Reproducibility Report

Jiahang Wang, Li Jiang, Liting Chen
McGill University

Abstract

This reproducibility report assesses "Offline Reinforcement Learning with Implicit Q-Learning" [5], focusing on validating IQL's efficacy in offline RL. We test key claims: IQL's ability to learn without querying unseen actions, the need for a large expectile (τ) in complex tasks, and its computational efficiency versus baseline methods. Additionally, we conduct an ablation study on parameters like expectile τ, temperature α, dropout rate, batch size, network depth, and learning rate, to understand their impact on IQL's performance. Based on our experimental findings, especially when τ approaches 1, we demonstrate the conflict between the theoretical conclusion and experiments. The performance noticeably declines when the expectile parameter τ approaches the extreme value near 1, contradicting the theoretical assertion that setting τ close to 1 should yield an optimal policy. This report aims to deepen the understanding of IQL in offline RL, both in practice and theory.

1 Introduction

In this reproducibility report, we delve into the findings of "Offline Reinforcement Learning with Implicit Q-Learning (IQL)" [5]. Our primary objective is to assess and validate the core assertions of the original work, specifically focusing on the efficacy of IQL in the realm of offline reinforcement learning (RL). The seminal paper in question addresses a pivotal challenge in offline RL: devising an enhanced policy over the behavior policy responsible for the dataset collection, while concurrently minimizing deviation from this behavior policy to mitigate errors stemming from distributional shifts.

Our exploration is grounded on a series of systematic experiments and detailed analyses that aim to replicate the key claims posited in the original paper. These claims include IQL's capability to avert querying values of unseen actions during training, the necessity of a larger expectile (τ) in tasks that require "stitching," and the computational efficiency of IQL compared to baseline methods in JAX.

Through our reproducibility efforts, we endeavor to provide a comprehensive and insightful examination of IQL's methodologies and claims. Our approach encompasses a thorough assessment of the original implementation, datasets, and hardware setups, as well as a critical evaluation of model architecture and performance metrics.

Furthermore, we extend our investigation beyond the primary claims of the original paper to include a comprehensive ablation study. This study scrutinizes various critical components and hyperparameters within the IQL framework, enabling us to dissect and understand their individual and collective impacts on model performance. Our ablation study encompasses the following key areas: expectile τ, temperature α, dropout rate p, batch size B, network depth d, and learning rate lr.

Lastly, based on our experimental findings, we question and analyze the theorem underpinning IQL in our discussion, aiming to reconcile theoretical assertions with empirical findings. This aspect of our study not only probes the practical applications of the theorem but also seeks to uncover any discrepancies or limitations in the theoretical framework as applied in real-world scenarios. This comprehensive approach ensures a thorough understanding of both the practical and theoretical dimensions of IQL, contributing significantly to the field of offline reinforcement learning.

2 Scope of reproducibility

In this work, we aim to reproduce and validate key findings on "Offline Reinforcement Learning with Implicit Q-Learning (IQL)" [5]. The original paper focuses on addressing a fundamental challenge in offline reinforcement learning (RL): learning an improved policy over the behavior policy that collected the dataset, while minimizing deviation from this behavior policy to avoid errors due to distributional shift. We will evaluate the claims made in the paper through systematic experiments and analysis. The key claims from the original paper that we aim to reproduce and assess are:

COMP 551 2023, Applied Machine Learning, McGill University.

- **Claim 1 (performance superior):** IQL successfully avoids querying values of unseen actions during training while still being able to perform multi-step dynamic programming updates. This claim suggests that IQL can learn effective policies without the need for explicit exploration of unseen actions in the training dataset.
- **Claim 2 (requires large expectile):** IQL demonstrates that it is crucial to compute a larger expectile (τ) on tasks that require "stitching". With larger values of τ, it approximates Q-learning better, leading to better performance on the Ant Maze tasks.
- **Claim 3 (computation efficient):** IQL is computationally faster than baseline methods in JAX, which generally offer faster performance than their original implementations.

3 Methodology

In this section, we introduce we implementation details, including the codebase source, dataset, hardware setups, model architecture, and metric evaluation approaches.

Codebase. In our attempt to reproduce the findings of the IQL, we primarily relied on the codebase provided by the authors[1]. This decision was made to ensure a high fidelity to the original implementation, facilitating direct comparison of results and minimizing discrepancies that might arise from differences in coding or algorithm interpretation.

Dataset and tasks. D4RL [3] is a suite of benchmarks specifically designed for offline reinforcement learning (RL), where policies are learned from a static dataset rather than from online interaction with the environment, as shown in Fig. 1. It aims to take advantage of large, previously-collected datasets to address key properties relevant to real-world applications of offline RL, such as hand-designed controllers, human demonstrations, multitask scenarios, and mixtures of policies.

The `AntMaze` task involves a navigation domain with a more complex 8-DoF "Ant" quadruped robot replacing the 2D ball from Maze2D. The domain tests the ability to "stitch" together trajectories in a morphologically complex setting that mimics real-world robotic navigation, using a sparse 0-1 reward activated upon reaching the goal.

The `Gym-MuJoCo` tasks, which include Hopper, HalfCheetah, and Walker2d, these are standard benchmarks used in offline deep RL with standardized datasets. These tasks are meant to test algorithms on the impact of heterogeneous policy mixtures and expect that methods relying on regularizing to the behavior policy might fail when the data contains poorly performing trajectories.

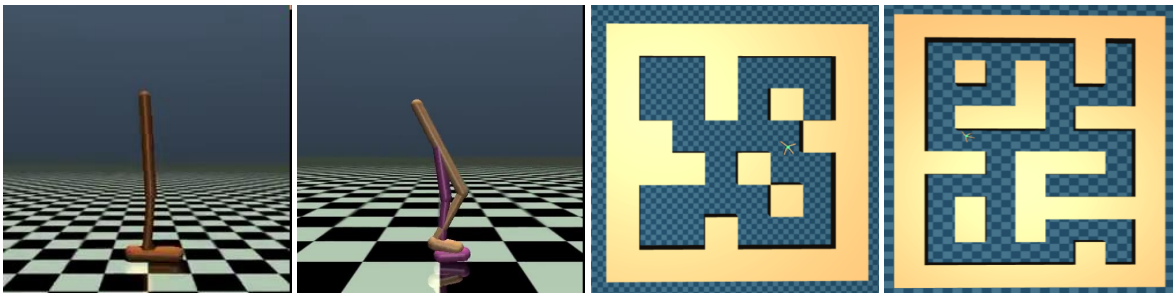

Figure 1: Demonstration Tasks in D4RL.

Hardware & software setups. In this section, we provide the experimental details of our paper. We use the following hardware and software for our training:

- CPU: i11 7700HQ
- GPUs: NVIDIA GeForce RTX 3090Ti
- Python 3.11
- Jax 0.3.13 [1]
- Gym 0.23.1 [2]
- Mujoco 2.1.4 [6]
- mujoco-py 2.1.2.14

[1] https://github.com/ikostrikov/implicit_q_learning

Model Architecture. Tab. 1 shows the training hyperparameters of IQL in our reproducibility report.

Table 1: The default hyperparameters of IQL in our report.

	Hyperparameter	Value
Architecture	Value network hidden dim	64
	Value network hidden layers	2
	Value network activation function	ReLU
	Guide-policy hidden dim	64
	Guide-policy hidden layers	2
	Guide-policy activation function	ReLU
	Execute-policy hidden dim	64
	Execute-policy hidden layers	2
	Execute-policy activation function	ReLU
Training Hyperparameters	Optimizer	Adam [4]
	Value network learning rate	1e-4
	Target V moving average	0.05
	Guide-policy learning rate	1e-4
	Execute-policy learning rate	1e-4
	Mini-batch size	256
	Discount factor	0.99
	Normalize	False
	τ	0.9
	α	10.0

Metric Evaluation. For MuJoCo locomotion and Kitchen tasks, we average mean returns over 10 evaluations every 5000 training steps, over 5 random seeds. For AntMaze tasks, we average over 100 evaluations every 0.1M training steps, over 5 random seeds. Followed by IQL, we standardize the rewards by dividing the difference in returns of the best and worst trajectories in MuJoCo and kitchen tasks, we subtract 1 to rewards in AntMaze tasks. In our study, we primarily evaluate the efficacy of IQL on the AntMaze tasks from the D4RL benchmark [3]. This evaluation demonstrates the considerable potential of IQL in this context.

4 Results

This section presents a high-level overview of the results obtained from our experiments aimed at reproducing the findings of IQL. Our experimental results largely support the main claims of the original paper. Furthermore, we demonstrate substantial ablation study on different components of IQL, e.g., expectile τ, temperature α, batch size, network depth, and dropout.

4.1 Results reproducing original paper

Claim 1 (performance superior). Our experiments align with the original paper's assertion that IQL effectively avoids querying values of unseen actions during training while still executing multi-step dynamic programming updates, as demonstrated in Tab. 2.

Claim 2 (requires large expectile). The experimental results corroborate the necessity of a larger expectile (τ) as posited in the original paper, particularly in tasks involving "stitching", i.e., AntMaze tasks. Our findings demonstrate that increasing the value of τ does indeed lead to an approximation of Q-learning that is more effective, as shown by the enhanced performance in Section 4.2 Tab. 3. This supports the original claim about the critical role of a larger expectile in IQL's functioning.

However, our findings demonstrate that an extreme setting of $\tau = 0.99$ results in suboptimal performance compared to $\tau = 0.9$. This observation contradicts the theoretical framework established by IQL, where approaching the optimal value function V^* is theoretically achieved as $\tau \to 1$.

Table 2: Averaged normalized scores on MuJoCo locomotion and AntMaze tasks. Our reproducibility experiments demonstrate the consistency of the superior performance of IQL on the challenging Ant Maze tasks, which require dynamic programming and are competitive with the best prior methods on the locomotion tasks.

Dataset	BC	10%BC	DT	AWAC	Onestep RL	TD3+BC	CQL	IQL
halfcheetah-medium-v2	42.6	42.5	42.6	43.5	**48.4**	**48.3**	44.0	**47.4**±0.5
hopper-medium-v2	52.9	56.9	**67.6**	57.0	59.6	59.3	58.5	**66.2**±5.7
walker2d-medium-v2	75.3	75.0	74.0	72.4	**81.8**	83.7	72.5	76.5±8.7
halfcheetah-medium-replay-v2	36.6	40.6	36.6	40.5	38.1	**44.6**	**45.5**	**44.2**±1.2
hopper-medium-replay-v2	18.1	75.9	82.7	37.2	**97.5**	60.9	**95.0**	**94.7**±1.6
walker2d-medium-replay-v2	26.0	62.5	66.6	27.0	49.5	**81.8**	77.2	73.8±7.4
halfcheetah-medium-expert-v2	55.2	**92.9**	86.8	42.8	**93.4**	**90.7**	**91.6**	86.7±5.3
hopper-medium-expert-v2	52.5	**110.9**	**107.6**	55.8	103.3	98.0	**105.4**	91.5±14.3
walker2d-medium-expert-v2	**107.5**	**109.0**	**108.1**	74.5	**113.0**	**110.1**	**108.8**	**109.6**±1.0
antmaze-umaze-v0	54.6	62.8	59.2	56.7	64.3	78.6	74.0	**87.5**±2.6
antmaze-umaze-diverse-v0	45.6	50.2	53.0	49.3	60.7	71.4	**84.0**	66.2±13.8
antmaze-medium-play-v0	0.0	5.4	0.0	0.0	0.3	10.6	61.2	**71.2**±7.3
antmaze-medium-diverse-v0	0.0	9.8	0.0	0.7	0.0	3.0	53.7	**70.0**±10.9
antmaze-large-play-v0	0.0	0.0	0.0	0.0	0.0	0.2	15.8	**39.6**±5.8
antmaze-large-diverse-v0	0.0	6.0	0.0	1.0	0.0	0.0	14.9	**47.5**±9.5
runtime	10m	10m	960m	20m	$\approx$ 20m*	20m	80m	20m

Table 3: Averaged normalized scores on MuJoCo locomotion and AntMaze tasks with different ablation study hyperparameters. Our reproducibility experiments demonstrate the consistency of the superior performance of IQL on the challenging Ant Maze tasks, which require dynamic programming and are competitive with the best prior methods on the locomotion tasks.

	Params	Large-d	Large-p	Medium-d	Medium-p	Unmaze-d	Unmaze
Expectile (τ)	0.5	0.0±0.0	0.0±0.0	0.0±0.0	0.0±0.0	62.5±28.75	32.5±6.25
	0.6	0.0±0.0	0.0±0.0	2.5±6.25	0.0±0.0	35.0±25.0	77.5±12.5
	0.7	0.0±0.0	0.0±0.0	2.5±6.25	7.5±12.5	57.5±38.75	**90.0**±10.0
	0.8	2.5±6.25	2.5±6.25	27.5±6.25	55.0±20.0	**67.5**±12.5	**90.0**±10.0
	0.9	**47.5**±2.6	**39.6**±5.8	**70.0**±10.9	**71.2**±7.3	66.2±13.8	87.5±2.6
	0.99	0.0±0.0	2.5±6.25	5.0±0.0	0.0±0.0	35.0±30.0	2.5±6.25
Temperature (α)	5	**85.0**±10.0	**42.5**±16.2	67.5±12.5	70.0±25.0	65.0±225.0	**95.0**±0.0
	10	47.5±2.6	39.6±5.8	**70.0**±10.9	**71.2**±7.3	**66.2**±13.8	87.5±2.6
	20	**85.0**±25.0	37.5±12.5	67.5±12.5	60.0±32.5	55.0±20.0	90.0±0.0
Dropout (p)	0	44.5±2.6	39.6±5.8	70.0±10.9	78.2±7.3	**66.2**±13.8	**87.5**±2.6
	0.1	45.0±25.0	22.5±6.25	**87.5**±36.25	80.0±25.0	32.5±56.25	85.0±25.0
	0.3	**62.5**±26.25	**45.0**±0.0	72.5±6.25	**80.0**±9.4	37.5±12.25	80.0±0.0
	0.5	20.0±25.0	42.5±9.53	77.5±9.87	67.5±12.5	15±7.5	42.5±5.63
Batch size (B)	256	37.5±2.6	**39.6**±5.8	70.0±10.9	71.2±7.3	66.2±13.8	87.5±2.6
	512	**40.0**±25.0	35.0±20.0	**77.5**±16.25	**80.0**±3.2	60.0±22.5	85.0±25.0
	1024	37.5±16.25	27.5±6.25	67.5±16.25	77.5±6.25	**75.0**±20.0	**90.0**±1.25
network depth (d)	2	**47.5**±2.6	**39.6**±5.8	**70.0**±10.9	**71.2**±7.3	**66.2**±13.8	87.5±2.6
	3	47.2±6.25	35.0±15.0	30.0±10.0	37.5±6.25	52.5±12.5	**97.5**±12.5
	5	5.0±12.1	0.0±7.3	5.0±2.0	12.5±62.5	42.5±36.25	90.0±5.0
Learning rate (lr)	2e-4	43.0±16.4	30.0±25.0	**70.0**±25.0	67.5±12.5	**70.0**±3.0	**95.0**±4.2
	3e-4	**47.5**±2.6	**39.6**±5.8	**70.0**±10.9	71.2±7.3	66.2±13.8	87.5±2.6
	4e-4	42.5±42.5	15.0±0.0	67.5±22.5	**75.0**±0.0	50.0±20.0	90.0±6.5

Claim 3 (computation efficient). Consistent with the original paper, our reproducibility experiments confirm that IQL operates with greater computational efficiency than the baseline methods in JAX, shown in Tab. 2. This observation upholds the claim of IQL's computational efficiency, especially in environments where rapid processing is advantageous.

4.2 Ablation Study

Expectile τ. The experiment results shown in tab. 3 reveal that in AntMaze environments, the performance of IQL varies significantly with different Expectile (τ) values. Lower τ values (0.5 to 0.7) generally result in poor performance, especially in complex mazes. A notable improvement is observed at τ = 0.8, indicating a better balance between exploration and exploitation. The peak performance near τ = 0.9 aligns with the original paper's benchmark, suggesting it is the optimal setting for the AntMaze tasks, and Claim 2. However, a sharp decline at τ = 0.99 underscores a critical threshold, beyond which higher τ values adversely affect performance, particularly in challenging environments.

Temperature α. The experiment results of Temperature (α) across AntMaze environments shown in tab. 3 indicate a nuanced influence on performance. Lower α (5) generally enhances performance, especially in more complex mazes, suggesting better learning efficiency. The benchmark α (10) shows a mixed outcome, effective in some environments

but less so in others, questioning its universal optimality. Higher α (20) maintains effectiveness in some mazes but diminishes in others, highlighting the potential drawbacks of excessively high temperatures. This suggests the importance of carefully tuning α to balance exploration and exploitation for optimal performance in different maze complexities.

Dropout p. The performance across AntMaze environments with varying Dropout (p) parameters shown in tab. 3 indicates that the absence of dropout (p = 0), as benchmarked in the original paper, generally provides stable and decent performance across all environments. However, introducing dropout (p = 0.1 and 0.3) leads to mixed results, with some environments showing improvement (e.g., Medium-d with p = 0.1) and others showing a decline (e.g., Unmaze-d with p = 0.1). A higher dropout rate (p = 0.5) significantly decreases performance in most environments. This suggests that while a small amount of dropouts can be beneficial in certain contexts, too much can adversely affect the model's ability to learn effectively.

Batch size B. The variation of batch sizes shown in tab. 3 shows that the original benchmark of 256 offers a reliable performance across different environments. Increasing the batch size to 512 yields some improvements in specific tasks but not universally. A further increase to 1024 does not consistently enhance performance and can sometimes reduce it, indicating that a too-large batch size might not be as effective. This suggests that a moderate batch size, as chosen in the original study, generally provides a good balance for learning efficiency in various maze complexities.

Network depth d. In the experiment on the impact of hidden dimension network depth (d) on IQL performance in AntMaze environments shown in tab. 3, it is observed that the benchmark depth of 2 layers provides consistent, moderate to high performance across various mazes. Increasing the depth to 3 layers shows a mixed impact, with notable improvements in some environments but declines in others, suggesting that additional complexity doesn't uniformly benefit performance. A further increase to 5 layers leads to a significant drop in most environments, indicating that overly complex networks might hinder learning in this context. This suggests that a moderate level of network depth, as in the original benchmark, is generally optimal for diverse tasks.

Learning rate lr. In the experiment on the impact of Learning Rate (lr) on IQL performance in AntMaze tasks shown in tab. 3, the benchmark rate of 3e-4 shows balanced and effective performance across environments. Lowering the rate to 2e-4 results in a slight decrease in performance, indicating less optimal learning. Increasing to 4e-4 leads to more variability and lower performance in some tasks, suggesting that too high a learning rate can negatively impact the algorithm's efficiency. This indicates that the original benchmark learning rate provides a good balance for effective learning in diverse maze complexities.

5 Discussion

Practical methods for estimating various statistics of a random variable have been thoroughly studies in applied statistics and econometrics. The $\tau \in (0, 1)$ expectile of some random variable X is defined as a solution to the asymmetric least squares problem:

$$\arg\min_{m_\tau} \mathbb{E}_{x\sim X}\left[L_2^\tau\left(x - m_\tau\right)\right]$$

where $L_2^\tau(u) = |\tau - \mathbb{I}(u < 0)|u^2$.

That is, for $\tau > 0.5$, this asymmetric loss function downweights the contributions of x values smaller than m_τ while giving more weights to larger values.

Lemma 5.1. *Let X be a real-valued random variable with a bounded support and supremum of the support is x^*. Then,*

$$\lim_{\tau\to 1} m_\tau = x^*$$

Proof Sketch. One can show that expectiles of a random variable have the same supremum x^*. Moreover, for all τ_1 and τ_2 such that $\tau_1 < \tau_2$, we get $m_{\tau_1} \leq m_{\tau_2}$. Therefore, the limit follows from the properties of bounded monotonically non-decreasing functions. □

Theorem 5.2.

$$\lim_{\tau\to 1} V_\tau(s) = \max_{\substack{a\in\mathcal{A} \\ s.t.\ \pi_\beta(a|s)>0}} Q^*(s, a).$$

IQL states that it is allowed to achieve the optimal value function (V^*) by setting ($\tau \to 1$), as shown in 5.2. However, our experiment setup on the ablation study of τ reveals one disturbing issue of IQL, i.e., the role of τ does not have a perfect match between theory and practice. In theory, τ should be close to 1 to obtain an optimal policy while in

practice a larger τ may give a worse result, i.e., $\tau = 0.99$ in our ablation study. Further, we find a further paper, IVR [7], gives a deeper understanding of how IQL handles the distributional shift: it is doing implicit value regularization, with the hyperparameter τ to control the strength. The analysis follows the behavior-regularized MDP problem:

$$\max_{\pi} \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t \left(r\left(s_t, a_t\right) - \alpha \cdot f\left(\frac{\pi\left(a_t \mid s_t\right)}{\mu\left(a_t \mid s_t\right)}\right)\right)\right],$$

where $f(\cdot)$ is a regularization function.

6 Conclusions and Future Work

In conclusion, our reproducibility report effectively substantiates the original claims regarding IQL, i.e., performance superior, requiring large expectile, and computation efficient. However, it also uncovers critical theoretical-practical misalignments in IQL, particularly with the expectile parameter τ at extreme values. These findings highlight the need for a more throughout understanding of IQL's underlying mechanisms and its practical implications. Future research directions should therefore focus on integrating theoretical insights with empirical observations to refine and enhance offline RL models. This endeavor should include exploring a broader range of hyperparameters to optimize performance and understand the dynamics of offline RL algorithms in various scenarios.

7 Contribution

In this project, Li Jiang was primarily in charge of coding and setting up the experiment, while Jiahang Wang and Liting Chen focused on analyzing the results and writing the report. Their collaborative efforts ensured a well-rounded execution of the project, from technical development to comprehensive documentation.

References

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

[2] Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. Openai gym. *arXiv preprint arXiv:1606.01540*, 2016.

[3] Justin Fu, Aviral Kumar, Ofir Nachum, George Tucker, and Sergey Levine. D4rl: Datasets for deep data-driven reinforcement learning. *arXiv preprint arXiv:2004.07219*, 2020.

[4] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proc. of ICLR*, 2015.

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

[6] Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ international conference on intelligent robots and systems*, pages 5026–5033. IEEE, 2012.

[7] Haoran Xu, Li Jiang, Jianxiong Li, Zhuoran Yang, Zhaoran Wang, Victor Wai Kin Chan, and Xianyuan Zhan. Offline rl with no ood actions: In-sample learning via implicit value regularization. *arXiv preprint arXiv:2303.15810*, 2023.