}{p(s)\tilde{\mu}_{\theta,\phi}(a|s)}\right] - \mathbb{E}_{\mu_{\theta,\phi}(s,a)}[1] + \mathbb{E}_{p(s)}\mathbb{E}_{\pi_\theta(a|s)}\left[e^{T_\phi(s,a)-1}\right] \\
&= D_{\text{GKL}}\left(\mu_{\theta,\phi}(s,a)||p(s)\tilde{\mu}_{\theta,\phi}(a|s)\right) \geq 0,
\end{aligned} \tag{23}$$

where $p(s)\tilde{\mu}_{\theta,\phi}(a|s)$ is an unnormalized joint distribution. Therefore, we have $I(S,A) \geq \mathcal{I}_{\text{MISA}} \geq \mathcal{I}_{\text{MISA-DV}} \geq \mathcal{I}_{\text{MISA-}f}$.

A.2 Derivation of MISA Gradients

We detail how the unbiased gradient is derived in Sec.4.3.

$$\begin{aligned}
\frac{\partial \mathcal{I}_{\text{MISA}}}{\partial\theta} &= \mathbb{E}_{s,a\sim D}\left[\frac{\log\pi_\theta(a \mid s)}{\partial\theta}\right] - \mathbb{E}_{s\sim D}\left[\frac{\partial \log \mathbb{E}_{\pi_\theta(a|s)}[e^{Q_\phi(s,a)}]}{\partial\theta}\right] \\
&= \mathbb{E}_{s,a\sim D}\left[\frac{\log\pi_\theta(a \mid s)}{\partial\theta}\right] - \mathbb{E}_{s\sim\mathcal{D}}\left[\mathbb{E}_{\pi_\theta(a|s)}\left[\frac{e^{Q_\phi(s,a)}}{\mathbb{E}_{\pi_\theta(a|s)}\left[e^{Q_\phi(s,a)}\right]}\frac{\log\pi_\theta(a \mid s)}{\partial\theta}\right]\right] \quad (24) \\
&= \mathbb{E}_{s,a\sim D}\left[\frac{\log\pi_\theta(a \mid s)}{\partial\theta}\right] - \mathbb{E}_{s\sim D, a\sim p_{\theta,\phi}(a|s)}\left[\frac{\log\pi_\theta(a \mid s)}{\partial\theta}\right] \quad (25)
\end{aligned}$$

for Eqn. 24, we use the log-derivative trick.

B Implementation Details

We follow the network architectures of CQL [25] and IQL [24], where a neural network of 3 encoding layers of size 256 is used for antmaze-v0 environments, and 2 encoding layers for other tasks, followed

by an output layer. We use ELU activation function [11] and SAC [17] as the base RL algorithm. Besides, we use a learning rate of 1×10^{-4} for both the policy network and Q-value network with a cosine learning rate scheduler. When approximating $\mathbb{E}_{\pi_\theta(a|s)}\left[e^{T_\psi(s,a)}\right]$, we use 50 Monte-Carlo samples. To sample from the non-parametric distribution $p_{\theta,\phi}(a \mid s) = \frac{\pi_\theta(a|s)e^{Q_\phi(s,a)}}{\mathbb{E}_{\pi_\theta(a|s)}\left[e^{Q_\phi(s,a)}\right]}$, we use Hamiltonian Monte Carlo algorithm. In addition, for unbiased gradient estimation with MCMC samples, we use a burn-in steps of 5. For all tasks, we average the mean returns over 10 evaluation trajectories and 5 random seeds. In particular, following [24], we evaluate the antmaze-v0 environments for 100 episodes instead. To stabilize the training of our agents in antmaze-v0 environments, we follow [25] and normalize the reward by $r' = (r - 0.5) * 4$. As MCMC sampling is slow, we trade-off its accuracy with efficiency by choosing moderately small iteration configurations. We set the MCMC burn-in steps to 5, number of leapfrog steps to 2, and MCMC step size to 1.

For practical implementations, we follow the CQL-Lagrange [25] implementation by constraining the Q-value update by a "budget" variable τ and rewrite Eqn. 12 as

$$\min_{Q} \max_{\gamma_1 \geq 0} \gamma_1 (\mathbb{E}_{s\sim\mathcal{D}}\left[\log \mathbb{E}_{\pi_\theta(a|s)}\left[e^{Q_\phi(s,a)}\right]\right] - \mathbb{E}_{s,a\sim\mathcal{D}}\left[Q_\phi(s,a)\right] - \tau) - J_Q^{\mathcal{B}}(\phi). \tag{26}$$

Eqn. 26 implies that if the expected value of Q-value difference is less than the threshold τ, γ_1 will adjust to close to 0; if the Q-value difference is higher than the threshold τ, γ_1 will be larger and penalize Q-values harder. We set $\tau = 10$ for antmaze-v0 environments and $\tau = 3$ for adroit-v0 and kitchen-v0 environments. For gym-locomotion-v2 tasks, we disable this function and direction optimize Eqn. 12, because these tasks have a relatively short horizon and dense reward, and further constraining the Q values is less necessary. Our code is implemented in JAX [7] with Flax [19]. All experiments are conducted on NVIDIA 3090 GPUs.