Leveraging Score-based Models for Generating Penalization in Model-based Offline Reinforcement Learning

Zeyuan Liu*
Tsinghua Shenzhen International Graduate School, Tsinghua University
Shenzhen, China
gritmaybe@gmail.com

Zhirui Fang*
Tsinghua Shenzhen International Graduate School, Tsinghua University
Shenzhen, China
fzr23@mails.tsinghua.edu.cn,

Jiafei Lyu
Tsinghua Shenzhen International Graduate School, Tsinghua University
Shenzhen, China
lvjf20@mails.tsinghua.edu.cn

Xiu Li†
Tsinghua Shenzhen International Graduate School, Tsinghua University
Shenzhen, China
li.xiu@sz.tsinghua.edu.cn

ABSTRACT

A core challenge in model-based offline reinforcement learning is constructing penalties over the state-action space of the offline dataset, which is typically high-dimensional. We define "cliffs" as regions in the state-action space where data density changes sharply, and our investigation shows that existing approaches struggle with accuracy near these cliffs. The formation of cliffs could be influenced by human-defined parameters and objective physical laws, often beyond the understanding of RL agents. This results in a lack of established methods to address this issue. To overcome these limitations, we propose Score as a Penalty for Model-based Offline Reinforcement Learning (ScorePen-MORL). This innovative approach generates penalties based on the gradient filed of dataset density in the state-action space. ScorePen-MORL is a plug-and-play solution that can achieve impressive results independently while also enhancing the performance of baseline algorithms through the joint effect. Our empirical findings demonstrate that cliff regions in the dataset are a significant bottleneck in offline model-based RL, and ScorePen-MORL effectively addresses this issue by generating highly sensitive penalties for these cliff regions. Through the empirical results on the D4RL and NeoRL benchmarks, we find our method outperforms recent strong model-based offline RL baseline algorithms.

KEYWORDS

Score-based Model, Diffusion Model, Offline RL, Model-based RL

ACM Reference Format:
Zeyuan Liu*, Zhirui Fang*, Jiafei Lyu, and Xiu Li†. 2025. Leveraging Score-based Models for Generating Penalization in Model-based Offline Reinforcement Learning. In *Proc. of the 24th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2025), Detroit, Michigan, USA, May 19 – 23, 2025*, IFAAMAS, 10 pages.

*: Equal contribution. †: Corresponding authors.

Proc. of the 24th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2025), Y. Vorobeychik, S. Das, A. Nowé (eds.), May 19 – 23, 2025, Detroit, Michigan, USA. © 2025 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org).

1 INTRODUCTION

Offline reinforcement learning (RL), or batch RL, enables agents to learn from pre-collected datasets, thus avoiding the need for costly and potentially hazardous online exploration. This is especially beneficial in safety-critical fields like robotics [15, 41, 42] and autonomous driving [19], where online trial-and-error is impractical. By using static datasets, offline RL ensures controlled training and reduces risks associated with online data collection.

However, this approach poses significant challenges, with distribution shifts being a primary concern [5, 22, 28, 33, 58**?**]. Distribution shifts refer to the situation where the agent encounters state-action pairs that substantially deviate from the training data, potentially propagating inaccurate or extreme value estimates. This can lead to poor policy evaluation and suboptimal performance [5, 22]. Addressing distribution shifts is essential for enhancing the robustness and generalization of offline RL algorithms, ensuring their effectiveness in dynamic real-world environments. Solutions to this issue in offline RL typically fall into model-free [23, 38] and model-based categories [16, 27, 34, 36, 57]. Model-free approaches often incorporate conservatism by penalizing value functions for out-of-distribution (OOD) actions or constraining the learned policy to remain close to the behavior policy, thereby improving stability and robustness in learning [2–4, 21–23, 49, 53, 54].

While model-free approaches are constrained to the data within the offline dataset, model-based offline reinforcement learning aims to improve data efficiency and generalization by employing learned dynamic models to generate synthetic data. This synthetic data enables the exploration of states not represented in the original dataset, thereby expanding the available state space and potentially enhancing the policy's performance. However, model-based offline RL introduces new technical challenges and reveals gaps in existing theoretical frameworks. The introduction of learned models entails the risk of errors, making conservatism essential. Recent research addresses this issue by incorporating penalties based on metrics such as the total variation distance between learned and ground truth models. Yet, accurately computing this distance remains challenging, prompting alternative strategies like regularizing value functions or adversarially modifying transition dynamics without directly quantifying model uncertainty. Techniques such

as MOPO [57] utilize aleatoric uncertainty derived from state predictions to penalize rewards, and data density estimates have been explored as a basis for conservatism [32]. Despite these advances, model-based offline RL continues to face technical challenges, with recent theoretical insights suggesting that measures like the Bellman error may not fully capture ideal uncertainty [45], potentially leading to less optimal performance compared to some model-free methods.

The complexities of model-based offline reinforcement learning remain unresolved, with many challenges still the subject of ongoing research. For example, the "edge-of-reach" issue identified by RAVL [40] illustrates a scenario where value overestimation occurs for states that are only reachable at the final steps of rollouts in the learned dynamics model, underscoring the complexities involved in ensuring reliable policy performance. This provides insight that many of the bottlenecks and challenges in model-based offline RL may reside in subtle, hard-to-detect areas.

In this paper, we begin by identifying a critical challenge in model-based offline RL, referred to as the “cliff regions” issue. This issue exposes the limitations of existing penalty mechanisms, which often depend on uncertainty measures and data density estimates, particularly in regions of the offline dataset where data density undergoes abrupt changes. We observe that existing methods generate penalties near these regions that fail to accurately reflect uncertainty, resulting in ineffective utilization of the knowledge in these regions. This issue is exacerbated when critical information for optimizing policies is located in or near such regions, such as the endpoints for a navigation task, leading to poor performance in addressing the task. Furthermore, cliff regions pose significant analytical challenges; they are difficult to locate within the dataset due to the unknown ground truth density function of the original data distribution. Additionally, analyzing the diverse and unpredictable origins of cliff regions, such as physical constraints like joint limits in robotic systems, is challenging.

To address the "cliff regions" issue, our work leverages information from the dataset’s gradient field by reconstructing it with a score-based diffusion model and generating penalties based on the model’s score outputs. We introduce a theoretical analysis demonstrating the effectiveness of using scores as penalties to tackle the cliff regions challenge, thereby expanding the current discourse on the limitations of existing model-based offline reinforcement learning (RL) approaches. Utilizing a fundamental property of score-based diffusion models, which produce larger gradients for out-of-distribution samples, we also investigate the exclusive use of scores for penalty generation during training. Empirical results show that we achieve competitive performance compared to a wide range of methods that overlook the cliff regions issue, including those utilizing conservatism [55], model uncertainty [13, 36], and complexities of data distribution in synthetic rollouts [22, 52] for penalty derivation. This work aims to enhance the robustness of model-based offline RL methodologies, better equipping them to address the challenges posed by dynamic environments and diverse task requirements. The overall structure of our method refers to Figure 1.

Our contributions are summarized as follows: We identify a previously unrecognized issue in model-based offline RL, termed the "cliff regions" issue, and demonstrate its presence and impact on existing methods through simple experiments. To address this challenge, we propose Score as a Penalty for Model-based Offline RL (ScorePen-MORL), which leverages a score-based diffusion model to generate penalties based on the gradient field of data density, effectively mitigating the difficulties posed by cliff regions and enhancing the robustness of the learning process. We extensively evaluate ScorePen-MORL on multiple benchmarks, including D4RL and NeoRL, and provide a detailed analysis of how the algorithm formulates penalties that encourage conservatism. Our results demonstrate that ScorePen alleviates cliff region issues and improves policy performance in model-based offline RL. Additionally, ScorePen-MORL is a plug-and-play solution that can be seamlessly integrated with other baseline algorithms, offering further performance improvements.

2 RELATED WORKS

2.1 Offline RL

Offline (or batch) reinforcement learning (RL) aims to derive effective policies from a fixed dataset, which is gathered by unknown behavior policies. A key challenge in this area is the distribution shift, where agents often overestimate and favor out-of-distribution (OOD) actions, resulting in diminished performance [5, 29]. To tackle this issue, various strategies have been developed. These include constraining the learned policy to remain close to the original behavior policy [4, 5, 21, 25, 39, 52], regularizing the critic to adopt a more pessimistic outlook through ensemble techniques for OOD actions [1, 2, 22], and implementing model-based approaches that utilize uncertainty measurements. Importance sampling [6, 24, 47] techniques also play a vital role in alleviating the adverse effects of distribution shifts.

Particularly noteworthy are methods that learn density models of the training data to constrain the agent’s behavior within the data distribution [5, 6, 18, 22, 30, 35]. Additionally, some approaches utilize the gradient field of the dataset to guide corrections in the offline RL agent’s actions [26]. Building on these insights, our work introduces a novel solution that leverages the score-based diffusion model and the knowledge from the gradient field of the offline dataset to address the cliff regions issue, offering greater flexibility in tackling bootstrapping errors and providing a more robust framework for policy learning in offline RL. This approach enables more precise value function approximations, enhancing overall performance in offline RL tasks.

2.2 Model-based Offline RL

In the domain of model-based offline reinforcement learning (RL), recent advancements have increasingly emphasized the integration of penalization strategies designed to address model estimation errors [16, 34, 57]. A notable example is the Method of Penalization for Offline Policy Optimization (MOPO) [57], which implements a penalization mechanism that adjusts the reward function in accordance with the maximum aleatoric uncertainty arising from discrepancies between the true and estimated models. This approach effectively aims to temper overoptimistic estimates that could lead to suboptimal policy decisions.

The Model-based Reinforcement Learning (MOReL) framework [16] employs a pessimistic strategy using an unknown state-action

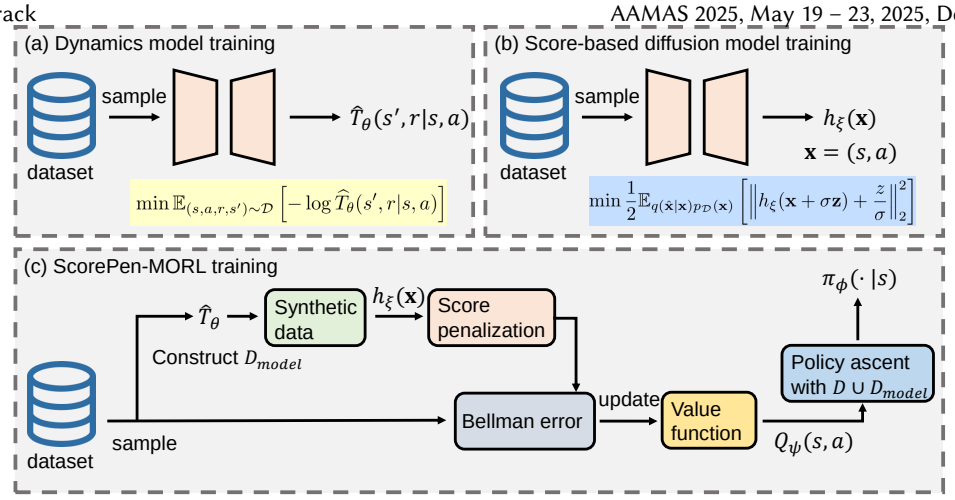


Figure 1: The overall structure of ScorePen-MORL involves training an ensemble of dynamics models to generate synthetic data and a score-based diffusion model to reconstruct the gradient field of the offline dataset. During value function optimization, the scores produced by the diffusion model are used to penalize the targets associated with the synthetic data.

detector to set a threshold that mitigates risks from model uncertainty, highlighting the importance of conservative estimates in uncertain environments. LOMPO [34] extends this approach by incorporating latent dynamics models for image data and quantifying uncertainty through log-likelihood variance. COMBO [56] adopts a Dyna-style approach, applying CQL to enforce low Q-values on out-of-distribution (OOD) samples generated by the dynamics model. RAMBO [37] incorporates conservatism by adversarially training the dynamics model to minimize the value function while ensuring accurate transition predictions. Concurrently proposed with our method, CBOP [14] utilizes the model-based value expansion (MVE) framework, adapting the weighting of h-step returns and employing value variance across a model ensemble for conservative estimation. The latest methods, including Count-MORL [17], which enhances model-based offline RL by introducing count-based conservatism through state-action pair counts to quantify estimation error, and MOBILE [46] which penalize Bellman estimation inconsistencies using uncertainty from a model ensemble.

2.3 Score-based Model

Recently, score-based generative models [8–10, 43, 44, 48] have garnered significant attention in the field of machine learning. The core principle behind these models is the representation of real data distributions through a score function, which constitutes a vector field indicating the direction of the greatest probability increase for the data. By utilizing the learned score function as a prior, we can employ Langevin Markov Chain Monte Carlo (MCMC) [50] sampling techniques to generate high-quality data from random noise. Score-based models have demonstrated remarkable success across various modalities, including images [12, 44], audio [20], and graphs [31].

In reinforcement learning (RL), various approaches [51] have integrated score-based models to tackle challenges like object rearrangement. These methods typically aim to increase the likelihood of states within the original distribution but often require joint training with baseline algorithms, complicating implementation and limiting scalability. DiffCPS [11] introduces diffusion models to offline RL by addressing constrained policy search with a primal-dual approach, approximating policy solutions through dual iterations to achieve competitive performance. In contrast, our method, ScorePen-MORL, incorporates score-based diffusion models into the training of model-based offline RL, functioning as a plug-and-play solution. The penalties generated by ScorePen-MORL can be applied independently or in conjunction with other baseline algorithms, enhancing their performance.

3 PRELIMINARIES

3.1 Markov Decision Process

We consider a Markov decision process (MDP), defined by the tuple $M = (\mathcal{S}, \mathcal{A}, P, r, d_0, \gamma)$, where $\mathcal{S}$ and $\mathcal{A}$ denote the state and action spaces, respectively. The transition dynamics are described by $P : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$, and the reward function is $r : \mathcal{S} \times \mathcal{A} \to [-R_{\max}, R_{\max}]$. Here, $d_0 \in \Delta(\mathcal{S})$ represents the initial state distribution, and $\gamma \in [0, 1)$ is the discount factor. The policy $\pi : \mathcal{S} \to \Delta(\mathcal{A})$ defines a probability distribution over actions for each state.

The value function $V^\pi_{P,r}(s)$ is expressed as:

$$V^\pi_{P,r}(s) := \mathrm{E}_{\pi,P}\left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) \mid s_0 = s\right], \tag{1}$$

which captures the expected cumulative discounted reward when starting from state s and following policy π, under the dynamics governed by P and the reward function r.

To describe the discounted state visitation distribution under policy π and transition dynamics P, we define:

$$d^\pi_P(s) := (1 - \gamma) \sum_{t=0}^{\infty} \gamma^t \Pr(s_t = s \mid \pi, P), \tag{2}$$

which gives the discounted probability of being in state s at any time step t. Similarly, the discounted state-action visitation distribution is given by:

$$d^\pi_P(s, a) := d^\pi_P(s)\pi(a \mid s), \tag{3}$$

which represents the joint distribution of visiting state s and taking action a under policy π and transition dynamics P.

3.2 RL and Offline RL

The objective of reinforcement learning (RL) is to learn a policy $\pi(a|s)$ that maximizes the expected cumulative return under d_0:

$$\max_{\pi} V_M^{\pi} := \mathbb{E}_{s\sim d_0}\left[V_{P,r}^{\pi}(s)\right] = \frac{1}{1-\gamma}\mathbb{E}_{(s,a)\sim d_P^{\pi}}[r(s,a)]. \quad (4)$$

In offline reinforcement learning (RL), the objective is to learn a policy based solely on a pre-collected, static dataset without further interaction with the environment. The agent is provided with the offline dataset:

$$\mathcal{D} = \left\{\left(s_i, a_i, r_i, s_i'\right)\right\}_{i=1}^{n}, \quad (5)$$

which contains transition tuples gathered by a behavior policy π_β. The specifics of this policy is usually unknown. A major challenge in offline RL arises from the fact that $\mathcal{D}$ only covers a limited subset of the full state-action space. This limitation becomes particularly problematic when critical transitions or actions that the optimal policy depends on are missing from the dataset. As such, offline RL aims to design algorithms capable of learning a policy $\hat{\pi}$ from this fixed dataset, with the goal of minimizing the sub-optimality gap, represented as $V_M^{\pi^*} - V_M^{\hat{\pi}}$, where the goal is to make $\hat{\pi}$ approximate the optimal policy π^* as closely as possible.

3.3 Model-based Offline RL

In model-based offline reinforcement learning (RL), the objective is to derive an optimal policy by leveraging a learned dynamics model. Given an offline dataset $\mathcal{D}$, the dynamics model $\widehat{T}$ is typically trained using maximum likelihood estimation (MLE), minimizing the negative log-likelihood of state transitions in the dataset as follows:

$$\min_{\widehat{T}} \mathbb{E}_{(s,a,s')\sim\mathcal{D}}\left[-\log \widehat{T}(s' \mid s, a)\right]. \quad (6)$$

Throughout this process, the reward function $r(s,a)$ is assumed to be known or well learned. With the trained model $\widehat{T}$, we can construct an estimated MDP, denoted as $\widehat{\mathcal{M}}$. Any reinforcement learning or planning algorithm can then be applied to $\widehat{\mathcal{M}}$ to derive the optimal policy. However, since the dataset $\mathcal{D}$ only covers a limited portion of the state-action space, the learned model may be inaccurate for unobserved state-action pairs, making the policy vulnerable to model exploitation. To address this issue, conservative model-based methods like MOPO [57] and MOReL [16] introduce uncertainty-aware optimization, penalizing policies for taking actions in regions where the model is uncertain. These methods aim to optimize a lower bound on policy performance by incorporating uncertainty estimates into the reward function, thereby discouraging the policy from exploiting unreliable model predictions.

Despite the use of synthetic rollouts, model-based offline RL methods must contend with the challenge of model inaccuracies. A common solution is to incorporate conservatism, as seen in uncertainty-penalized methods that adjust the reward by subtracting a penalty proportional to model uncertainty. MOPO [57] propose a penalized reward function of the form:

$$\tilde{r}(s,a) = r(s,a) - \lambda u(s,a), \quad (7)$$

where $u(s,a)$ represents the estimated model uncertainty for the state-action pair (s,a). The policy is then optimized to maximize the cumulative penalized rewards:

$$\mathbb{E}_{s\sim d_0}\left[V_{\widehat{P},\tilde{r}}^{\pi}(s)\right] = \frac{1}{1-\gamma}\mathbb{E}_{(s,a)\sim d_{\widehat{P}}^{\pi}}\left[r(s,a) - \lambda u(s,a)\right]. \quad (8)$$

However, the primary challenge in such approaches, including those of MOPO [57], Morel [16], and COMBO [55], is the difficulty of obtaining a reliable uncertainty estimate $u(s,a)$, which accurately reflects the model's estimation error for unobserved state-action pairs in the offline dataset.

4 METHOD

In this section, we propose our method, Score as a Penalty for Model-based Offline Reinforcement Learning (ScorePen-MORL). First, we describe the training process of our score-based diffusion model. Next, we provide the motivation and theoretical analysis for using the score as a penalty in model-based offline RL. Finally, we explain how the penalty is applied during training and provide the complete structure including pseudocode for ScorePen-MORL.

4.1 Training Score-based Diffusion Model to Learn Offline Dataset Gradient Field

Before delving into the core details of our method, we first describe how a score-based diffusion model is used to approximate the gradient field of the dataset, $\nabla_x \log p_{\mathcal{D}}(x)$, where $p_{\mathcal{D}}(x)$ denotes the probability distribution of state-action pairs $x = (s,a)$ in the offline dataset $\mathcal{D}$. Utilizing the denoising score-matching model, this can be achieved through minimizing the following loss:

$$\mathcal{L}_\theta = \frac{1}{2}\mathbb{E}_{q(\tilde{\mathbf{x}}|\mathbf{x})p_{\mathcal{D}}(\mathbf{x})}\left[\left\|h_\xi(\tilde{\mathbf{x}}) - \nabla_{\tilde{x}}\log q(\tilde{\mathbf{x}} \mid \mathbf{x})\right\|_2^2\right]. \quad (9)$$

Here, $\tilde{\mathbf{x}}$ represents the noisy state-action pair, which follows the forward transition distribution in the diffusion model:

$$q(\tilde{\mathbf{x}} \mid \mathbf{x}) := \mathcal{N}(\tilde{x} \mid \alpha_t x, \sigma_t^2 I),$$

where α_t and σ_t are pre-defined noise schedules. However, this loss is hard to optimize in practive. Thanks to the score-based diffusion model, We can then rewrite the loss function:

$$\mathcal{L}_\theta = \frac{1}{2}\mathbb{E}_{q(\tilde{\mathbf{x}}|\mathbf{x})p_{\mathcal{D}}(\mathbf{x})}\left[\left\|h_\xi(\mathbf{x}+\sigma\mathbf{z}) + \frac{z}{\sigma}\right\|_2^2\right], \quad (10)$$

where $z \sim N(0,I)$. When the loss converges, we can sample the gradient $\nabla_x \log p_{\mathcal{D}}(\mathbf{x}) \approx h_\xi(\mathbf{x})$ from the learned score function $h^*(x)$ by solving the diffusion ordinary differential equations (ODEs) / stochastic differential equations (SDEs).

4.2 Motivation and Theoretical Analysis of Using Score as Penalty in Model-based Offline RL

Methods like Count-MORL [18] have shown that the estimation error between the estimated and true transition dynamics is smaller for state-action pairs more frequently observed in the offline dataset $\mathcal{D}$, providing an upper bound on the convergence rate based on frequency. We introduce Theorem 1 and Corollary 1 of the Count-MoRL [18] as our Lemma 4.1 and 4.2:

Lemma 4.1 (Theorem 1 in [18]). *Fix $\delta \in (0,1)$, assume $|M| < \infty$ and $P^* \in M$. Given a state-action pair (s,a) is observed in D with*

$D_{s,a} = \{(s_i, a_i, s'_i)\}_{s_i=s,a_i=a}$ and $n(s,a) = |D_{s,a}|$. Define the MLE of transition dynamics as

$$\widehat{P}(\cdot \mid s,a) \in \arg\max_{P \in \mathcal{M}} \sum_{(s,a,s') \in \mathcal{D}_{s,a}} \log P\left(s' \mid s,a\right) \tag{11}$$

for a given (s,a). Then, with probability at least $1-\delta$,

$$\mathrm{TV}\left(\widehat{P}(\cdot \mid s,a), P^{\star}(\cdot \mid s,a)\right) \leq \sqrt{\frac{2\log(|\mathcal{M}|/\delta)}{n(s,a)}} \tag{12}$$

Lemma 4.2 (Corollary 1 in [18]). *Given a state-action pair $(s,a) \in S \times A$, with probability at least $1-\delta$, the estimated transition dynamics $\hat{P}$ satisfies the following inequality:*

$$\mathrm{TV}\left(\widehat{P}(\cdot \mid s,a), P^{\star}(\cdot \mid s,a)\right) \leq C^{\delta}_{\widehat{P}}(s,a) \tag{13}$$

where $C^{\delta}_{\widehat{P}}(s,a) := \min\left(1, \sqrt{\frac{2\log(|\mathcal{M}|/\delta)}{n(s,a)}}\right)$.

By focusing on the dataset's gradient field rather than data density, we aim to reformulate this lemma to express the relationship between the estimation error and the score $\nabla_x \log p_{\mathcal{D}}(x)$, which directly captures the gradient field information. We then use the true dataset density function to approximate the $p(s,a)$, which is the original distribution the dataset $D_{s,a}$ is sampled from. We can have Definition 4.3 similar to the Definition 1 in Count-MORL [18]:

Definition 4.3. Given true density function $p(s,a)$ and the volume of the whole dataset $|D_{s,a}|$, we define the estimation error bound $\widehat{C}^{\delta}_{\widehat{P}} : \mathcal{S} \times \mathcal{A} \to [0,1]$ based on the true dataset density:

$$\widehat{C}^{\delta}_{\widehat{P}}(s,a) := \min\left(1, \sqrt{\frac{2\log(|\mathcal{M}|/\delta)}{p(s,a) \cdot |D_{s,a}|}}\right) \tag{14}$$

The approximation error can be defined as follows:

Definition 4.4. Define the maximal approximation error between estimation error bounds based on the true count and approximate count over all state-action pairs as

$$\epsilon := \sup_{(s,a) \in \mathcal{S} \times \mathcal{A}} \left| C^{\delta}_{\widehat{P}}(s,a) - \widehat{C}^{\delta}_{\widehat{P}}(s,a) \right|. \tag{15}$$

The following lemma shows the gap of returns between the estimated MDP $\hat{M}$ and the true MDP M^* of any given policy π:

Lemma 4.5 (Lemma 1 in [18]). *Suppose M^* and $\hat{M}$ are two MDPs with the true transition dynamics P^* and estimated transition dynamics $\hat{P}$, respectively. Given the estimation error bound based on the approximate count and the maximal approximation error ϵ. Then, with probability at least $1-\delta$, for any policy π,*

$$V^{\pi}_{\hat{M}} - V^{\pi}_{M^*} \leq \frac{\gamma R_{\max}}{(1-\gamma)^2} \mathbb{E}_{(s,a) \sim d^{\pi}_{\hat{P}}} \left[\hat{C}^{\delta}_{\hat{P}}(s,a) \right] + \frac{\gamma R_{\max}}{(1-\gamma)^2} \epsilon \tag{16}$$

For the term $\widehat{C}^{\delta}_{\widehat{P}}$, after training the score-based diffusion network $s_\theta(s,a)$, for any $x = (s,a)$ in the entire state-action space of the offline dataset, we can use Langevin dynamics to sample a new point $x' = (s', a')$ with an expected higher dataset density:

$$x' = x + \epsilon \cdot h_{\xi}(x), \tag{17}$$

where ϵ is a constant. If $p_{\mathcal{D}}(x') \geq C_p$, where C_p is a constant such that $0 < C_p < 1$, we establish the following theorem through a straightforward derivation:

Theorem 4.6. *Given $x = (s,a)$ and $x' = (s', a')$ sampled from x using (9), given $p(x') \geq C_p$, assume any order derivative of assume $p(x)$ exists and is continuous and bounded, we have*

$$\log\left(\sqrt{\frac{2\log(|\mathcal{M}|/\delta)}{p(s,a) \cdot |D_{s,a|}|}}\right) \leq \frac{1}{2} \cdot \varepsilon \cdot ||\nabla_x \log p(s,a)||_2^2 + C, \tag{18}$$

where $C = -\frac{1}{2}\log C_p + \frac{1}{2} \cdot \varepsilon^n \cdot C_R + \frac{1}{2}\log\left(\frac{2\log(|\mathcal{M}|/\delta)}{|D_{s,a|}|}\right)$, C_R is the maximum of the lagrange remainder $\sum_{n=2} R_n(x)\,(x'-x)^n$.

This theorem provides the theoretical basis for directly using the score, $\nabla_x \log p_{\mathcal{D}}(x)$, as a penalty. The penalty will have a high value when x' is within the distribution (i.e., $p(x') \geq C_p$), while x is outside it. Suppose the agent is constrained to remain within the dataset's support, meaning $p(x') \geq C_p$ is always satisfied. This penalty will theoretically prevent the agent from entering transitions where the dataset's density function rapidly decreases.

Notably, the score-based diffusion model naturally predicts a large gradient when a data point is outside the dataset's support. This makes it feasible to use $h_{\xi}(x)$ directly as a penalty in the model-based offline RL setting, which assigns a large penalty to a given transition (s,a,s') when either (s,a) is outside the dataset distribution or the transition experiences a rapid decline in data density. In Section 5, we conduct experiments to prove this hypothesis.

4.3 Employing Penalties in Training Process

We develop ScorePen-MORL based on MOPO [57]. Similar to MOPO, ScorePen-MORL trains an ensemble of dynamics models, $\left\{\widehat{T}^i_{\theta}\right\}_{i=1}^{N}$, where the aggregate model is defined as $\widehat{T}_{\theta} = \frac{1}{N}\sum_{i=1}^{N} \widehat{T}^i_{\theta}$. Each component $\widehat{T}^i_{\theta}$ is a neural network that outputs a Gaussian distribution over the next state and reward, trained via maximum likelihood.

ScorePen-MORL is trained in both synthetic and real data. For each update during training, we generate synthetic rollouts with h steps from states sampled from $\mathcal{D}$ and then add them to the synthetic dataset $\mathcal{D}_{model}$. We use the popular structure of soft actor-critic (SAC) [7] to train the agent:

$$\mathcal{L}_{\text{critic}} = \mathbb{E}_{(s,a,r,s') \sim \mathcal{D} \cup \mathcal{D}_{\text{model}}} \left[\left(Q_{\psi_k} - y \right)^2 \right], \tag{19}$$

where the target value y is:

$$y = \begin{cases} r + \gamma \left[\min_{k=1,2} Q_{\psi^-_k}(s',a') - \alpha \log \pi_{\phi}(a' \mid s') \right], \\ \qquad\qquad\qquad\qquad \text{for } (s,a,r,s') \in \mathcal{D} \\ r + \gamma \left[\min_{k=1,2} Q_{\psi^-_k}(s',a') - \alpha \log \pi_{\phi}(a' \mid s') \right] - \beta ||h_{\xi}(s,a)||_2^2. \\ \qquad\qquad\qquad\qquad \text{for } (s,a,r,s') \in \mathcal{D}_{\text{model}} \end{cases} \tag{20}$$

As shown in Equation 20, we use only the second norm of the score to penalize the targets of the samples from D_{model}, following Lemma 4.6. The policy $\pi_{\phi}(a|s)$ is optimized by solving:

$$\pi_{\phi} := \max_{\phi} \mathbb{E}_{\substack{s \sim \mathcal{D} \cup \mathcal{D}_{\text{model}} \\ a \sim \pi_{\phi}}} \left[\min_{k=1,2} Q_{\psi_k}(s,a) - \alpha \log \pi_{\phi}(a \mid s) \right]. \tag{21}$$

Algorithm 1 ScorePen-MORL

Require: Dataset $\mathcal{D}$, dynamics models $\left\{\widehat{T}_\theta^i\right\}_{i=1}^N$, initialized policy π_ϕ and critics $\{Q_{\psi_1}, Q_{\psi_2}\}$.

Train the ensemble of dynamics models $\widehat{T}_\theta(s', r|s, a) = \mathcal{N}(\mu_\theta(s, a), \Sigma_\theta(s, a))$ on the offline dataset $\mathcal{D}$.
Train the score-based diffusion model via minimizing Eq. 10.
for epoch = 1 to N **do**
  Generate synthetic h-step rollouts by $\hat{T}_\theta$ and add synthetic data to $\mathcal{D}_{model}$.
  Sample a mini-batch Batch = $\{s, a, r, s'\}$ from $\mathcal{D} \cup \mathcal{D}_{model}$.
  Compute targets for Batch according to Eq. 20.
  Update critics ψ_1, ψ_2 with gradient descent via minimizing Eq. 19.
  Update actor ϕ with gradient ascent via minimizing Eq. 21.
end for

The pseudocode for the ScorePen-MORL algorithm is presented in Algorithm 1.

5 EXPERIMENTS

In this section, we present a series of experiments aimed at addressing the following questions:

Q1: How does the "cliff regions" issue impact or limit current model-based offline reinforcement learning algorithms?

Q2: How does our approach tackle the issue of the "cliff regions"?

Q3: How does ScorePen-MORL measure up against previous state-of-the-art baseline algorithms?

Q4: Given that ScorePen-MORL and existing baseline algorithms concentrate on different aspects, could integrating ScorePen-MORL with them enhance overall performance?

5.1 Analysis with a Simple Environment

To address **Q1** and **Q2**, we design a simple environment and intentionally generate a dataset that creates the "cliff regions" issue. States are represented as pairs (x, y), with bounded 2D actions (δ_x, δ_y) that update the agent's position according to:

$$(x, y) \xrightarrow{(\delta_x, \delta_y)} (x + \delta_x, y + \delta_y),$$

subject to $|\delta_x|, |\delta_y| \leq 0.05$. The reward function, defined as $R(s, a) = \exp\left(-\frac{1}{2}\|s - g\|_2^2\right)$, decreases exponentially as the agent moves away from the goal $g = (0.2, 0.4)$. The initial state distribution is centered at the origin, modeled by $\mu_0 = U([-0.1, 0.1]^2)$. The maximum steps of one episode is 30. The dataset is confined to a square region with a side length of 1, centered within the environment, with data points uniformly distributed inside and none outside. The boundary of this dataset naturally forms the cliff region. We apply ScorePen-MORL, Count-MORL (density-based), and MOBILE (uncertainty-based) to this dataset to illustrate the distinctions between their respective penalty mechanisms.

Figure 2 illustrates the environment, along with the penalties generated by all the algorithms and their corresponding performance. We conduct a parameter search for each method over the penalty coefficient, $\beta \in [0.25, 2]$ in increments of 0.25. When the penalty scale is low, all algorithms perform poorly as the agents consistently approach states outside the dataset boundary while neglecting g. As β increases, the performance of each algorithm gradually improves and then decreases after reaching the optimal β. However, during this search, only ScorePen-MORL successfully finds the optimal path. The baseline algorithms, Count-MORL and MOBILE [45], after learning to stay away from the dataset boundaries, also become more conservative, maintaining a certain distance from both g and the boundaries rather than truly approaching g. Also, ScorePen-MORL's penalties closely follow the dataset's boundaries, whereas other methods struggle to capture the rectangular structure. Those results indicate that existing model-based offline RL algorithms, due to their inability to analyze the cliff region in the dataset accurately, struggle to find a trade-off point that allows agents to effectively utilize information near the cliff while remaining within the dataset's distribution support.

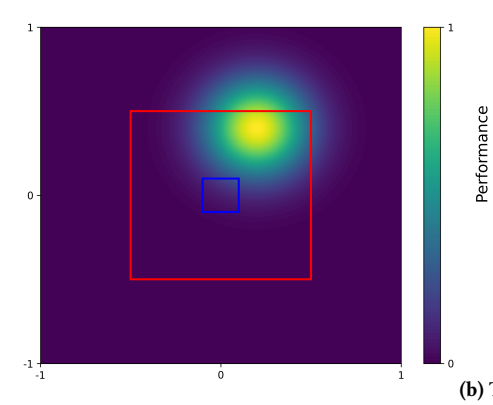
We continue to analyze why existing model-based offline RL algorithms struggle with penalties near the cliff region. For density-based approaches, penalties are generated using estimated dataset density functions, making their effectiveness highly dependent on the accuracy of these estimates. Inaccurate reconstructed density model can result in cumulative errors in gradient estimation, hindering the propagation of critical information from cliff regions to the RL agent. Count-based methods, which discretize the space using tools such as hash codes, further reduce the precision of gradient information. Uncertainty-based approaches rely on ensembles to estimate model uncertainty. Still, the limited expressiveness of these ensembles often leads to inaccurate uncertainty estimates in regions with sparse data, such as cliff areas. This misjudgment can result in penalties that fail to guide the agent's behavior near these challenging regions appropriately.

5.2 Evaluation on the D4RL Benchmark

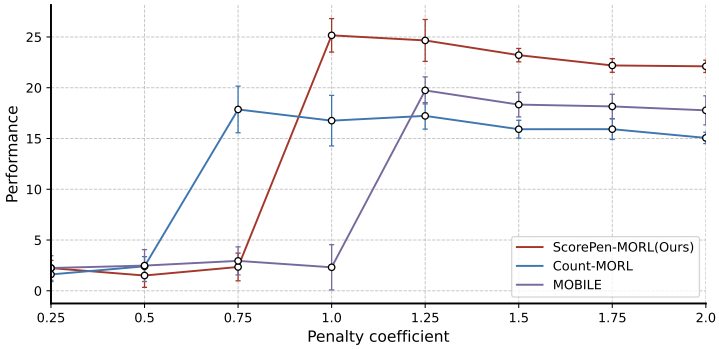
To answer **Q3**, we evaluate ScorePen-MORL on the standard offline RL benchmark D4RL. We compare ScorePen-MORL with various offline RL algorithms, categorized as follows: 1) **Model-free methods**: BC, CQL, TD3+BC, and EDAC. 2) **Model-based methods**: MOPO, COMBO, TT, RAMBO, Count-MORL, and MOBILE.

We assess these approaches across twelve datasets for the Gym domain, covering three environments: hopper, walker2d, and halfcheetah, which represent different robots. Each environment features four types of datasets: random, medium, medium-replay, and medium-expert. The random dataset contains transitions collected by a random policy. The medium dataset consists of transitions collected by an early-stopped SAC policy. Medium-replay includes the replay buffer generated during the training of the medium policy. The medium-expert dataset contains a mixture of suboptimal and expert data. All datasets utilized in our experiments are of the "v2" version.

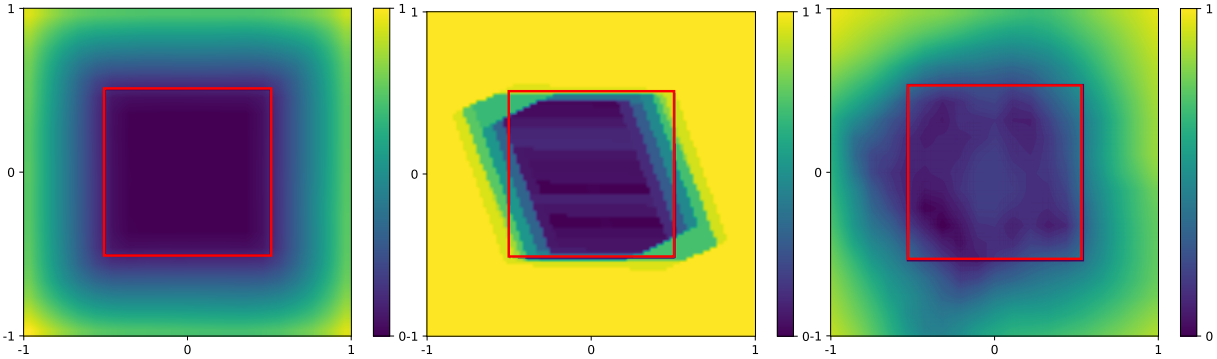
Table 1 presents the results of the normalized score and the average performance across all datasets on the d4rl benchmark. ScorePen-MORL outperforms all baseline algorithms in 8 out of 12 datasets and achieves the highest average score among all methods. These empirical results indicate that ScorePen-MORL, which exploits the gradient field of the dataset to obtain conservatism



(a) Environment Setting



(b) The performance of algorithms during the hyperparameter optimization of the penalty coefficient β averaged among 5 random seeds.



(c) Penalties in ScorePen-MORL (d) Penalties in Count-MORL (e) Penalties in MOBILE

Figure 2: The environmental setting is presented in (a). The performance results of algorithms with different penalty coefficients β are shown in (b). In (c)-(e), we compare the differences of generated penalties between ScorePen-MORL and the baseline algorithms. In (d), the penalties of Count-MORL are generated using tools like hash codes, resulting in a form that appears as a linear transformation of the ground truth data density function. However, its boundaries do not completely align with the dataset distribution's edges. In (e), although uncertainty-based methods perform well in penalizing near the boundaries of the original dataset distribution, there is a significant discrepancy in the correlation with data density near the dataset's edges. The penalties of MOBILE are visualized after 1 million training steps.

knowledge without requiring ensembles and their higher computational cost, can deliver a better policy than those based on dataset density and uncertainty measures.

As ScorePen-MORL mainly focuses on the cliff regions and the gradient field of the dataset, which is different from recent strong baseline algorithms, we expect combining ScorePen-MORL with them will achieve better performance, which will answer to **Q4**. In Table 2, we conclude the related experimental results. Combining our method with Count-MORL or MOBILE results in performance improvements on 7 out of 12 datasets, with 5 datasets showing better results compared to using our method alone. The results suggest that integrating ScorePen-MORL with other algorithms effectively addresses the cliff region issue and leads to improved performance, aligning with our initial hypothesis.

5.3 NEORL

NeoRL is a benchmark designed to simulate real-world scenarios by collecting datasets using a more conservative policy, better reflecting actual data-collection practices. The resulting narrow and limited data presents challenges for offline RL algorithms. Our study focuses on nine datasets, encompassing three environments: HalfCheetah-v3, Hopper-v3, and Walker2d-v3. These datasets are categorized into three types (L, M, H), representing low, medium, and high-quality data. NeoRL provides varying training trajectories (100, 1000, and 10000) for each task, and we selected a uniform sample of 1000 trajectories for our experiments.

In table 3, our evaluation assesses the ScorePen-MORL's performance against recent strong baselines. We do not include COMBO, TT, RAMBO, BC, TD3+BC, EDAC, and Count-MORL for low peformance or unknown parameter settings. As a result, a fair comparison would not be possible without appropriate tuning and publicly available performance metrics for these methods.

Table 1: Experimental results on the D4RL benchmark. ScorePen-MORL outperforms model-free and model-based baseline algrotithms. All the results are averaged between 5 random seeds. The results of MOPO are derived from the reproduced findings reported in Table 1 of the MOBILE paper.

Task Name	BC	CQL	TD3+BC	EDAC	MOPO	COMBO	TT	RAMBO	Count-MORL	MOBILE	ScorePen-MORL
ha-r	2.2	31.3	11.0	28.4	38.5	38.8	6.1	39.5	**41.0**	39.3	40.3±0.2
ho-r	3.7	5.3	8.5	25.3	31.7	17.9	6.9	25.4	30.7	31.9	**32.3**±0.6
wa-r	1.3	5.4	1.6	16.6	7.4	7.0	5.9	0.0	**21.9**	17.9	19.9±1.1
ha-m	43.2	46.9	48.3	65.9	73.0	54.2	46.9	77.9	76.5	74.6	**79.2**±0.4
ho-m	54.1	61.9	59.3	101.6	62.8	97.2	67.4	87.0	103.6	**106.6**	106.0±0.7
wa-m	70.9	79.5	83.7	92.5	84.1	81.9	81.3	84.9	87.6	87.7	**91.7**±0.8
ha-mr	37.6	45.3	44.6	61.3	72.1	55.1	44.1	68.7	71.5	71.7	**77.6**±1.8
ho-mr	16.6	86.3	60.9	101.0	103.5	89.5	99.4	99.5	101.7	103.9	**105.8**±0.6
wa-mr	20.3	76.8	81.8	87.1	85.6	56.0	82.6	89.2	87.7	89.9	**93.0**±1.3
ha-me	44.0	95.0	90.7	106.3	90.8	90.0	95.0	95.4	100.0	108.2	**112.3**±4.2
ho-me	53.9	96.9	98.0	110.7	81.6	111.1	110.0	88.2	111.4	112.6	**113.2**±0.4
wa-me	90.1	109.1	110.1	114.7	112.9	103.3	101.9	56.7	112.3	**115.2**	112.5±1.1
Average	36.5	61.6	58.2	76.0	70.3	66.8	62.3	67.7	78.8	80.0	**82.0**±1.1

Table 2: Experimental results of ScorePen-MORL combined with recent strong baseline algorithms Count-MORL and MOBILE on the D4RL benchmark. ScorePen-MORL improves baseline algorithms in 7 of 12 datasets. All the results are averaged between 5 random seeds.

Task Name	Count-MORL (w/ ours)	MOBILE (w/ ours)
ha-r	41.0±0.9 → **47.2**±1.2	39.3±3.0 → 44.6±2.8
ho-r	30.7±1.3 → 31.7±0.1	**31.9**±0.6 → 31.5±0.2
wa-r	**21.9**±0.2 → 20.4±0.3	17.9±6.6 → 19.9±0.7
ha-m	76.5±1.7 → **78.8**±0.2	74.6±1.2 → **78.7**±0.1
ho-m	103.6±3.7 → 105.4±0.6	**106.6**±0.6 → 106.1±0.7
wa-m	87.6±3.7 → **90.0**±0.2	87.7±1.1 → 89.7±1.8
ha-mr	71.5±1.8 → 71.0±1.5	**71.7**±1.2 → 71.5±1.1
ho-mr	101.7±0.8 → 103.1±0.7	103.9±1.0 → **104.3**±0.6
wa-mr	87.7±3.0 → 89.0±1.2	89.9±1.5 → **91.1**±0.6
ha-me	100.0±4.9 → 100.7±3.6	108.2±2.5 → **110.0**±1.9
ho-me	111.4±0.5 → 113.1±0.4	112.6±0.2 → **113.3**±0.3
wa-me	112.3±1.8 → 113.8±0.4	**115.2**±0.7 → 114.2±3.1
Average	78.8±2.0 → 80.4±0.9	80.0±1.7 → **81.2**±1.2

Table 3: Normalized average returns on NeoRL tasks, averaged over 5 random seeds, indicate that ScorePen-MORL outperforms all baseline algorithms on 5 out of 9 datasets and achieves the highest overall average performance across all datasets.

Task Name	CQL	MOPO	MOBILE	Ours
HalfCheetah-L	38.2	40.1	54.7	49.6±1.2
Hopper-L	16.0	6.2	17.4	**21.1**±2.3
Walker2d-L	44.7	11.6	37.6	**51.4**±1.4
HalfCheetah-M	54.6	62.3	77.8	77.4±1.0
Hopper-M	64.5	1.0	51.1	**90.9**±1.3
Walker2d-M	57.3	39.9	62.2	**65.8**±1.6
HalfCheetah-H	77.4	65.9	83.0	81.4±1.0
Hopper-H	76.6	11.5	87.8	86.3±1.3
Walker2d-H	75.3	18.0	74.9	**78.0**±1.8
Average	56.1	28.5	60.7	**67.0**±1.4

Our results demonstrate that ScorePen-MORL consistently achieves superior or competitive performance across most tasks by outperforming baseline algorithms in 5 of 9 datasets.

6 CONCLUSION

We identify the cliff region issue, a previously overlooked model-based offline reinforcement learning bottleneck. We analyze how this issue limits existing methods through logical reasoning and illustrative experiments. To address the cliff region issue, we propose Score as a Penalty for Model-based Offline Reinforcement Learning (ScorePen-MORL). This novel plug-and-play algorithm employs a score-based diffusion model to penalize out-of-distribution (OOD) synthetic state-action pairs using insights from the dataset's gradient field, an area largely unexplored in previous research. We provide a detailed motivation and theoretical analysis for our approach. Empirical results demonstrate that ScorePen-MORL achieves comparable or superior performance to recent strong baselines on its own and enhances the performance of these baselines when integrated. These findings underscore the importance of the cliff region issue and the impact of ScorePen-MORL on advancing model-based offline RL.

ACKNOWLEDGMENTS

This work was supported by the STI 2030-Major Projects under Grant 2021ZD0201404. The authors also thank the anonymous reviewers for valuable comments.

REFERENCES

[1] Rishabh Agarwal, Dale Schuurmans, and Mohammad Norouzi. 2020. An optimistic perspective on offline reinforcement learning. In *International conference on machine learning*. PMLR, 104–114.

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

[3] Chenjia Bai, Lingxiao Wang, Zhuoran Yang, Zhihong Deng, Animesh Garg, Peng Liu, and Zhaoran Wang. 2022. Pessimistic bootstrapping for uncertainty-driven offline reinforcement learning. *arXiv preprint arXiv:2202.11566* (2022).

[4] Scott Fujimoto and Shixiang Shane Gu. 2021. A minimalist approach to offline reinforcement learning. *Advances in neural information processing systems* 34 (2021), 20132–20145.

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

[6] Carles Gelada and Marc G Bellemare. 2019. Off-policy deep reinforcement learning by bootstrapping the covariate shift. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 3647–3655.

[7] Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. 2018. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International conference on machine learning*. PMLR, 1861–1870.

[8] Chunming He, Chengyu Fang, Yulun Zhang, Tian Ye, Kai Li, Longxiang Tang, Zhenhua Guo, Xiu Li, and Sina Farsiu. 2023. Reti-diff: Illumination degradation image restoration with retinex-based latent diffusion model. *arXiv preprint arXiv:2311.11638* (2023).

[9] Chunming He, Kai Li, Yachao Zhang, Longxiang Tang, Yulun Zhang, Zhenhua Guo, and Xiu Li. 2023. Camouflaged object detection with feature decomposition and edge reconstruction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 22046–22055.

[10] Chunming He, Yuqi Shen, Chengyu Fang, Fengyang Xiao, Longxiang Tang, Yulun Zhang, Wangmeng Zuo, Zhenhua Guo, and Xiu Li. 2024. Diffusion Models in Low-Level Vision: A Survey. *arXiv preprint arXiv:2406.11138* (2024).

[11] Longxiang He, Li Shen, Linrui Zhang, Junbo Tan, and Xueqian Wang. 2024. DiffCPS: Diffusion Model based Constrained Policy Search for Offline Reinforcement Learning. arXiv:2310.05333 [cs.LG] https://arxiv.org/abs/2310.05333

[12] Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems* 33 (2020), 6840–6851.

[13] Michael Janner, Justin Fu, Marvin Zhang, and Sergey Levine. 2019. When to trust your model: Model-based policy optimization. *Advances in neural information processing systems* 32 (2019).

[14] Jihwan Jeong, Xiaoyu Wang, Michael Gimelfarb, Hyunwoo Kim, Baher Abdulhai, and Scott Sanner. 2023. Conservative Bayesian Model-Based Value Expansion for Offline Policy Optimization. arXiv:2210.03802 [cs.LG] https://arxiv.org/abs/2210.03802

[15] Jun Jin, Daniel Graves, Cameron Haigh, Jun Luo, and Martin Jagersand. 2020. Offline learning of counterfactual perception as prediction for real-world robotic reinforcement learning. *arXiv preprint arXiv:2011.05857* (2020).

[16] Rahul Kidambi, Aravind Rajeswaran, Praneeth Netrapalli, and Thorsten Joachims. 2020. Morel: Model-based offline reinforcement learning. *Advances in neural information processing systems* 33 (2020), 21810–21823.

[17] Byeongchan Kim and Min hwan Oh. 2023. Model-based Offline Reinforcement Learning with Count-based Conservatism. arXiv:2307.11352 [cs.LG] https://arxiv.org/abs/2307.11352

[18] Byeongchan Kim and Min-hwan Oh. 2023. Model-based offline reinforcement learning with count-based conservatism. In *International Conference on Machine Learning*. PMLR, 16728–16746.

[19] B Ravi Kiran, Ibrahim Sobh, Victor Talpaert, Patrick Mannion, Ahmad A Al Sallab, Senthil Yogamani, and Patrick Pérez. 2021. Deep reinforcement learning for autonomous driving: A survey. *IEEE Transactions on Intelligent Transportation Systems* 23, 6 (2021), 4909–4926.

[20] Zhifeng Kong, Wei Ping, Jiaji Huang, Kexin Zhao, and Bryan Catanzaro. 2020. Diffwave: A versatile diffusion model for audio synthesis. *arXiv preprint arXiv:2009.09761* (2020).

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

[22] Aviral Kumar, Justin Fu, Matthew Soh, George Tucker, and Sergey Levine. 2019. Stabilizing off-policy q-learning via bootstrapping error reduction. *Advances in neural information processing systems* 32 (2019).

[23] Aviral Kumar, Aurick Zhou, George Tucker, and Sergey Levine. 2020. Conservative q-learning for offline reinforcement learning. *Advances in Neural Information Processing Systems* 33 (2020), 1179–1191.

[24] Yao Liu, Adith Swaminathan, Alekh Agarwal, and Emma Brunskill. 2019. Off-policy policy gradient with state distribution correction. *arXiv preprint arXiv:1904.08473* (2019).

[25] Yao Liu, Adith Swaminathan, Alekh Agarwal, and Emma Brunskill. 2020. Provably good batch off-policy reinforcement learning without great exploration. *Advances in neural information processing systems* 33 (2020), 1264–1274.

[26] Zeyuan Liu, Kai Yang, and Xiu Li. 2024. CDSA: Conservative Denoising Score-based Algorithm for Offline Reinforcement Learning. arXiv:2406.07541 [cs.LG] https://arxiv.org/abs/2406.07541

[27] Cong Lu, Philip J Ball, Jack Parker-Holder, Michael A Osborne, and Stephen J Roberts. 2021. Revisiting design choices in offline model-based reinforcement learning. *arXiv preprint arXiv:2110.04135* (2021).

[28] Jiafei Lyu, Xiu Li, and Zongqing Lu. 2022. Double Check Your State Before Trusting It: Confidence-Aware Bidirectional Offline Model-Based Imagination. In *Advances in Neural Information Processing Systems*, Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (Eds.).

[29] Jiafei Lyu, Xiaoteng Ma, Xiu Li, and Zongqing Lu. 2022. Mildly Conservative Q-learning for Offline Reinforcement Learning. In *Thirty-sixth Conference on Neural Information Processing Systems*.

[30] Rowan McAllister, Gregory Kahn, Jeff Clune, and Sergey Levine. 2019. Robustness to out-of-distribution inputs via task-aware generative uncertainty. In *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2083–2089.

[31] Chenhao Niu, Yang Song, Jiaming Song, Shengjia Zhao, Aditya Grover, and Stefano Ermon. 2020. Permutation invariant graph generation via score-based generative modeling. In *International Conference on Artificial Intelligence and Statistics*. PMLR, 4474–4484.

[32] Georg Ostrovski, Marc G Bellemare, Aäron Oord, and Rémi Munos. 2017. Count-based exploration with neural density models. In *International conference on machine learning*. PMLR, 2721–2730.

[33] Zhongjian Qiao, Jiafei Lyu, Kechen Jiao, Qi Liu, and Xiu Li. 2024. SUMO: Search-Based Uncertainty Estimation for Model-Based Offline Reinforcement Learning. *CoRR* abs/2408.12970 (2024).

[34] Rafael Rafailov, Tianhe Yu, Aravind Rajeswaran, and Chelsea Finn. 2021. Offline reinforcement learning from images with latent space models. In *Learning for dynamics and control*. PMLR, 1154–1168.

[35] Charles Richter and Nicholas Roy. 2017. Safe visual navigation via deep learning and novelty detection. (2017).

[36] Marc Rigter, Bruno Lacerda, and Nick Hawes. 2022. Rambo-rl: Robust adversarial model-based offline reinforcement learning. *Advances in neural information processing systems* 35 (2022), 16082–16097.

[37] Marc Rigter, Bruno Lacerda, and Nick Hawes. 2022. RAMBO-RL: Robust Adversarial Model-Based Offline Reinforcement Learning. arXiv:2204.12581 [cs.LG] https://arxiv.org/abs/2204.12581

[38] Laixi Shi, Gen Li, Yuting Wei, Yuxin Chen, and Yuejie Chi. 2022. Pessimistic q-learning for offline reinforcement learning: Towards optimal sample complexity. In *International conference on machine learning*. PMLR, 19967–20025.

[39] Noah Y Siegel, Jost Tobias Springenberg, Felix Berkenkamp, Abbas Abdolmaleki, Michael Neunert, Thomas Lampe, Roland Hafner, Nicolas Heess, and Martin Riedmiller. 2020. Keep doing what worked: Behavioral modelling priors for offline reinforcement learning. *arXiv preprint arXiv:2002.08396* (2020).

[40] Anya Sims, Cong Lu, and Yee Whye Teh. [n.d.]. RAVL: Reach-Aware Value Learning for the Edge-of-Reach Problem in Offline Model-Based Reinforcement Learning. ([n. d.]).

[41] Avi Singh, Albert Yu, Jonathan Yang, Jesse Zhang, Aviral Kumar, and Sergey Levine. 2020. Cog: Connecting new skills to past experience with offline reinforcement learning. *arXiv preprint arXiv:2010.14500* (2020).

[42] Samarth Sinha, Ajay Mandlekar, and Animesh Garg. 2022. S4rl: Surprisingly simple self-supervision for offline reinforcement learning in robotics. In *Conference on Robot Learning*. PMLR, 907–917.

[43] Yang Song and Stefano Ermon. 2019. Generative modeling by estimating gradients of the data distribution. *Advances in Neural Information Processing Systems* 32 (2019).

[44] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. 2020. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456* (2020).

[45] Yihao Sun, Jiaji Zhang, Chengxing Jia, Haoxin Lin, Junyin Ye, and Yang Yu. 2023. Model-Bellman inconsistency for model-based offline reinforcement learning. In *International Conference on Machine Learning*. PMLR, 33177–33194.

[46] Yihao Sun, Jiaji Zhang, Chengxing Jia, Haoxin Lin, Junyin Ye, and Yang Yu. 2023. Model-Bellman Inconsistency for Model-based Offline Reinforcement Learning. In *Proceedings of the 40th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 202)*, Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (Eds.). PMLR, 33177–33194. https://proceedings.mlr.press/v202/sun23q.html

[47] Richard S Sutton, A Rupam Mahmood, and Martha White. 2016. An emphatic approach to the problem of off-policy temporal-difference learning. *The Journal of Machine Learning Research* 17, 1 (2016), 2603–2631.

[48] Arash Vahdat, Karsten Kreis, and Jan Kautz. 2021. Score-based generative modeling in latent space. *Advances in Neural Information Processing Systems* 34 (2021), 11287–11302.

[49] Zhendong Wang, Jonathan J Hunt, and Mingyuan Zhou. 2022. Diffusion policies as an expressive policy class for offline reinforcement learning. *arXiv preprint arXiv:2208.06193* (2022).
[50] Max Welling and Yee W Teh. 2011. Bayesian learning via stochastic gradient Langevin dynamics. In *Proceedings of the 28th international conference on machine learning (ICML-11)*. 681–688.
[51] Mingdong Wu, Fangwei Zhong, Yulong Xia, and Hao Dong. 2022. Targf: Learning target gradient field for object rearrangement. *arXiv preprint arXiv:2209.00853* (2022).
[52] Yifan Wu, George Tucker, and Ofir Nachum. 2019. Behavior regularized offline reinforcement learning. *arXiv preprint arXiv:1911.11361* (2019).
[53] Yue Wu, Shuangfei Zhai, Nitish Srivastava, Joshua Susskind, Jian Zhang, Ruslan Salakhutdinov, and Hanlin Goh. 2021. Uncertainty weighted actor-critic for offline reinforcement learning. *arXiv preprint arXiv:2105.08140* (2021).
[54] Kai Yang, Jian Tao, Jiafei Lyu, and Xiu Li. 2024. Exploration and Anti-Exploration with Distributional Random Network Distillation. *arXiv preprint arXiv:2401.09750* (2024).
[55] Tianhe Yu, Aviral Kumar, Rafael Rafailov, Aravind Rajeswaran, Sergey Levine, and Chelsea Finn. 2021. Combo: Conservative offline model-based policy optimization. *Advances in neural information processing systems* 34 (2021), 28954–28967.
[56] Tianhe Yu, Aviral Kumar, Rafael Rafailov, Aravind Rajeswaran, Sergey Levine, and Chelsea Finn. 2022. COMBO: Conservative Offline Model-Based Policy Optimization. arXiv:2102.08363 [cs.LG] https://arxiv.org/abs/2102.08363
[57] Tianhe Yu, Garrett Thomas, Lantao Yu, Stefano Ermon, James Y Zou, Sergey Levine, Chelsea Finn, and Tengyu Ma. 2020. Mopo: Model-based offline policy optimization. *Advances in Neural Information Processing Systems* 33 (2020), 14129–14142.
[58] Junjie Zhang, Jiafei Lyu, Xiaoteng Ma, Jiangpeng Yan, Jun Yang, Le Wan, and Xiu Li. 2023. Uncertainty-driven Trajectory Truncation for Model-based Offline Reinforcement Learning. *CoRR* abs/2304.04660 (2023).