# A Detailed SAQ-IQL derivation

In this section, we describe in detail the derivation for the SAQ-IQL algorithm. We start with the original optimization objective from IQL that employs an explicit policy constraint w.r.t. to a behavior policy

$$\begin{aligned}\pi^* &= \arg\max_{\pi\in\Pi} \mathbb{E}_{a\sim\pi(\cdot|s)}[A^\pi(s,a)] \\ \text{s.t. } & D_{KL}(\pi(\cdot,s)||\pi_\beta(\cdot|s)) \leq \epsilon \\ & \sum_a \pi(a|s) = 1\end{aligned} \tag{11}$$

We can write down the Lagrangian of 11 and then solve for KKT conditions:

$$L(\pi,\lambda,\alpha) = \mathbb{E}_{a\sim\pi(\cdot|s)}[A^\pi(s,a)] + \lambda(\epsilon - D_{kl}(\pi(\cdot|s)||\pi_\beta(\cdot|s))) + \alpha(1 - \sum_a \pi(a|s))$$

Differentiating the Lagrangian term w.r.t. $\pi$ and dividing by the constant $\lambda$ gives

$$\frac{\partial L(\pi,\lambda,\alpha)}{\partial\pi}\cdot\frac{1}{\lambda} = \frac{A^\pi(s,a)}{\lambda} + \log\pi_\beta(\cdot|s) - \log\pi(\cdot|s) + \lambda - \alpha$$

setting it to zero gives the closed-form solution

$$\pi^* = \frac{1}{z(s)}\exp[\frac{A^\pi(s,a)}{\lambda} + \log\pi_\beta]$$

where $z(s)$ is a normalizing term.

# B Ablation Studies for SAQ

| Task | No State | State | No State | State | No State | State |
|---|---|---|---|---|---|---|
| | SAQ-CQL | SAQ-CQL | SAQ-IQL | SAQ-IQL | SAQ-BRAC | SAQ-BRAC |
| halfcheetah-medium-replay-v2 | 1.56 | **47.07** | 1.56 | **36.2** | -1.6 | **40.25** |
| hopper-medium-replay-v2 | 15.24 | **94.73** | 11.74 | **59.43** | 21.56 | **68.87** |
| walker2d-medium-replay-v2 | 4.67 | **81.72** | 6.89 | **45.64** | -0.25 | **53.52** |
| average | 7.16 | **74.51** | 6.73 | **47.09** | 6.57 | **54.21** |

Table 3: Comparing the performance of state-conditioned action discretization against unconditioned action discretization with CQL. The state-conditioned discretization scheme significantly outperforms the unconditioned one since unconditioned action discretization cannot compress the action space into few number of bins.

**Comparing discretization methods.** To understand the importance of the state-conditioned discretization method, we compare it against a naive discretization method where the VQ-VAE discretizes the actions without conditioning on the states and present the results in Table 3. We see that the state-conditioning allows is indeed highly important in compressing the action space into a small number of bins, resulting in much higher performance than a state-agnostic discretization scheme.

**Codebook size robustness.** One key design choice we make in this paper is the use of a VQ-VAE, one natural question is then our method's robustness against the codebook size in the VQ-VAE; since that can be crucial in determining the quality of the performed discretization through it. Towards this end, we empirically experiment with varying codebook sizes across all three algorithms. We present the results in Table 4, and we found that our method's performance is consistent across codebook sizes; which further resonates with the practical utility of adopting our method.

| Codebook Size | 16 | 32 | 64 | 128 |
|---|---|---|---|---|
| SAQ-CQL | 108.7 | 111.6 | 110.8 | 103.2 |
| SAQ-IQL | 106.9 | 104.8 | 104.2 | 94.42 |
| SAQ-BRAC | 106.5 | 108.3 | 105.7 | 107 |

Table 4: Comparing the performance of SAQ-IQL, SAQ-CQL, and SAQ-BRAC on hopper-expert-v2 while varying the codebook size. It can be observed that the discretized algorithms are invariant to codebook size changes.

**Controlling policy constraint levels.** As stated in Sec.4.1, one key premise of SAQ is that we can enforce policy constraint or value conservatism exactly; which associates with the practical performance of offline RL methods. To further verify this hypothesis empirically; we pick one task from the Gym locomotion suite and vary the weight coefficients for policy constraint or value conservatism to observe the resulting performance. We present our results in Fig. 5, we can see that small constraint enforcement leads to poor performance initially; then the performance ramps up when we increase the coefficients; finally converges with sufficient large coefficients. This observation confirms our conjecture in the paper.

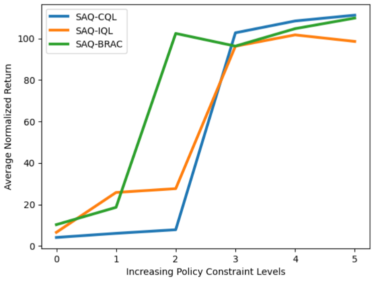

Figure 5: Increasing policy constraint levels on hopper-expert-v2 environment from Gym locomotion.

## C Additional Experiment Results

| Task | BRAC | SAQ-BRAC | IQL | SAQ-IQL | CQL | SAQ-CQL | BC | SAQ-BC |
|---|---|---|---|---|---|---|---|---|
| halfcheetah-expert-v2 | 67.99 | 64 | 94.78 | 90.3 | 43.9 | 92.65 | 7.3 | 108.4 |
| halfcheetah-medium-expert-v2 | 73.07 | 90.31 | 88.06 | 89.05 | 50.32 | 91.69 | 36.43 | 57.65 |
| halfcheetah-medium-replay-v2 | -3.3 | 35.28 | 44.24 | 36.2 | 47.07 | 40.25 | 19.91 | 3.76 |
| halfcheetah-medium-v2 | 47.08 | 43 | 47.3 | 42.52 | 48.56 | 43.89 | 36.39 | 47.02 |
| hopper-expert-v2 | 67.34 | 110 | 108.8 | 100.3 | 100.5 | 109.9 | 28.49 | 92.63 |
| hopper-medium-expert-v2 | 50.74 | 98.75 | 32.85 | 81.56 | 67.08 | 98.47 | 38.09 | 58.94 |
| hopper-medium-replay-v2 | 36.81 | 32.54 | 61.75 | 59.43 | 94.73 | 68.87 | 14.1 | 32.93 |
| hopper-medium-v2 | 57.06 | 54.16 | 54.3 | 50.53 | 70.32 | 38.25 | 47.92 | 42.47 |
| walker2d-expert-v2 | 108.4 | 107.5 | 110.12 | 107.6 | 109.2 | 107.3 | 95.92 | 104.5 |
| walker2d-medium-expert-v2 | 109 | 102.9 | 109.56 | 100.3 | 110.7 | 108.6 | 70.21 | 103.2 |
| walker2d-medium-replay-v2 | 73.72 | 0.4 | 68.71 | 45.64 | 81.72 | 53.52 | 11.36 | 15.27 |
| walker2d-medium-v2 | 81.94 | 64.8 | 81.25 | 68 | 83.11 | 74.77 | 60.18 | 62.52 |
| locomotion average | 64.16 | 66.98 | 75.14 | 76.67 | 75.6 | 77.35 | 38.86 | **60.77** |
| antmaze-medium-diverse-v2 | 26 | 47.6 | 76.67 | 68.33 | 72.75 | 75.47 | 0 | 0 |
| antmaze-medium-play-v2 | 48.67 | 56.93 | 78.67 | 74.33 | 67.04 | 68.67 | 0 | 0 |
| antmaze-large-diverse-v2 | 0.66 | 9.73 | 31.67 | 41 | 35.62 | 36 | 0 | 0 |
| antmaze-large-play-v2 | 0 | 3.73 | 34.33 | 40 | 45.18 | 47.33 | 0 | 0 |
| antmaze average | 18.84 | **29.5** | 55.34 | 55.92 | 55.15 | 56.87 | 0 | 0 |
| door-human-v0 | -1.01 | 35.42 | 1.79 | 9.26 | 0.84 | 2.12 | 3.29 | 9.28 |
| hammer-human-v0 | -1.42 | 20.52 | 1.41 | 1.57 | 0.27 | 0.6 | 0.8 | 1.38 |
| pen-human-v0 | 98.15 | 98.41 | 69.69 | 80.25 | 41.24 | 82.73 | 45.28 | 73.3 |
| relocate-human-v0 | -0.28 | 6 | 8.38 | 0.2 | -0.05 | 0.02 | 0.04 | 0.02 |
| adroit average | 23.86 | **40.09** | 20.32 | **22.82** | 10.58 | **21.37** | 12.35 | **21** |
| kitchen-mixed-v0 | 10.33 | 53.33 | 48.92 | 52.92 | 62 | 57.67 | 11.67 | 34 |
| kitchen-complete-v0 | 31.67 | 10 | 66 | 76.76 | 14 | 47.67 | 16.67 | 90.33 |
| kitchen-partial-v0 | 5.33 | 45 | 37.58 | 46.25 | 70 | 92.33 | 23.33 | 46.67 |
| kitchen average | 15.78 | **36.11** | 50.83 | **58.61** | 48.67 | **65.89** | 17.22 | **57** |

Table 5: Full table of averaged normalized scores on locomotion, Adroit, AntMaze, and kitchen domains from D4RL.