Supplementary Material: Diffusion Policies Creating a Trust Region for Offline Reinforcement Learning

A Related Work

Expressive Generative Models for Behavior Cloning Behavior cloning refers to the task of learning the behavior policy that was used to collect static datasets. Generative models are often employed for behavior cloning due to their expressive power. For instance, EMaQ [Ghasemipour et al., 2021] uses an auto-regressive model for behavior cloning. BCQ [Fujimoto et al., 2019] utilizes a Conditional Variational Autoencoder (VAE), while Florence et al. [2022] employ energy-based models. GAN-Joint [Yang et al., 2022] leverages GANs, and several studies [Wang et al., 2022a, Janner et al., 2022, Pearce et al., 2023] utilize diffusion models for behavior cloning. Diffusion models have demonstrated strong performance due to their ability to capture multimodal distributions. However, they may suffer from increased training and inference times because of the iterative denoising process required for sampling.

Efficiency Improvement in Diffusion-Based RL Methods. Several studies aim to accelerate the training of diffusion models in offline RL settings. One approach involves using specialized diffusion ODE solvers, such as the DDIM solver [Song et al., 2020a] or the DPM-solver [Lu et al., 2022], to speed up iterative sampling. Another strategy is to avoid iterative denoising during training or inference. EDP [Kang et al., 2024] and IDQL [Hansen-Estruch et al., 2023] both focus on avoiding iterative sampling during training. EDP adopts an approximate diffusion sampling scheme to minimize the required sampling steps, although it still requires iterative denoising during inference. IDQL accelerates the training process by only training a behavior cloning policy without denoising sampling. However, it requires iterative sampling during inference by selecting from a batch of candidate generated actions. SRPO [Chen et al., 2023] employs score distillation methods to avoid iterative denoising in both training and inference.

Distillation Methods. Distillation methods for diffusion models have been proposed to enable one-step generation of images or 3D objects. Examples of such methods include SDS [Poole et al., 2022], VSD [Wang et al., 2024], Diff Instruct [Luo et al., 2024], and DMD [Yin et al., 2023]. The core idea of these methods is to minimize the KL divergence between a pre-trained diffusion model and a target one-step generation model. SiD [Zhou et al., 2024] uses a different divergence metric but shares the same goal of mimicking the distribution learned by a pre-trained diffusion model. The distillation strategy can also be applied in the offline RL field to accelerate training and inference. However, directly adopting these methods may result in suboptimal performance.

B Diffusion Schedule

This diffusion training schedule is the same for training the behavior-cloning policy in Equation 2 and the diffusion trust region loss in Equation 4.

Noise Schedule We illustrate the EDM diffusion training schedule in our setting. First, we need to define some prespecified parameters: $\sigma_{\text{data}} = 0.5$, $\sigma_{\text{min}} = 0.002$, $\sigma_{\text{max}} = 80$. The noise schedule is defined by $\boldsymbol{a}_t = \alpha_t \boldsymbol{a} + \sigma_t \boldsymbol{\varepsilon}$, where $\boldsymbol{\varepsilon} \sim \mathcal{N}(0, \boldsymbol{I})$. We set $\alpha_t = 1$ and $\sigma_t = t$. The variable $\log(t)$ follows a logistic distribution with location parameter $\log \sigma_{\text{data}}$ and scale parameter 0.5. The original EDM paper samples $\log(t)$ from $\mathcal{N}(-1.2, 1.2^2)$, but this difference does not significantly affect our algorithm.

Denoiser The denoiser μ_ϕ is defined as:

$$\mu_\phi(\boldsymbol{a}_t, t|\boldsymbol{s}) = c_{\text{skip}}(\sigma)\boldsymbol{a}_t + c_{\text{out}}(\sigma)F_\phi(c_{\text{in}}(\sigma)\boldsymbol{a}_t, c_{\text{noise}}(\sigma)|\boldsymbol{s}),$$

where $\sigma = \sigma_t = t$ and F_ϕ represents the raw neural network layer. We also define:

$$c_{\text{skip}}(\sigma) = \frac{\sigma_{\text{data}}^2}{\sigma^2 + \sigma_{\text{data}}^2}, \;\; c_{\text{out}}(\sigma) = \frac{\sigma \cdot \sigma_{\text{data}}}{\sqrt{\sigma^2 + \sigma_{\text{data}}^2}},$$
$$c_{\text{in}}(\sigma) = \frac{1}{\sigma^2 + \sigma_{\text{data}}^2}, \;\; c_{\text{noise}}(\sigma) = \frac{1}{4}\log(\sigma).$$

Weight Schedule The final loss is given by:

$$\mathbb{E}_{\sigma, \boldsymbol{a}, \boldsymbol{s}, \boldsymbol{\varepsilon}} \left[\lambda(\sigma) c_{\text{out}}^2(\sigma) \left\| F_\phi(c_{\text{in}}(\sigma) \cdot (\boldsymbol{a} + \boldsymbol{\varepsilon}), c_{\text{noise}}(\sigma) | \boldsymbol{s}) - \frac{1}{c_{\text{out}}(\sigma)} (\boldsymbol{a} - c_{\text{skip}}(\sigma) \cdot (\boldsymbol{a} + \boldsymbol{\varepsilon})) \right\|_2^2 \right],$$

where $\lambda(\sigma) = \frac{1}{c_{\text{out}}^2(\sigma)}$.

C Details in KL Behavior Regularization

Here we introduce how we implement KL divergence regularization. The idea is similar to previous KL-based distillation methods [Wang et al., 2024, Luo et al., 2024, Yin et al., 2023], but adapted to our setting. Our loss function is defined as:

$$\mathcal{L}_{\text{KL}}(\theta) = D_{\text{KL}}[\pi_\theta(\cdot|\boldsymbol{s}) || \mu_\phi(\cdot|\boldsymbol{s})] = \mathbb{E}_{\boldsymbol{\varepsilon} \sim \mathcal{N}(0, \boldsymbol{I}), \boldsymbol{s} \sim \mathcal{D}, \pi_\theta(\boldsymbol{s}, \boldsymbol{\varepsilon})} \left[\log \frac{p_{\text{fake}}(\boldsymbol{a}_\theta | \boldsymbol{s})}{p_{\text{real}}(\boldsymbol{a}_\theta | \boldsymbol{s})} \right] \tag{12}$$

The gradient of $\mathcal{L}_{\text{KL}}(\theta)$ is given by:

$$\nabla_\theta \mathcal{L}_{\text{KL}}(\theta) = \mathbb{E}_{\boldsymbol{\varepsilon}, \boldsymbol{s}, \boldsymbol{a}_\theta = \pi_\theta(\boldsymbol{s}, \boldsymbol{\varepsilon})} \left[(s_{\text{fake}}(\boldsymbol{a}_\theta | \boldsymbol{s}) - s_{\text{real}}(\boldsymbol{a}_\theta | \boldsymbol{s})) \nabla_\theta \pi_\theta \right]$$

where $s_{\text{real}}(\boldsymbol{a}_\theta | \boldsymbol{s}) = \nabla_{\boldsymbol{a}_\theta} \log p_{\text{real}}(\boldsymbol{a}_\theta | \boldsymbol{s})$ and $s_{\text{fake}}(\boldsymbol{a}_\theta | \boldsymbol{s}) = \nabla_{\boldsymbol{a}_theta} \log p_{\text{fake}}(\boldsymbol{a}_\theta | \boldsymbol{s})$. By using the Score-ODE given in [Song et al., 2020b], we can estimate $s_{\text{real}}(\boldsymbol{a}_\theta | \boldsymbol{s})$ and $s_{\text{fake}}(\boldsymbol{a}_\theta | \boldsymbol{s})$ with a diffusion model. Let $\boldsymbol{a}_{\theta,t} = \alpha_t \boldsymbol{a}_\theta + \sigma_t \boldsymbol{\varepsilon}$, the real score can be estimated by:

$$s_{\text{real}}(\boldsymbol{a}_{\theta,t}, t | \boldsymbol{s}) = -\frac{\boldsymbol{a}_{\theta,t} - \alpha_t \mu_\phi(\boldsymbol{a}_{\theta,t}, t | \boldsymbol{s})}{\sigma_t^2}$$

where μ_ϕ is the pre-trained diffusion behavior cloning model that learns the true data distribution.

Similarly, we can estimate the fake score by:

$$s_{\text{fake}}(\boldsymbol{a}_{\theta,t}, t | \boldsymbol{s}) = -\frac{\boldsymbol{a}_{\theta,t} - \alpha_t \mu_\xi(\boldsymbol{a}_{\theta,t}, t | \boldsymbol{s})}{\sigma_t^2}$$

where μ_ξ is trained using fake data:

$$\mathcal{L}(\xi) = \|\mu_\xi(\boldsymbol{a}_{\theta,t}, t | \boldsymbol{s}) - \boldsymbol{a}_\theta\|_2^2$$

which is trained with generated fake action data.

Thus, the gradient of $\mathcal{L}_{\text{KL}}(\theta)$ can be expressed as:

$$\nabla_\theta \mathcal{L}_{\text{KL}}(\theta) = \mathbb{E}_{\boldsymbol{\varepsilon}, \boldsymbol{s}, \boldsymbol{a}_\theta, \boldsymbol{a}_{\theta,t}} \left[w_t \alpha_t \left(s_{\text{fake}}(\boldsymbol{a}_{\theta,t}, t | \boldsymbol{s}) - s_{\text{real}}(\boldsymbol{a}_{\theta,t}, t | \boldsymbol{s}) \right) \nabla_\theta \pi_\theta \right]$$

where $w_t = \frac{\sigma_t^2}{\alpha_t} \frac{A}{\|\mu_\phi(\boldsymbol{a}_{\theta,t}, t) - \boldsymbol{a}_\theta\|_1}$ and A is the dimension of the action space.

The algorithm for KL regularization is shown below:

D Implementation Details

Diffusion Policy We build our policy as an MLP-based conditional diffusion model. The model itself is an action prediction model. We model μ_ϕ and μ_ξ as 4-layer MLPs with Mish activations, using 256 hidden units for all networks. The input to μ_ϕ and μ_ξ is the concatenation of the noisy action vector, the current state vector, and the sinusoidal positional embedding of timestep t. The output of μ_ϕ and μ_ξ is the predicted action at diffusion timestep t.

Algorithm 2 KL Regularization

```
Initialize policy network $\pi_\theta$, $\mu_\phi$, $\mu_\xi$
for each iteration do
    Sample transition mini-batch $\mathcal{B} = \{(\boldsymbol{s}_t, \boldsymbol{a}_t, r_t, \boldsymbol{s}_{t+1})\} \sim \mathcal{D}$
    Diffusion Policy Learning: Update $\mu_\phi$ by $\mathcal{L}(\phi)$
end for
Initialize policy and fake score network: $\theta \leftarrow \phi, \xi \leftarrow \phi$
for each iteration do
    Sample transition mini-batch $\mathcal{B} = \{(\boldsymbol{s}_t, \boldsymbol{a}_t, r_t, \boldsymbol{s}_{t+1})\} \sim \mathcal{D}$, generate $\boldsymbol{a}_\theta$
    Random timestep and add noise: Choose $t$, $\boldsymbol{a}_{\theta_t} = \alpha_t \boldsymbol{a}_\theta + \sigma_t \varepsilon$
    with_no_grad():
        $pred\_fake\_action = \mu_\xi(\boldsymbol{a}_{\theta_t}, t|\boldsymbol{s})$
        $pred\_real\_action = \mu_\phi(\boldsymbol{a}_{\theta_t}, t|\boldsymbol{s})$
    $weighting\_factor = \text{abs}(\boldsymbol{a}_\theta - pred\_real\_action)$.mean(keepdim=True)
    $grad = \frac{pred\_fake\_action - pred\_real\_action}{weighting\_factor}$
    $loss = 0.5 \times \text{mse\_loss}(\boldsymbol{a}_\theta, \text{stopgrad}(\boldsymbol{a}_\theta - grad))$
    Update $\pi_\theta$ by $loss$
    Diffusion Fake Policy Learning: Update $\mu_\xi$ by $\mathcal{L}(\xi)$
end for
```

Q and V Networks We build two Q networks and a V network with the same MLP setting as our diffusion policy. Each network comprises 4-layer MLPs with Mish activations and 256 hidden units.

Stochastic Max Q Trick Similar to DQL Wang et al. [2022a], during inference, we generate N candidate actions and then randomly select an action according to $\exp(Q(\boldsymbol{a}, \boldsymbol{s}))$. Here, N is fixed at 1024 and remains unchanged across different tasks.

One-Step Policy We build a Gaussian policy using 3-layer MLPs with ReLU activations, utilizing 256 hidden units. After sampling an action, we apply a tanh activation to ensure the action lies between $[-1, 1]$. If an implicit policy is instantiated, its structure is the same as that of the diffusion policy.

Pretrain In our implementation, we pretrain the diffusion policy μ_ϕ and the Q function Q_η for 50 epochs to ensure they can better guide π_θ. Then, μ_ϕ, Q_η, and π_θ are concurrently trained for the epochs specified in Table 4. We found that introducing a pretrain schedule does not significantly influence the final performance. Our ablation study on the Gym Medium Task revealed that while pretraining yields slightly better results, the final rewards are largely similar. Therefore, we maintain a 50-epoch pretrain for all our tasks. The results are shown in Table 3.

Table 3: The performance with and without pretraining on D4RL Gym tasks.

Environment	Pretrain	No Pretrain
halfcheetah-medium-v2	57.9	57.5
hopper-medium-v2	99.6	87.6
walker2d-medium-v2	89.4	88.7

E Hyperparamaters

Table 4: Hyperparameters for D4RL benchmarks. One epoch represents 1k steps, and the optimizer used is Adam.

Gym	α	τ	**NLL Term**	**Pretrain Epochs**	**Training Epochs**	**Learning Rate**	**Lr decay**
halfcheetah-medium-v2	1	0.7	False	50	1000	3×10^{-4}	False
halfcheetah-medium-replay-v2	5	0.7	False	50	1000	3×10^{-4}	False
halfcheetah-medium-expert-v2	50	0.7	False	50	1000	3×10^{-4}	False
hopper-medium-v2	5	0.7	False	50	1000	1×10^{-4}	True
hopper-medium-replay-v2	5	0.7	False	50	1000	3×10^{-4}	False
hopper-medium-expert-v2	20	0.7	False	50	1000	3×10^{-4}	False
walker2d-medium-v2	5	0.7	False	50	1000	3×10^{-4}	True
walker2d-medium-replay-v2	5	0.7	False	50	1000	3×10^{-4}	True
walker2d-medium-expert-v2	5	0.7	False	50	1000	3×10^{-4}	True
antmaze-umaze-v0	1	0.9	True	50	500	3×10^{-4}	False
antmaze-umaze-diverse-v0	1	0.9	True	50	500	3×10^{-5}	True
antmaze-medium-play-v0	1	0.9	True	50	400	3×10^{-4}	False
antmaze-medium-diverse-v0	1	0.9	True	50	400	3×10^{-4}	False
antmaze-large-play-v0	1	0.9	True	50	350	3×10^{-4}	False
antmaze-large-diverse-v0	0.5	0.9	True	50	300	3×10^{-4}	False
antmaze-umaze-v2	1	0.9	True	50	500	3×10^{-4}	False
antmaze-umaze-diverse-v2	1	0.9	True	50	500	3×10^{-5}	True
antmaze-medium-play-v2	1	0.9	True	50	500	3×10^{-4}	False
antmaze-medium-diverse-v2	1	0.9	True	50	500	3×10^{-4}	False
antmaze-large-play-v2	1	0.9	True	50	500	3×10^{-4}	False
antmaze-large-diverse-v2	0.5	0.9	True	50	500	3×10^{-4}	False
pen-human-v1	1500	0.9	False	50	300	3×10^{-5}	True
pen-cloned-v1	1500	0.7	False	50	200	1×10^{-5}	False
kitchen-complete-v0	200	0.7	False	50	500	1×10^{-4}	True
kitchen-partial-v0	100	0.7	False	50	1000	1×10^{-4}	True
kitchen-mixed-v0	200	0.7	False	50	500	3×10^{-4}	True

F Additional Experiments

F.1 Complete 2D Toy Experiments

We also conducted some 2D bandit experiments with different reward scenarios. In Figure 6, red points are generated by the one-step policy π_θ.

In the first column, where the four corners have the same high reward, $\mathcal{L}_{\text{KL}}$ tends to encourage exploration of all these high-reward regions, resulting in some suboptimal reward actions. In contrast, $\mathcal{L}_{\text{TR}}$ generates actions that randomly select one of the high-reward regions, thereby avoiding suboptimal actions. The same situation occurs in the fourth and fifth columns of Figure 6, where $\mathcal{L}_{\text{KL}}$ covers some suboptimal regions while $\mathcal{L}_{\text{TR}}$ adheres closely to the highest reward regions.

However, when the data have only one mode with the highest reward, such as in the second and third columns of Figure 6, both $\mathcal{L}_{\text{KL}}$ and $\mathcal{L}_{\text{TR}}$ guide the policy to generate high-reward actions.

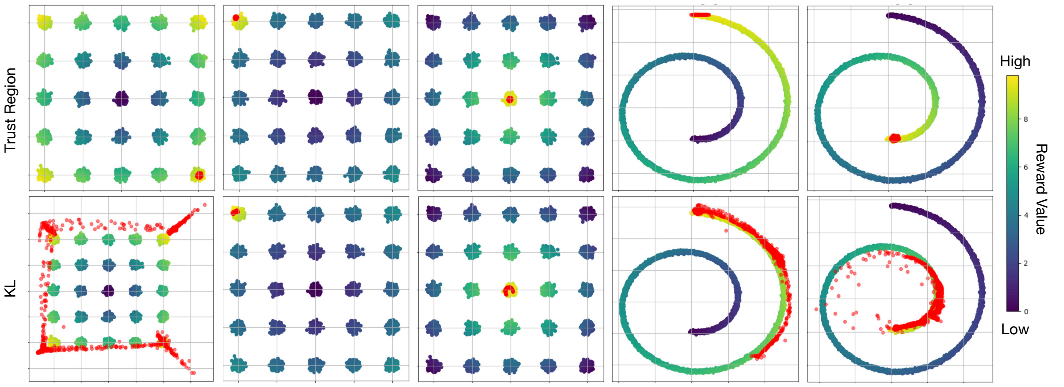

Figure 6: 2D Bandit toy examples, where the behavior regularization is conducted by $\mathcal{L}_{\text{TR}}$ and $\mathcal{L}_{\text{KL}}$ in different behavior data and reward scenarios. The first row uses behavior regularization by $\mathcal{L}_{\text{TR}}$, and the second row uses $\mathcal{L}_{\text{KL}}$. Yellow indicates the highest reward, and dark blue indicates the lowest reward.

F.2 Comparison with KL behavior Regularization in Gym Tasks

In addition to testing on 2D bandit scenarios, we also evaluated the performance of two losses $\mathcal{L}_{\text{KL}}$ and $\mathcal{L}_{\text{TR}}$ on the Mujoco Gym Medium task. The behavior regularization loss $\mathcal{L}_{\text{TR}}(\theta)$ consistently outperformed $\mathcal{L}_{\text{KL}}(\theta)$ in terms of achieving higher rewards. The results are presented in Table 5, and the training curves are depicted in Figure 8.

Table 5: The performance of $\mathcal{L}_{\text{TR}}(\theta)$ and $\mathcal{L}_{\text{KL}}(\theta)$ on D4RL Gym tasks. Results correspond to the mean of normalized scores over 50 random rollouts (5 independently trained models and 10 trajectories per model).

Environment	$\mathcal{L}_{\text{TR}}(\theta)$	$\mathcal{L}_{\text{KL}}(\theta)$
halfcheetah-medium-v2	**57.9**	24.1
hopper-medium-v2	**99.6**	15.0
walker2d-medium-v2	**89.4**	3.4

F.3 Comparison with SRPO on Antmaze-v2 Datasets

Since SRPO uses Antmaze-v2 for their D4RL benchmarks, we also conducted experiments on Antmaze-v2 using our algorithm, with the same hyperparameters as those used in Antmaze-v0 but with more training epochs. Hyperparameters details can be found in Table 4. The results for Antmaze-v2 from SRPO are taken directly from their paper.

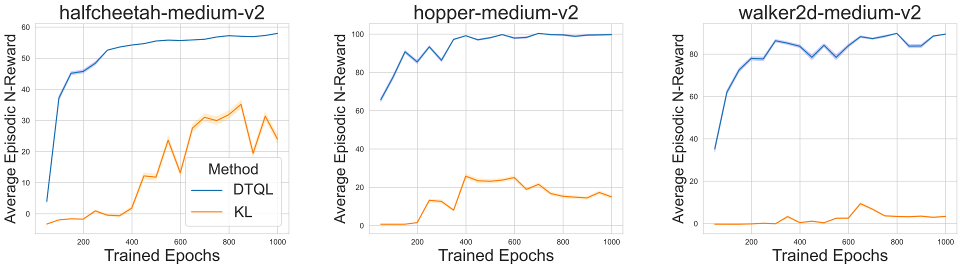

Figure 8: Training curves comparing policy learning with diffusion trust region loss and KL loss across three Gym medium tasks demonstrate that diffusion trust region regularization (DTQL) consistently outperforms KL-based behavior regularization in policy learning.

The results for Antmaze-v2 are shown in Table 6. Our observations indicate that, on average, our method achieves a higher score and exhibits significant performance improvements in complex Antmaze tasks, such as *antmaze-medium-diverse*, *antmaze-large-play*, and *antmaze-large-diverse*.

Table 6: The performance of Our methods and SOTA baselines on D4RL AntMaze-v2 tasks. Results for DTQL correspond to the mean and standard errors of normalized scores over 500 random rollouts.

Antmaze	**SRPO**	**Ours**
antmaze-umaze-v2	97.1	92.6±1.24
antmaze-umaze-diverse-v2	82.1	74.4±1.95
antmaze-medium-play-v2	80.7	76±1.91
antmaze-medium-diverse-v2	75.0	**80.6**±1.77
antmaze-large-play-v2	53.6	**59.2**±2.19
antmaze-large-diverse-v2	53.6	**62**±2.17
Average	73.6	**74.1**

F.4 Overall Training and Inference Time

In Table 7, we show the total training and inference wall time recorded on 8 RTX-A5000 GPU servers, which include all training epochs specified in Table 4 and the entire evaluation process. For evaluation, we test 10 trajectories for gym tasks and 100 trajectories for all other tasks.

Table 7: Total training and inference wall time for D4RL benchmarks

Tasks	Overall Training and Inference Time	Training Epochs
halfcheetah-medium-v2	5.1h	1000
halfcheetah-medium-replay-v2	5.1h	1000
halfcheetah-medium-expert-v2	5.5h	1000
hopper-medium-v2	5.0h	1000
hopper-medium-replay-v2	5.4h	1000
hopper-medium-expert-v2	5.2h	1000
walker2d-medium-v2	4.9h	1000
walker2d-medium-replay-v2	4.9h	1000
walker2d-medium-expert-v2	4.9h	1000
antmaze-umaze-v0	3.3h	500
antmaze-umaze-diverse-v0	4.0h	500
antmaze-medium-play-v0	3.1h	400
antmaze-medium-diverse-v0	3.2h	400
antmaze-large-play-v0	2.3h	350
antmaze-large-diverse-v0	2.6h	300
antmaze-umaze-v2	3.3h	500
antmaze-umaze-diverse-v2	3.1h	500
antmaze-medium-play-v2	3.1h	500
antmaze-medium-diverse-v2	3.1h	500
antmaze-large-play-v2	3.3h	500
antmaze-large-diverse-v2	3.3h	500
pen-human-v1	1.4h	300
pen-cloned-v1	0.6h	200
kitchen-complete-v0	3.0h	500
kitchen-partial-v0	6.1h	1000
kitchen-mixed-v0	3.0h	500

G Training Curves

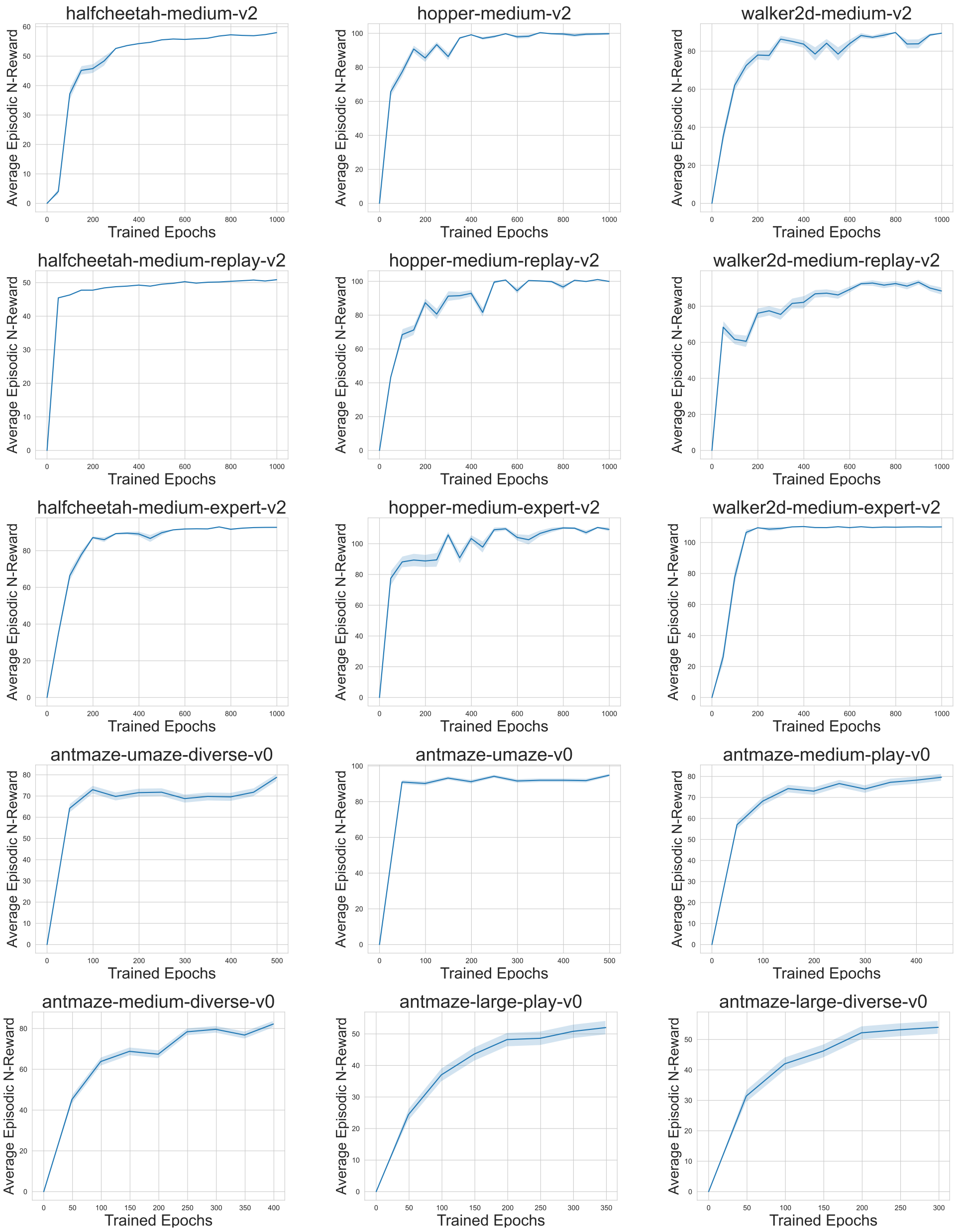

Figure 9: Training curves. Rewards evaluated after every 50 epochs.

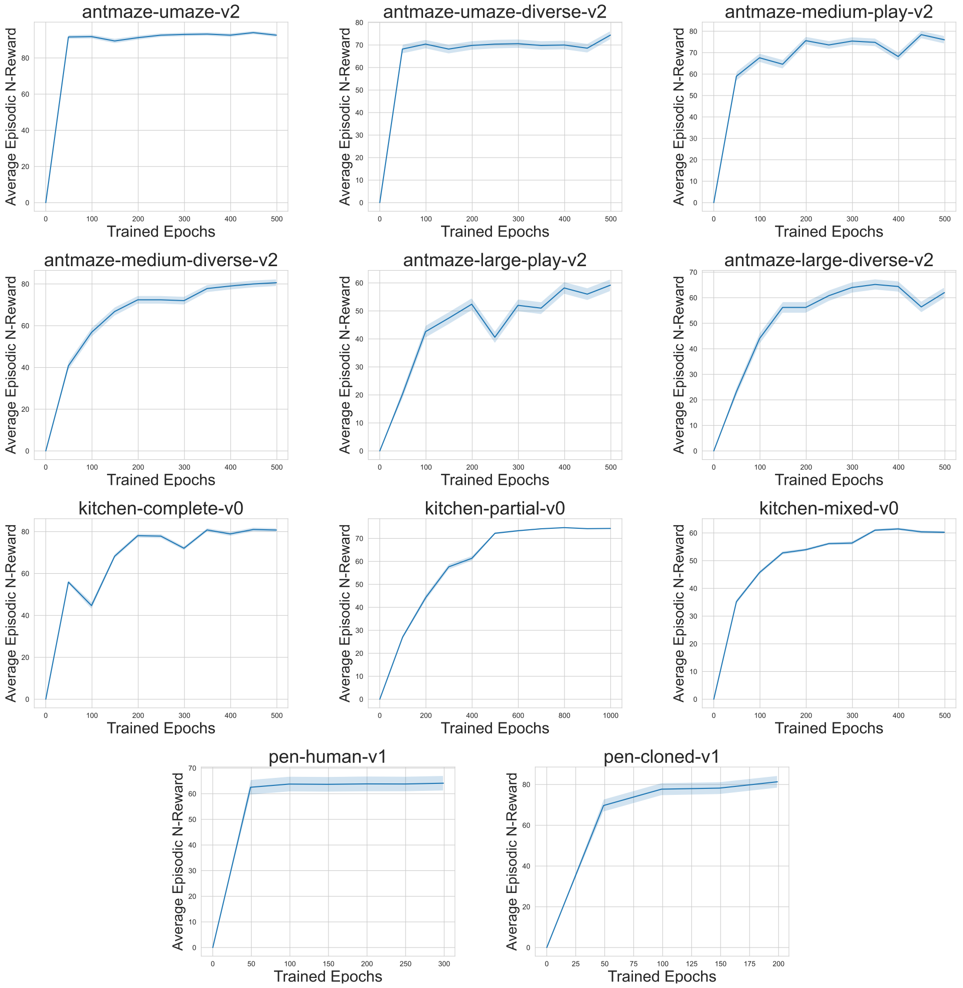

Figure 10: Training curves. Rewards evaluated after every 50 epochs.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Our contribution is accurately reflected in abstract and introduction.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: It has been discuss in Section 5.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: All theorems are proven in main text or by reference.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The algorithms are given in Algorithm 1 and Appendix C. Implement details are discuss in Appendix D and hyperparameters are discussed in E.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The code has already been published in `https://github.com/TianyuCodings/Diffusion_Trusted_Q_Learning`.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: Expriments details are all contained in Appendices B to F.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Standard error is provided in Table 1 and also plotted in training curves, Appendix G.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Computer resources have been discussed in Section 4.2.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: No societal impact of the work performed.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: We are using simulation dataset D4RL which has no such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We are using D4RL datasets and it is explicitly cited.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [No]

    Justification: The primary new asset is the code of our algorithm. It is not published yet, but will be made public once the paper is accepted.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: No human subjects are involved in this research.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.