Offline Reinforcement Learning with Implicit Q-Learning

- Ilya Kostrikov, Ashvin Nair and Sergey Levine

Ayush Sawarni (19776), Piyush Tiwary (19897), Manan Tayal (20133)

Department of Computer Science and Automation
Department of Electrical Communication Engineering
Robert Bosch Center for Cyber Physical Systems

August 9, 2022

Table of Contents

What is Offline RL?

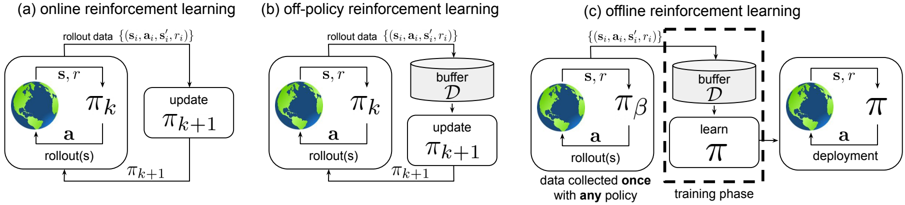


Figure: Different paradigms of Reinforcement Learning [Levine et al., 2020]

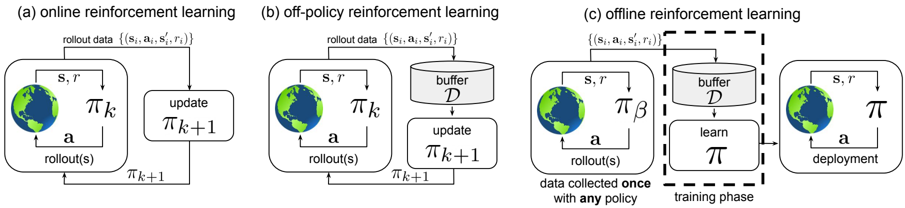


Figure: Different paradigms of Reinforcement Learning [Levine et al., 2020]

- Why is it hard?
  1. There is no feedback from the environment.
  2. The distribution on which we minimize our loss comes from the behaviour policy (π_β) however, the distribution over which we will run the policy will be a new policy (π_θ)

What is Offline RL?

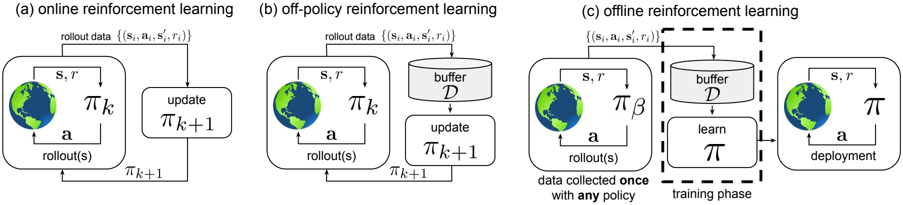


Figure: Different paradigms of Reinforcement Learning [Levine et al., 2020]

- Why is it hard?
  1. There is no feedback from the environment.
  2. The distribution on which we minimize our loss comes from the behaviour policy (π_β) however, the distribution over which we will run the policy will be a new policy (π_θ)
- Previous approaches in the literature to solve this problem can be broadly classified into 2 types:
  1. Distribution Constraint based
  2. Support Constraint based

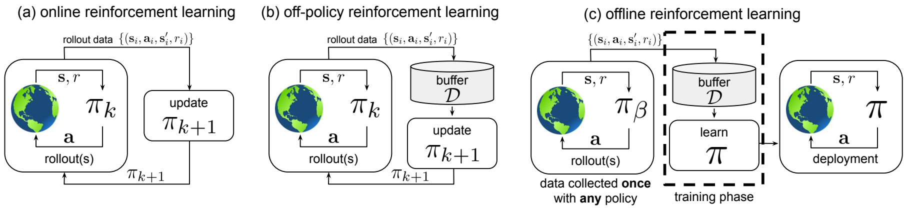


Figure: Different paradigms of Reinforcement Learning [Levine et al., 2020]

- Why is it hard?
  1. There is no feedback from the environment.
  2. The distribution on which we minimize our loss comes from the behaviour policy (π_β) however, the distribution over which we will run the policy will be a new policy (π_θ)
- Previous approaches in the literature to solve this problem can be broadly classified into 2 types:
  1. Distribution Constraint based
  2. Support Constraint based ✓

Proposed Methodology

Offline reinforcement learning objective

$$L_{TD}(\theta) = \mathbb{E}_{(s,a,s')\sim\mathcal{D}}\left[\left(r(s,a) + \gamma \max_{a'} Q_{\hat{\theta}}\left(s', a'\right) - Q_{\theta}(s,a)\right)^2\right]$$
$$\theta^* = \arg\min_{\theta} L_{TD}(\theta) \quad (1)$$

Proposed Methodology

Offline reinforcement learning objective

$$L_{TD}(\theta) = \mathbb{E}_{(s,a,s')\sim\mathcal{D}}\left[\left(r(s,a) + \gamma \max_{a'} Q_{\hat{\theta}}\left(s',a'\right) - Q_{\theta}(s,a)\right)^2\right]$$
$$\theta^* = \arg\min_{\theta} L_{TD}(\theta) \quad (1)$$

- Problem: Q_θ obtained using the above objective would give arbitrary values.

Proposed Methodology

Offline reinforcement learning objective

$$L_{TD}(\theta) = \mathbb{E}_{(s,a,s')\sim\mathcal{D}}\left[\left(r(s,a) + \gamma \max_{a'} Q_{\hat{\theta}}\left(s', a'\right) - Q_\theta(s,a)\right)^2\right]$$
$$\theta^* = \arg\min_\theta L_{TD}(\theta) \quad (1)$$

- Problem: Q_θ obtained using the above objective would give arbitrary values.
  Why?: $\max_{a'} Q_{\hat{\theta}}(s', a')$ is not constrained to in-dataset actions, hence maximization over unseen actions would give arbitrary values.

Proposed Methodology

Offline reinforcement learning objective

$$L_{TD}(\theta) = \mathbb{E}_{(s,a,s')\sim\mathcal{D}}\left[\left(r(s,a) + \gamma \max_{a'} Q_{\hat{\theta}}\left(s',a'\right) - Q_{\theta}(s,a)\right)^2\right]$$
$$\theta^* = \arg\min_{\theta} L_{TD}(\theta) \tag{1}$$

- Problem: Q_θ obtained using the above objective would give arbitrary values.
  Why?: $\max_{a'} Q_{\hat{\theta}}(s',a')$ is not constrained to in-dataset actions, hence maximization over unseen actions would give arbitrary values.
- In other words, this maximization may lead to cases where for a given state, unseen actions have the maximum Q value.

Proposed Methodology

- SARSA-like objective function:

$$L(\theta) = \mathbb{E}_{(s,a,s',a') \sim \mathcal{D}} \left[\left(r(s,a) + \gamma Q_{\hat{\theta}}(s',a') - Q_{\theta}(s,a) \right)^2 \right]$$

won't suffer from this problem as the tuple (s, a, s', a') comes from dataset itself.

Proposed Methodology

- SARSA-like objective function:

$$L(\theta) = \mathbb{E}_{(s,a,s',a')\sim\mathcal{D}}\left[\left(r(s,a) + \gamma Q_{\hat{\theta}}(s',a') - Q_{\theta}(s,a)\right)^2\right]$$

won't suffer from this problem as the tuple (s, a, s', a') comes from dataset itself. But it learns the **behavior policy** value function, not the optimal value function

Proposed Methodology

- SARSA-like objective function:

$$L(\theta) = \mathbb{E}_{(s,a,s',a')\sim\mathcal{D}}\left[\left(r(s,a) + \gamma Q_{\hat{\theta}}(s',a') - Q_{\theta}(s,a)\right)^2\right]$$

  won't suffer from this problem as the tuple (s, a, s', a') comes from dataset itself. But it learns the **behavior policy** value function, not the optimal value function

- IDEA: Combine these 2 IDEAs i.e, while computing $\max_{a'} Q_{\hat{\theta}}(s', a')$ choose a' which are in-dataset.

Proposed Methodology

- SARSA-like objective function:

$$L(\theta) = \mathbb{E}_{(s,a,s',a')\sim\mathcal{D}}\left[\left(r(s,a) + \gamma Q_{\hat{\theta}}(s',a') - Q_\theta(s,a)\right)^2\right]$$

  won't suffer from this problem as the tuple (s, a, s', a') comes from dataset itself. But it learns the **behavior policy** value function, not the optimal value function

- IDEA: Combine these 2 IDEAs i.e, while computing $\max_{a'} Q_{\hat{\theta}}(s', a')$ choose a' which are in-dataset.

Proposed loss function

$$L(\theta) = \mathbb{E}_{(s,a,s')\sim\mathcal{D}}\left[\left(r(s,a) + \gamma \max_{\substack{a'\in\mathcal{A} \\ \text{s.t. } \pi_\beta(a'|s')>0}} Q_{\hat{\theta}}(s',a') - Q_\theta(s,a)\right)^2\right] \tag{2}$$

Proposed Methodology

- SARSA-like objective function:

$$L(\theta) = \mathbb{E}_{(s,a,s',a')\sim\mathcal{D}}\left[\left(r(s,a) + \gamma Q_{\hat{\theta}}(s',a') - Q_\theta(s,a)\right)^2\right]$$

  won't suffer from this problem as the tuple (s, a, s', a') comes from dataset itself. But it learns the **behavior policy** value function, not the optimal value function

- IDEA: Combine these 2 IDEAs i.e, while computing $\max_{a'} Q_{\hat{\theta}}(s', a')$ choose a' which are in-dataset.

Proposed loss function

$$L(\theta) = \mathbb{E}_{(s,a,s')\sim\mathcal{D}}\left[\left(r(s,a) + \gamma \max_{\substack{a'\in\mathcal{A} \\ \text{s.t. } \pi_\beta(a'|s')>0}} Q_{\hat{\theta}}(s',a') - Q_\theta(s,a)\right)^2\right] \quad (2)$$

- We approximate "$\max_{\substack{a'\in\mathcal{A} \\ \text{s.t. } \pi_\beta(a'|s')>0}} Q_{\hat{\theta}}(s',a')$" using Expectile Regression.

Expectile Regression

Definition

The $\tau \in (0,1)$ expectile of some random variable X is defined as a solution to the asymmetric least squares problem:

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

where $L_2^\tau(u) = |\tau - \mathbb{1}(u < 0)|u^2$.

Expectile Regression

Definition

The $\tau \in (0, 1)$ expectile of some random variable X is defined as a solution to the asymmetric least squares problem:

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

where $L_2^\tau(u) = |\tau - \mathbb{1}(u < 0)| u^2$.

INTUITION: For $(\tau > 0.5)$, punish positive errors (underestimation) more than negative errors, using an expectile loss. The larger the expectile (τ) the closer this is to the maximum.

Expectile Regression

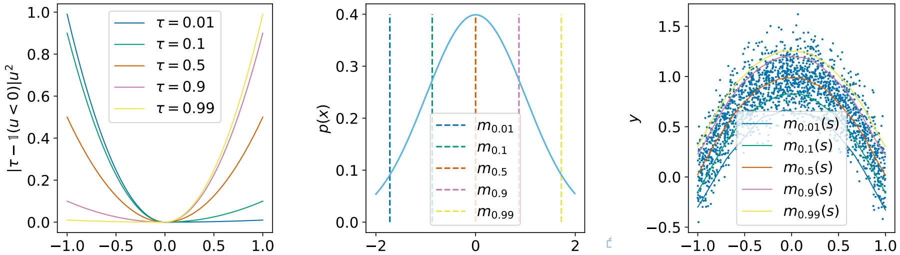
Definition

The $\tau \in (0, 1)$ expectile of some random variable X is defined as a solution to the asymmetric least squares problem:

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

where $L_2^\tau(u) = |\tau - \mathbb{1}(u < 0)|u^2$.

INTUITION: For $(\tau > 0.5)$, punish positive errors (underestimation) more than negative errors, using an expectile loss. The larger the expectile (τ) the closer this is to the maximum.



Proposed Methodology

- IDEA: Approximate $\max\limits_{\substack{a' \in \mathcal{A} \\ \text{s.t. } \pi_\beta\left(a'|s'\right)>0}} Q_{\dot{\theta}}\left(s', a'\right)$ by using Expectile Regression.

Proposed Methodology

- IDEA: Approximate $\max_{\substack{a' \in \mathcal{A} \\ \text{s.t. } \pi_\beta(a'|s')>0}} Q_{\dot{\theta}}(s', a')$ by using Expectile Regression.

In particular:

Value function loss

$$L_V(\psi) = \mathbb{E}_{(s,a)\sim\mathcal{D}}\left[L_2^\tau(Q_\theta(s,a) - V_\psi(s))\right] \tag{3}$$

Proposed Methodology

- IDEA: Approximate $\max\limits_{\substack{a' \in \mathcal{A} \\ \text{s.t. } \pi_\beta(a'|s')>0}} Q_{\hat{\theta}}(s', a')$ by using Expectile Regression.

In particular:

Value function loss

$$L_V(\psi) = \mathbb{E}_{(s,a)\sim\mathcal{D}}[L_2^\tau(Q_\theta(s,a) - V_\psi(s))] \tag{3}$$

- Hence, if the value of τ is kept high (≈ 1),

$$V_\psi(s') \approx \max_{\substack{a' \in \mathcal{A} \\ \text{s.t. } \pi_\beta(a'|s')>0}} Q_\theta(s', a')$$

Proposed Methodology

- IDEA: Approximate $\max_{\substack{a' \in \mathcal{A} \\ \text{s.t. } \pi_\beta(a'|s')>0}} Q_{\hat{\theta}}(s', a')$ by using Expectile Regression.

In particular:

Value function loss

$$L_V(\psi) = \mathbb{E}_{(s,a)\sim\mathcal{D}}[L_2^\tau(Q_\theta(s,a) - V_\psi(s))] \tag{3}$$

- Hence, if the value of τ is kept high (≈ 1),

$$V_\psi(s') \approx \max_{\substack{a' \in \mathcal{A} \\ \text{s.t. } \pi_\beta(a'|s')>0}} Q_\theta(s', a')$$

- Now, this value can be substituted in our original loss function (2) to get:

Proposed Methodology

- IDEA: Approximate $\max\limits_{\substack{a' \in \mathcal{A} \\ \text{s.t. } \pi_\beta(a'|s')>0}} Q_{\dot{\theta}}(s', a')$ by using Expectile Regression.

In particular:

Value function loss

$$L_V(\psi) = \mathbb{E}_{(s,a)\sim\mathcal{D}}\left[L_2^\tau\left(Q_\theta(s,a) - V_\psi(s)\right)\right] \tag{3}$$

- Hence, if the value of τ is kept high (≈ 1),

$$V_\psi(s') \approx \max_{\substack{a' \in \mathcal{A} \\ \text{s.t. } \pi_\beta(a'|s')>0}} Q_\theta(s', a')$$

- Now, this value can be substituted in our original loss function (2) to get:

Q-Value function loss

$$L(\theta) = \mathbb{E}_{(s,a,s')\sim\mathcal{D}}\left[\left(r(s,a) + \gamma V_\psi(s') - Q_\theta(s,a)\right)^2\right] \tag{4}$$

Policy Extraction & Algorithm

Algorithm

Algorithm Implicit Q-learning

```
Initialize parameters ψ, θ, θ̂, φ.
TD learning (IQL):
for each gradient step do
    ψ ← ψ − λ_V ∇_ψ L_V(ψ)
    θ ← θ − λ_Q ∇_θ L_Q(θ)
    θ̂ ← (1 − α)θ̂ + αθ
end for
Policy extraction (AWR):
for each gradient step do
    φ ← φ + λ_π ∇_φ L_π(φ)
end for
```

Policy Extraction & Algorithm

- By design, IQL does not give us an optimal policy. A policy extraction step is required to learn an optimal policy from the learnt Q function.

Algorithm

Algorithm Implicit Q-learning

1: Initialize parameters $\psi, \theta, \hat{\theta}, \phi$.
2: TD learning (IQL):
3: **for** each gradient step **do**
4: $\quad \psi \leftarrow \psi - \lambda_V \nabla_\psi L_V(\psi)$
5: $\quad \theta \leftarrow \theta - \lambda_Q \nabla_\theta L_Q(\theta)$
6: $\quad \hat{\theta} \leftarrow (1-\alpha)\hat{\theta} + \alpha\theta$
7: **end for**
8: Policy extraction (AWR):
9: **for** each gradient step **do**
10: $\quad \phi \leftarrow \phi + \lambda_\pi \nabla_\phi L_\pi(\phi)$
11: **end for**

Policy Extraction & Algorithm

- By design, IQL does not give us an optimal policy. A policy extraction step is required to learn an optimal policy from the learnt Q function.
- Authors have used Advantage Weighted Regression which finds an optimal policy by maximizing the following objective.

$$L_\pi(\phi) = \mathbb{E}_{(s,a)\sim\mathcal{D}}\Big[\exp\left(\beta\left(Q_{\hat{\theta}}(s,a) - V_\psi(s)\right)\right) \times \log \pi_\phi(a \mid s)\Big]$$

Algorithm

Algorithm Implicit Q-learning

1: Initialize parameters $\psi, \theta, \hat{\theta}, \phi$.
2: TD learning (IQL):
3: **for** each gradient step **do**
4: $\quad \psi \leftarrow \psi - \lambda_V \nabla_\psi L_V(\psi)$
5: $\quad \theta \leftarrow \theta - \lambda_Q \nabla_\theta L_Q(\theta)$
6: $\quad \hat{\theta} \leftarrow (1-\alpha)\hat{\theta} + \alpha\theta$
7: **end for**
8: Policy extraction (AWR):
9: **for** each gradient step **do**
10: $\quad \phi \leftarrow \phi + \lambda_\pi \nabla_\phi L_\pi(\phi)$
11: **end for**

Policy Extraction & Algorithm

- By design, IQL does not give us an optimal policy. A policy extraction step is required to learn an optimal policy from the learnt Q function.
- Authors have used Advantage Weighted Regression which finds an optimal policy by maximizing the following objective.

$$L_\pi(\phi) = \mathbb{E}_{(s,a)\sim\mathcal{D}}\Big[\exp\left(\beta\left(Q_{\hat{\theta}}(s,a) - V_\psi(s)\right)\right) \times \log \pi_\phi(a \mid s)\Big]$$

- This is similar to Cross-Entropy loss where $\beta \in [0, \infty)$ is a hyper-parameter.

Algorithm

Algorithm Implicit Q-learning

1: Initialize parameters $\psi, \theta, \hat{\theta}, \phi$.
2: TD learning (IQL):
3: **for** each gradient step **do**
4: $\quad \psi \leftarrow \psi - \lambda_V \nabla_\psi L_V(\psi)$
5: $\quad \theta \leftarrow \theta - \lambda_Q \nabla_\theta L_Q(\theta)$
6: $\quad \hat{\theta} \leftarrow (1-\alpha)\hat{\theta} + \alpha\theta$
7: **end for**
8: Policy extraction (AWR):
9: **for** each gradient step **do**
10: $\quad \phi \leftarrow \phi + \lambda_\pi \nabla_\phi L_\pi(\phi)$
11: **end for**

Theoretical Analysis

Notations

- $V_\tau(s) = \mathbb{E}^{\tau}_{a\sim\mu(\cdot|a)}\left[Q_\tau(s,a)\right]$
- $Q_\tau(s,a) = r(s,a) + \gamma\mathbb{E}_{s'\sim p(\cdot|s,a)}\left[V_\tau(s')\right]$
- m_τ : τ^{th} expectile
- $\mathbb{E}^{\tau}_{a\sim\mu(\cdot|a)}(\cdot)$: Expectile operator

Theoretical Analysis

Lemma 1

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

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

Theoretical Analysis

Lemma 1

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

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

Proof

By the definition of m_τ, we have the following property:

$$m_\tau - \mu = \frac{(2\tau - 1)}{1 - \tau} \int_{m_\tau}^{+\infty} (x - m_\tau) dF(x)$$

where $F(x)$ is distribution function of X and $\mu = \mathbb{E}[X]$.

Theoretical Analysis

Lemma 1

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

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

Proof

By the definition of m_τ, we have the following property:

$$m_\tau - \mu = \frac{(2\tau - 1)}{1 - \tau} \int_{m_\tau}^{+\infty} (x - m_\tau) dF(x)$$

where $F(x)$ is distribution function of X and $\mu = \mathbb{E}[X]$.
Let $T(m) = \int_m^{+\infty} (x - m) dF(x)$ and $\alpha(\tau) = \frac{(2\tau - 1)}{(1 - \tau)}$.

Theoretical Analysis

Lemma 1

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

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

Proof

By the definition of m_τ, we have the following property:

$$m_\tau - \mu = \frac{(2\tau - 1)}{1 - \tau} \int_{m_\tau}^{+\infty} (x - m_\tau) dF(x)$$

where $F(x)$ is distribution function of X and $\mu = \mathbb{E}[X]$.
Let $T(m) = \int_m^{+\infty} (x - m) dF(x)$ and $\alpha(\tau) = \frac{(2\tau - 1)}{(1 - \tau)}$. It can be shown that $T(m)$ is a convex function in m.

Proof (Contd.)

Now, $T'(m) = \frac{dT(m)}{dm} = \int_m^\infty \frac{d}{dm}(x-m)dF(x) = \int_m^\infty (-1)dF(x) = F(m) - 1$

$$\implies \lim_{m\to\infty} T'(m) = 0 \quad \left[\text{Since,} \quad \lim_{m\to\infty} F(m) = 1\right]$$

Proof (Contd.)

Now, $T'(m) = \frac{dT(m)}{dm} = \int_m^\infty \frac{d}{dm}(x-m)dF(x) = \int_m^\infty (-1)dF(x) = F(m) - 1$

$$\implies \lim_{m\to\infty} T'(m) = 0 \quad \left[\text{Since,} \quad \lim_{m\to\infty} F(m) = 1\right]$$

Since, $T(m)$ is convex $\implies T''(m) \geq 0 \implies T'(m)$ is non-decreasing.

Proof (Contd.)

Now, $T'(m) = \frac{dT(m)}{dm} = \int_m^\infty \frac{d}{dm}(x-m)dF(x) = \int_m^\infty (-1)dF(x) = F(m) - 1$

$$\implies \lim_{m \to \infty} T'(m) = 0 \quad \left[\text{Since,} \quad \lim_{m \to \infty} F(m) = 1\right]$$

Since, $T(m)$ is convex $\implies T''(m) \geq 0 \implies T'(m)$ is non-decreasing. Now, $\lim_{m \to \infty} T'(m) = 0 \implies T'(m) \leq 0 \ \forall m \implies$ T(m) is non-increasing, i.e for $m_1 > m_2$, $T(m_1) \leq T(m_2)$.

Proof (Contd.)

Now, $T'(m) = \frac{dT(m)}{dm} = \int_m^\infty \frac{d}{dm}(x-m)dF(x) = \int_m^\infty (-1)dF(x) = F(m) - 1$

$$\implies \lim_{m \to \infty} T'(m) = 0 \quad \left[\text{Since,} \quad \lim_{m \to \infty} F(m) = 1\right]$$

Since, $T(m)$ is convex $\implies T''(m) \geq 0 \implies T'(m)$ is non-decreasing. Now, $\lim_{m \to \infty} T'(m) = 0 \implies T'(m) \leq 0 \ \forall m \implies$ T(m) is non-increasing, i.e for $m_1 > m_2$, $T(m_1) \leq T(m_2)$.

Now, $\alpha(\tau) > -1$ and $\frac{d\alpha(\tau)}{d\tau} = \frac{1}{(1-\tau)^2} > 0 \implies \alpha(\tau)$ is monotonically increasing in τ.

Now, using the fact that $\alpha(\tau)$ is monotonically increasing in τ and $T(m)$ is monotonically decreasing in m, one can show that $\alpha(\tau)T(m_\tau)$ is monotonically increasing in τ.

Proof (Contd.)

Now, $T'(m) = \frac{dT(m)}{dm} = \int_m^\infty \frac{d}{dm}(x-m)dF(x) = \int_m^\infty (-1)dF(x) = F(m) - 1$

$$\implies \lim_{m\to\infty} T'(m) = 0 \quad \left[\text{Since,} \quad \lim_{m\to\infty} F(m) = 1\right]$$

Since, $T(m)$ is convex $\implies T''(m) \geq 0 \implies T'(m)$ is non-decreasing. Now, $\lim_{m\to\infty} T'(m) = 0 \implies T'(m) \leq 0 \ \forall m \implies$ T(m) is non-increasing, i.e for $m_1 > m_2$, $T(m_1) \leq T(m_2)$.

Now, $\alpha(\tau) > -1$ and $\frac{d\alpha(\tau)}{d\tau} = \frac{1}{(1-\tau)^2} > 0 \implies \alpha(\tau)$ is monotonically increasing in τ.

Now, using the fact that $\alpha(\tau)$ is monotonically increasing in τ and $T(m)$ is monotonically decreasing in m, one can show that $\alpha(\tau)T(m_\tau)$ is monotonically increasing in τ. Using this, one can write for $\tau_1 < \tau_2 < \ldots < \tau_n$:

$$m_{\tau_1} \leq m_{\tau_2} \leq \ldots \leq m_{\tau_n}$$

Proof (Contd.)

Now, $T'(m) = \frac{dT(m)}{dm} = \int_m^\infty \frac{d}{dm}(x-m)dF(x) = \int_m^\infty(-1)dF(x) = F(m) - 1$

$$\implies \lim_{m\to\infty} T'(m) = 0 \quad \left[\text{Since,} \quad \lim_{m\to\infty} F(m) = 1\right]$$

Since, $T(m)$ is convex $\implies T''(m) \geq 0 \implies T'(m)$ is non-decreasing. Now, $\lim_{m\to\infty} T'(m) = 0 \implies T'(m) \leq 0 \;\; \forall m \implies$ T(m) is non-increasing, i.e for $m_1 > m_2$, $T(m_1) \leq T(m_2)$.

Now, $\alpha(\tau) > -1$ and $\frac{d\alpha(\tau)}{d\tau} = \frac{1}{(1-\tau)^2} > 0 \implies \alpha(\tau)$ is monotonically increasing in τ.

Now, using the fact that $\alpha(\tau)$ is monotonically increasing in τ and $T(m)$ is monotonically decreasing in m, one can show that $\alpha(\tau)T(m_\tau)$ is monotonically increasing in τ. Using this, one can write for $\tau_1 < \tau_2 < \ldots < \tau_n$:

$$m_{\tau_1} \leq m_{\tau_2} \leq \ldots \leq m_{\tau_n}$$

Also, since m_τ is upper bounded by $\sup x = x^*$, $\{m_{\tau_i}\}_{i=1}^n$ is bounded monotonically non-decreasing sequence.

Proof (Contd.)

Now, $T'(m) = \frac{dT(m)}{dm} = \int_m^\infty \frac{d}{dm}(x-m)dF(x) = \int_m^\infty (-1)dF(x) = F(m) - 1$

$$\implies \lim_{m \to \infty} T'(m) = 0 \quad \left[\text{Since,} \quad \lim_{m\to\infty} F(m) = 1\right]$$

Since, $T(m)$ is convex $\implies T''(m) \geq 0 \implies T'(m)$ is non-decreasing. Now, $\lim\limits_{m\to\infty} T'(m) = 0 \implies T'(m) \leq 0 \;\; \forall m \implies$ T(m) is non-increasing, i.e for $m_1 > m_2$, $T(m_1) \leq T(m_2)$.

Now, $\alpha(\tau) > -1$ and $\frac{d\alpha(\tau)}{d\tau} = \frac{1}{(1-\tau)^2} > 0 \implies \alpha(\tau)$ is monotonically increasing in τ.

Now, using the fact that $\alpha(\tau)$ is monotonically increasing in τ and $T(m)$ is monotonically decreasing in m, one can show that $\alpha(\tau)T(m_\tau)$ is monotonically increasing in τ. Using this, one can write for $\tau_1 < \tau_2 < \ldots < \tau_n$:

$$m_{\tau_1} \leq m_{\tau_2} \leq \ldots \leq m_{\tau_n}$$

Also, since m_τ is upper bounded by $\sup x = x^*$, $\{m_{\tau_i}\}_{i=1}^n$ is bounded monotonically non-decreasing sequence.Hence, we have the following limit:

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

Theoretical Analysis

Lemma 2

For all s, τ_1 and τ_2 such that $\tau_1 < \tau_2$ we get

$$V_{\tau_1}(s) \leq V_{\tau_2}(s) \tag{2}$$

Theoretical Analysis

Lemma 2

For all s, τ_1 and τ_2 such that $\tau_1 < \tau_2$ we get

$$V_{\tau_1}(s) \leq V_{\tau_2}(s) \tag{2}$$

Proof

$$V_{\tau_1}(s) = \mathbb{E}^{\tau_1}_{a \sim \mu(\cdot|s)} \left[r(s, a) + \gamma \mathbb{E}_{s' \sim p(\cdot|s,a)} \left[V_{\tau_1}\left(s'\right) \right] \right]$$

Theoretical Analysis

Lemma 2

For all s, τ_1 and τ_2 such that $\tau_1 < \tau_2$ we get

$$V_{\tau_1}(s) \leq V_{\tau_2}(s) \tag{2}$$

Proof

$$\begin{aligned} V_{\tau_1}(s) &= \mathbb{E}^{\tau_1}_{a\sim\mu(\cdot|s)}\left[r(s,a) + \gamma\mathbb{E}_{s'\sim p(\cdot|s,a)}\left[V_{\tau_1}\left(s'\right)\right]\right] \\ &\leq \mathbb{E}^{\tau_2}_{a\sim\mu(\cdot|s)}\left[r(s,a) + \gamma\mathbb{E}_{s'\sim p(\cdot|s,a)}\left[V_{\tau_1}\left(s'\right)\right]\right] \quad \text{[Using Lemma 1]} \end{aligned}$$

Theoretical Analysis

Lemma 2

For all s, τ_1 and τ_2 such that $\tau_1 < \tau_2$ we get

$$V_{\tau_1}(s) \leq V_{\tau_2}(s) \tag{2}$$

Proof

$$\begin{aligned}
V_{\tau_1}(s) &= \mathbb{E}^{\tau_1}_{a\sim\mu(\cdot|s)}\left[r(s,a) + \gamma\mathbb{E}_{s'\sim p(\cdot|s,a)}\left[V_{\tau_1}\left(s'\right)\right]\right] \\
&\leq \mathbb{E}^{\tau_2}_{a\sim\mu(\cdot|s)}\left[r(s,a) + \gamma\mathbb{E}_{s'\sim p(\cdot|s,a)}\left[V_{\tau_1}\left(s'\right)\right]\right] \quad \text{[Using Lemma 1]} \\
&= \mathbb{E}^{\tau_2}_{a\sim\mu(\cdot|s)}\left[r(s,a) + \gamma\mathbb{E}_{s'\sim p(\cdot|s,a)}\mathbb{E}^{\tau_1}_{a'\sim\mu(\cdot|s')}\left[r\left(s',a'\right) + \gamma\mathbb{E}_{s''\sim p(\cdot|s',a')}\left[V_{\tau_1}\left(s''\right)\right]\right]\right]
\end{aligned}$$

Theoretical Analysis

Lemma 2

For all s, τ_1 and τ_2 such that $\tau_1 < \tau_2$ we get

$$V_{\tau_1}(s) \leq V_{\tau_2}(s) \tag{2}$$

Proof

$$\begin{aligned}
V_{\tau_1}(s) &= \mathbb{E}^{\tau_1}_{a\sim\mu(\cdot|s)}\left[r(s,a) + \gamma\mathbb{E}_{s'\sim p(\cdot|s,a)}\left[V_{\tau_1}\left(s'\right)\right]\right] \\
&\leq \mathbb{E}^{\tau_2}_{a\sim\mu(\cdot|s)}\left[r(s,a) + \gamma\mathbb{E}_{s'\sim p(\cdot|s,a)}\left[V_{\tau_1}\left(s'\right)\right]\right] \quad \text{[Using Lemma 1]} \\
&= \mathbb{E}^{\tau_2}_{a\sim\mu(\cdot|s)}\left[r(s,a) + \gamma\mathbb{E}_{s'\sim p(\cdot|s,a)}\mathbb{E}^{\tau_1}_{a'\sim\mu(\cdot|s')}\left[r\left(s',a'\right) + \gamma\mathbb{E}_{s''\sim p(\cdot|s',a')}\left[V_{\tau_1}\left(s''\right)\right]\right]\right] \\
&\leq \mathbb{E}^{\tau_2}_{a\sim\mu(\cdot|s)}\left[r(s,a) + \gamma\mathbb{E}_{s'\sim p(\cdot|s,a)}\mathbb{E}^{\tau_2}_{a'\sim\mu(\cdot|s')}\left[r\left(s',a'\right) + \gamma\mathbb{E}_{s''\sim p(\cdot|s',a')}\left[V_{\tau_1}\left(s''\right)\right]\right]\right]
\end{aligned}$$

Theoretical Analysis

Lemma 2

For all s, τ_1 and τ_2 such that $\tau_1 < \tau_2$ we get

$$V_{\tau_1}(s) \leq V_{\tau_2}(s) \tag{2}$$

Proof

$$\begin{aligned}
V_{\tau_1}(s) &= \mathbb{E}^{\tau_1}_{a\sim\mu(\cdot|s)}\left[r(s,a) + \gamma \mathbb{E}_{s'\sim p(\cdot|s,a)}\left[V_{\tau_1}\left(s'\right)\right]\right] \\
&\leq \mathbb{E}^{\tau_2}_{a\sim\mu(\cdot|s)}\left[r(s,a) + \gamma \mathbb{E}_{s'\sim p(\cdot|s,a)}\left[V_{\tau_1}\left(s'\right)\right]\right] \quad \text{[Using Lemma 1]} \\
&= \mathbb{E}^{\tau_2}_{a\sim\mu(\cdot|s)}\left[r(s,a) + \gamma \mathbb{E}_{s'\sim p(\cdot|s,a)} \mathbb{E}^{\tau_1}_{a'\sim\mu(\cdot|s')}\left[r\left(s',a'\right) + \gamma \mathbb{E}_{s''\sim p(\cdot|s',a')}\left[V_{\tau_1}\left(s''\right)\right]\right]\right] \\
&\leq \mathbb{E}^{\tau_2}_{a\sim\mu(\cdot|s)}\left[r(s,a) + \gamma \mathbb{E}_{s'\sim p(\cdot|s,a)} \mathbb{E}^{\tau_2}_{a'\sim\mu(\cdot|s')}\left[r\left(s',a'\right) + \gamma \mathbb{E}_{s''\sim p(\cdot|s',a')}\left[V_{\tau_1}\left(s''\right)\right]\right]\right] \\
&\vdots \\
&\leq V_{\tau_2}(s)
\end{aligned}$$

Theoretical Analysis

Corollary 2.1

For any τ and s we have

$$V_\tau(s) \leq \max_{\substack{a \in \mathcal{A} \\ s.t\ \pi_\beta(a|s)>0}} Q^*(s,a) \tag{3}$$

where $V_\tau(s)$ is as defined earlier and $Q^*(s,a)$ is an optimal state-action value function constrained to the dataset and defined as:

$$Q^*(s,a) = r(s,a) + \gamma \mathbb{E}_{s' \sim p(\cdot|s,a)} \left[\max_{\substack{a' \in \mathcal{A} \\ s.t.\ \pi_\beta(a'|s')>0}} Q^*(s',a') \right]$$

Theoretical Analysis

Corollary 2.1

For any τ and s we have

$$V_\tau(s) \leq \max_{\substack{a \in \mathcal{A} \\ s.t \ \pi_\beta(a|s) > 0}} Q^*(s, a) \tag{3}$$

where $V_\tau(s)$ is as defined earlier and $Q^*(s, a)$ is an optimal state-action value function constrained to the dataset and defined as:

$$Q^*(s, a) = r(s, a) + \gamma \mathbb{E}_{s' \sim p(\cdot|s,a)} \left[\max_{\substack{a' \in \mathcal{A} \\ s.t. \ \pi_\beta(a'|s') > 0}} Q^*(s', a') \right]$$

Proof

From the definition of $V_\tau(s)$:

$$V_\tau(s) = \mathbb{E}^\tau_{a \sim \pi_\beta(\cdot|s)} [Q_\tau(s, a)] \leq \max_{\substack{a \in \mathcal{A} \\ s.t. \ \pi_\beta(a|s) > 0}} Q_\tau(s, a) \qquad \text{[Property of Expectile]}$$

Theoretical Analysis

Corollary 2.1

For any τ and s we have

$$V_\tau(s) \leq \max_{\substack{a \in \mathcal{A} \\ s.t\ \pi_\beta(a|s)>0}} Q^*(s,a) \tag{3}$$

where $V_\tau(s)$ is as defined earlier and $Q^*(s,a)$ is an optimal state-action value function constrained to the dataset and defined as:

$$Q^*(s,a) = r(s,a) + \gamma \mathbb{E}_{s' \sim p(\cdot|s,a)} \left[\max_{\substack{a' \in \mathcal{A} \\ s.t.\ \pi_\beta(a'|s')>0}} Q^*(s',a') \right]$$

Proof

From the definition of $V_\tau(s)$:

$$V_\tau(s) = \mathbb{E}^\tau_{a \sim \pi_\beta(\cdot|s)} [Q_\tau(s,a)] \leq \max_{\substack{a \in \mathcal{A} \\ s.t.\ \pi_\beta(a|s)>0}} Q_\tau(s,a) \qquad \text{[Property of Expectile]}$$

$$\leq \max_{\substack{a \in \mathcal{A} \\ s.t.\ \pi_\beta(a|s)>0}} Q^*(s,a) \qquad \text{[Since } Q^* \text{ is optimal Q-function]}$$

Theoretical Analysis

Theorem

$$\lim_{\tau \to 1} V_\tau(s) = \max_{\substack{a \in \mathcal{A} \\ s.t\ \pi_\beta(a|s) > 0}} Q^*(s, a) \tag{4}$$

Theoretical Analysis

Theorem

$$\lim_{\tau \to 1} V_\tau(s) = \max_{\substack{a \in \mathcal{A} \\ s.t\ \pi_\beta(a|s) > 0}} Q^*(s, a) \tag{4}$$

Proof

Consider a set $\mathcal{T} = \{\tau_i \mid \tau_i \in (0, 1) \forall i\}$. Then for a sequence $\tau_1 < \tau_2 < \tau_3 < \ldots < \tau_n$, from Lemma 2, we have:

$$V_{\tau_1}(s) \leq V_{\tau_2}(s) \leq V_{\tau_3}(s) \leq \ldots \leq V_{\tau_n}(s)$$

Theoretical Analysis

Theorem

$$\lim_{\tau \to 1} V_\tau(s) = \max_{\substack{a \in \mathcal{A} \\ s.t\ \pi_\beta(a|s) > 0}} Q^*(s, a) \tag{4}$$

Proof

Consider a set $\mathcal{T} = \{\tau_i \mid \tau_i \in (0, 1) \forall i\}$. Then for a sequence $\tau_1 < \tau_2 < \tau_3 < \ldots < \tau_n$, from Lemma 2, we have:

$$V_{\tau_1}(s) \leq V_{\tau_2}(s) \leq V_{\tau_3}(s) \leq \ldots \leq V_{\tau_n}(s)$$

Since, by Corollary 2.1 $V_\tau(s)$ is upper bounded by

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

Theoretical Analysis

Theorem

$$\lim_{\tau \to 1} V_\tau(s) = \max_{\substack{a \in \mathcal{A} \\ s.t\ \pi_\beta(a|s) > 0}} Q^*(s, a) \tag{4}$$

Proof

Consider a set $\mathcal{T} = \{\tau_i \mid \tau_i \in (0, 1) \forall i\}$. Then for a sequence $\tau_1 < \tau_2 < \tau_3 < \ldots < \tau_n$, from Lemma 2, we have:

$$V_{\tau_1}(s) \leq V_{\tau_2}(s) \leq V_{\tau_3}(s) \leq \ldots \leq V_{\tau_n}(s)$$

Since, by Corollary 2.1 $V_\tau(s)$ is upper bounded by

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

therefore, $\{V_{\tau_i}(s)\}_{i=1}^n$ is bounded monotonically non-decreasing sequence.

Theoretical Analysis

Theorem

$$\lim_{\tau \to 1} V_\tau(s) = \max_{\substack{a \in \mathcal{A} \\ s.t\ \pi_\beta(a|s) > 0}} Q^*(s,a) \tag{4}$$

Proof

Consider a set $\mathcal{T} = \{\tau_i \mid \tau_i \in (0,1) \forall i\}$. Then for a sequence $\tau_1 < \tau_2 < \tau_3 < \ldots < \tau_n$, from Lemma 2, we have:

$$V_{\tau_1}(s) \leq V_{\tau_2}(s) \leq V_{\tau_3}(s) \leq \ldots \leq V_{\tau_n}(s)$$

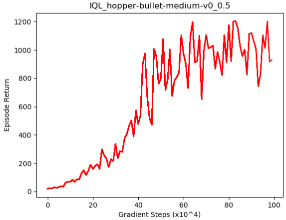
Since, by Corollary 2.1 $V_\tau(s)$ is upper bounded by

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

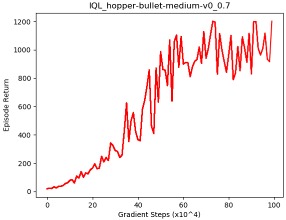
therefore, $\{V_{\tau_i}(s)\}_{i=1}^{n}$ is bounded monotonically non-decreasing sequence. Hence, we have the following limit:

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

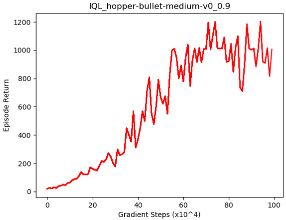
Experiments



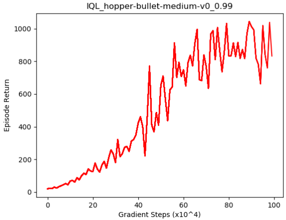
(a) $\tau = 0.5$



(b) $\tau = 0.7$



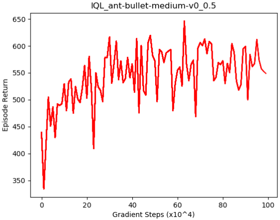
(c) $\tau = 0.9$



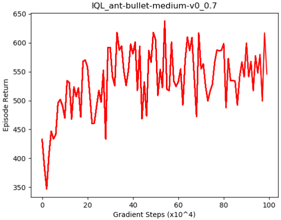
(d) $\tau = 0.99$

Figure: hopper-bullet-medium-v0

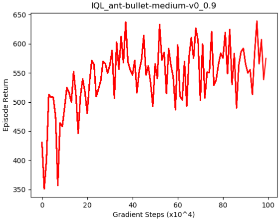
Experiments



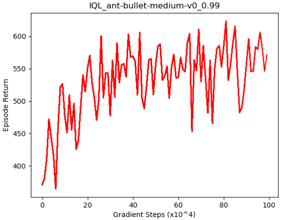
(a) $\tau = 0.5$



(b) $\tau = 0.7$

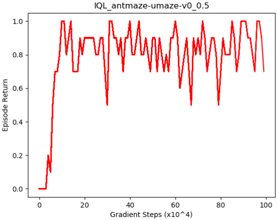


(c) $\tau = 0.9$

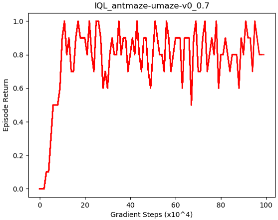


(d) $\tau = 0.99$

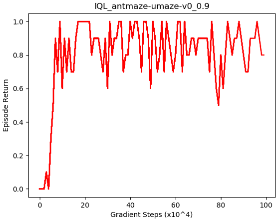
Figure: ant-bullet-medium-v0



(a) $\tau = 0.5$



(b) $\tau = 0.7$



(c) $\tau = 0.9$

Figure: antmaze-umaze-v0

Experiments (Online Fine-Tuning)

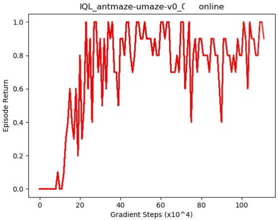


Figure: Online Tuning with $\tau = 0.7$

For fine-tuning experiments, we first run offline RL for 1M gradient steps. Then we continue training while collecting data actively in the environment and adding that data to the replay buffer, running 1 gradient update / environment step. All other training details are kept the same between the offline RL phase and the online RL phase

Drawback

- The theoretical analysis claims that $\tau \to 1$ closely approximates Q^*. However, during the experiments the authors use $\tau = 0.9$ or 0.7.

Drawback

- The theoretical analysis claims that $\tau \to 1$ closely approximates Q^*. However, during the experiments the authors use $\tau = 0.9$ or 0.7.
- Using $\tau \approx 1$ suffers from a serious problem. If the initialization of $V_\psi(s)$ already over-estimates the $\max_a Q(s, a)$, then the loss function will already be zero and parameters will never get updated.

Drawback

- The theoretical analysis claims that $\tau \rightarrow 1$ closely approximates Q^*. However, during the experiments the authors use $\tau = 0.9$ or 0.7.
- Using $\tau \approx 1$ suffers from a serious problem. If the initialization of $V_\psi(s)$ already over-estimates the $\max_a Q(s, a)$, then the loss function will already be zero and parameters will never get updated.

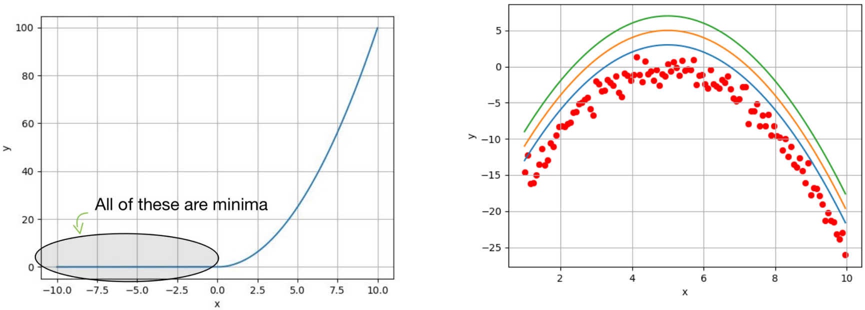


Figure: Drawbacks of Proposed Method

Potential Improvements

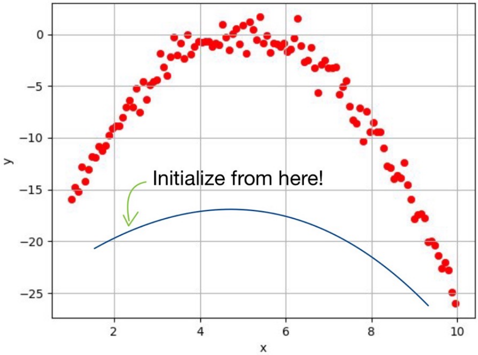
Idea 1

Fit $V_{\hat{\psi}}(s)$ using expectile regression with $\tau \approx 0$, so that it is an under-estimate of Q-values. Now, use $V_{\hat{\psi}}(s)$ as initialization for expectile regression with $\tau \approx 1$.

Potential Improvements

Idea 1

Fit $V_{\hat{\psi}}(s)$ using expectile regression with $\tau \approx 0$, so that it is an under-estimate of Q-values. Now, use $V_{\hat{\psi}}(s)$ as initialization for expectile regression with $\tau \approx 1$.



Potential Improvements

Idea 2

Penalize/Regularize $V_\psi(s)$ based on how far they are from $\max_a Q(s,a)$. To get this, choose some exemplar/candidate points for $\max_a Q(s,a)$ i.e, minimize

$$L'_V(\psi) = \mathbb{E}_{(s,a)\sim\mathcal{D}}\left[L_2^\tau(Q_{\hat{\theta}}(s,a) - V_\psi(s)) + \lambda_R(V_\psi(s) - \hat{Q}(s,a))\right]$$

where, $\hat{Q}(s,a)$ is an exemplar.

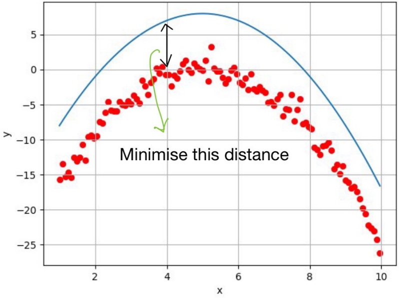
Potential Improvements

Idea 2

Penalize/Regularize $V_\psi(s)$ based on how far they are from $\max_a Q(s,a)$. To get this, choose some exemplar/candidate points for $\max_a Q(s,a)$ i.e, minimize

$$L'_V(\psi) = \mathbb{E}_{(s,a)\sim\mathcal{D}}\left[L_2^\tau(Q_{\hat{\theta}}(s,a) - V_\psi(s)) + \lambda_R(V_\psi(s) - \hat{Q}(s,a))\right]$$

where, $\hat{Q}(s,a)$ is an exemplar.



References I

Kostrikov, I., Nair, A., and Levine, S. (2021).
Offline reinforcement learning with implicit q-learning.
arXiv preprint arXiv:2110.06169.

Levine, S., Kumar, A., Tucker, G., and Fu, J. (2020).
Offline reinforcement learning: Tutorial, review, and perspectives on open problems.
arXiv preprint arXiv:2005.01643.

Link to our experiments: https://github.com/tayalmanan28/Offline˙learning˙IQL