References

[1] Anonymous. LS-IQ: Implicit reward regularization for inverse reinforcement learning. In *Submitted to The Eleventh International Conference on Learning Representations*, 2023. under review.

[2] Dimitri P Bertsekas and John N Tsitsiklis. Neuro-dynamic programming: an overview. In *Proceedings of 1995 34th IEEE conference on decision and control*, volume 1, pages 560–564. IEEE, 1995.

[3] Stephen Boyd, Stephen P Boyd, and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[4] Bo Dai, Albert Shaw, Niao He, Lihong Li, and Le Song. Boosting the actor with dual critic. *arXiv preprint arXiv:1712.10282*, 2017.

[5] Eric V Denardo. On linear programming in a markov decision problem. *Management Science*, 16(5):281–288, 1970.

[6] Ben Eysenbach, Sergey Levine, and Russ R Salakhutdinov. Replacing rewards with examples: Example-based policy search via recursive classification. *Advances in Neural Information Processing Systems*, 34:11541–11552, 2021.

[7] Pete Florence, Corey Lynch, Andy Zeng, Oscar A Ramirez, Ayzaan Wahid, Laura Downs, Adrian Wong, Johnny Lee, Igor Mordatch, and Jonathan Tompson. Implicit behavioral cloning. In *Conference on Robot Learning*, pages 158–168. PMLR, 2022.

[8] 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.

[9] Justin Fu, Aviral Kumar, Matthew Soh, and Sergey Levine. Diagnosing bottlenecks in deep q-learning algorithms. In *International Conference on Machine Learning*, pages 2021–2030. PMLR, 2019.

[10] Justin Fu, Katie Luo, and Sergey Levine. Learning robust rewards with adversarial inverse reinforcement learning. *arXiv preprint arXiv:1710.11248*, 2017.

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

[12] Scott Fujimoto, Herke Hoof, and David Meger. Addressing function approximation error in actor-critic methods. In *International conference on machine learning*, pages 1587–1596. PMLR, 2018.

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

[14] Divyansh Garg, Shuvam Chakraborty, Chris Cundy, Jiaming Song, and Stefano Ermon. Iq-learn: Inverse soft-q learning for imitation. *Advances in Neural Information Processing Systems*, 34:4028–4039, 2021.

[15] Divyansh Garg, Joey Hejna, Matthieu Geist, and Stefano Ermon. Extreme q-learning: Maxent rl without entropy. *arXiv preprint arXiv:2301.02328*, 2023.

[16] Seyed Kamyar Seyed Ghasemipour, Richard Zemel, and Shixiang Gu. A divergence minimization perspective on imitation learning methods. In *Conference on Robot Learning*, pages 1259–1277. PMLR, 2020.

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

[18] Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse domains through world models. *arXiv preprint arXiv:2301.04104*, 2023.

[19] Nicklas Hansen, Xiaolong Wang, and Hao Su. Temporal difference learning for model predictive control. *arXiv preprint arXiv:2203.04955*, 2022.

[20] Hado Hasselt. Double q-learning. *Advances in neural information processing systems*, 23, 2010.

[21] Jonathan Ho and Stefano Ermon. Generative adversarial imitation learning. *Advances in neural information processing systems*, 29:4565–4573, 2016.

[22] Hana Hoshino, Kei Ota, Asako Kanezaki, and Rio Yokota. Opirl: Sample efficient off-policy inverse reinforcement learning via distribution matching. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 448–454. IEEE, 2022.

[23] Ehsan Imani, Eric Graves, and Martha White. An off-policy policy gradient theorem using emphatic weightings. *Advances in Neural Information Processing Systems*, 31, 2018.

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

[25] Ray Jiang, Tom Zahavy, Zhongwen Xu, Adam White, Matteo Hessel, Charles Blundell, and Hado Van Hasselt. Emphatic algorithms for deep reinforcement learning. In *International Conference on Machine Learning*, pages 5023–5033. PMLR, 2021.

[26] Sham M Kakade. A natural policy gradient. *Advances in neural information processing systems*, 14, 2001.

[27] Liyiming Ke, Sanjiban Choudhury, Matt Barnes, Wen Sun, Gilwoo Lee, and Siddhartha Srinivasa. Imitation learning as f-divergence minimization. In *Algorithmic Foundations of Robotics XIV: Proceedings of the Fourteenth Workshop on the Algorithmic Foundations of Robotics 14*, pages 313–329. Springer International Publishing, 2021.

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

[29] Geon-Hyeong Kim, Seokin Seo, Jongmin Lee, Wonseok Jeon, HyeongJoo Hwang, Hongseok Yang, and Kee-Eung Kim. Demodice: Offline imitation learning with supplementary imperfect demonstrations. In *International Conference on Learning Representations*, 2022.

[30] Ilya Kostrikov, Kumar Krishna Agrawal, Debidatta Dwibedi, Sergey Levine, and Jonathan Tompson. Discriminator-actor-critic: Addressing sample inefficiency and reward bias in adversarial imitation learning. *arXiv preprint arXiv:1809.02925*, 2018.

[31] Ilya Kostrikov, Ofir Nachum, and Jonathan Tompson. Imitation learning via off-policy distribution matching. *arXiv preprint arXiv:1912.05032*, 2019.

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

[33] Aviral Kumar, Justin Fu, Matthew Soh, George Tucker, and Sergey Levine. Stabilizing off-policy q-learning via bootstrapping error reduction. *Advances in Neural Information Processing Systems*, 32, 2019.

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

[35] Jongmin Lee, Wonseok Jeon, Byungjun Lee, Joelle Pineau, and Kee-Eung Kim. Optidice: Offline policy optimization via stationary distribution correction estimation. In *International Conference on Machine Learning*, pages 6120–6130. PMLR, 2021.

[36] Timothy P Lillicrap, Jonathan J Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*, 2015.

[37] Yecheng Jason Ma, Andrew Shen, Dinesh Jayaraman, and Osbert Bastani. Smodice: Versatile offline imitation learning via state occupancy matching. *arXiv preprint arXiv:2202.02433*, 2022.

[38] Alan S Manne. Linear programming and sequential decisions. *Management Science*, 6(3):259–267, 1960.

[39] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, 2013.

[40] Ofir Nachum and Bo Dai. Reinforcement learning via fenchel-rockafellar duality. *arXiv preprint arXiv:2001.01866*, 2020.

[41] Ofir Nachum, Bo Dai, Ilya Kostrikov, Yinlam Chow, Lihong Li, and Dale Schuurmans. Algaedice: Policy gradient from arbitrary experience. *arXiv preprint arXiv:1912.02074*, 2019.

[42] Ashvin Nair, Abhishek Gupta, Murtaza Dalal, and Sergey Levine. Awac: Accelerating online reinforcement learning with offline datasets. *arXiv preprint arXiv:2006.09359*, 2020.

[43] Tianwei Ni, Harshit Sikchi, Yufei Wang, Tejus Gupta, Lisa Lee, and Ben Eysenbach. f-irl: Inverse reinforcement learning via state marginal matching. In *Conference on Robot Learning*, pages 529–551. PMLR, 2021.

[44] Julian Schrittwieser, Ioannis Antonoglou, Thomas Hubert, Karen Simonyan, Laurent Sifre, Simon Schmitt, Arthur Guez, Edward Lockhart, Demis Hassabis, Thore Graepel, et al. Mastering atari, go, chess and shogi by planning with a learned model. *Nature*, 588(7839):604–609, 2020.

[45] John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. Trust region policy optimization. In *International conference on machine learning*, pages 1889–1897. PMLR, 2015.

[46] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[47] Harshit Sikchi, Akanksha Saran, Wonjoon Goo, and Scott Niekum. A ranking game for imitation learning. *arXiv preprint arXiv:2202.03481*, 2022.

[48] Harshit Sikchi, Wenxuan Zhou, and David Held. Learning off-policy with online planning. In *Conference on Robot Learning*, pages 1622–1633. PMLR, 2022.

[49] Anikait Singh, Aviral Kumar, Quan Vuong, Yevgen Chebotar, and Sergey Levine. Offline rl with realistic datasets: Heteroskedasticity and support constraints. *arXiv preprint arXiv:2211.01052*, 2022.

[50] Gokul Swamy, Sanjiban Choudhury, J Andrew Bagnell, and Steven Wu. Of moments and matching: A game-theoretic framework for closing the imitation gap. In *International Conference on Machine Learning*, pages 10022–10032. PMLR, 2021.

[51] Sebastian Thrun and Anton Schwartz. Issues in using function approximation for reinforcement learning. In *Proceedings of the Fourth Connectionist Models Summer School*, volume 255, page 263. Hillsdale, NJ, 1993.

[52] Ikechukwu Uchendu, Ted Xiao, Yao Lu, Banghua Zhu, Mengyuan Yan, Joséphine Simon, Matthew Bennice, Chuyuan Fu, Cong Ma, Jiantao Jiao, et al. Jump-start reinforcement learning. *arXiv preprint arXiv:2204.02372*, 2022.

[53] Yifan Wu, George Tucker, and Ofir Nachum. Behavior regularized offline reinforcement learning. *arXiv preprint arXiv:1911.11361*, 2019.

[54] Tengyang Xie, Nan Jiang, Huan Wang, Caiming Xiong, and Yu Bai. Policy finetuning: Bridging sample-efficient offline and online reinforcement learning. *Advances in neural information processing systems*, 34:27395–27407, 2021.

[55] Tianhe Yu, Garrett Thomas, Lantao Yu, Stefano Ermon, James Y Zou, Sergey Levine, Chelsea Finn, and Tengyu Ma. Mopo: Model-based offline policy optimization. *Advances in Neural Information Processing Systems*, 33:14129–14142, 2020.

[56] Ruiyi Zhang, Bo Dai, Lihong Li, and Dale Schuurmans. Gendice: Generalized offline estimation of stationary values. *arXiv preprint arXiv:2002.09072*, 2020.

[57] Zhuangdi Zhu, Kaixiang Lin, Bo Dai, and Jiayu Zhou. Off-policy imitation learning from observations. *Advances in Neural Information Processing Systems*, 33:12402–12413, 2020.

[58] Brian D Ziebart, Andrew L Maas, J Andrew Bagnell, Anind K Dey, et al. Maximum entropy inverse reinforcement learning. In *Aaai*, volume 8, pages 1433–1438. Chicago, IL, USA, 2008.

[59] Konrad Zolna, Alexander Novikov, Ksenia Konyushkova, Caglar Gulcehre, Ziyu Wang, Yusuf Aytar, Misha Denil, Nando de Freitas, and Scott Reed. Offline learning from demonstrations and unlabeled experience. *arXiv preprint arXiv:2011.13885*, 2020.

A Appendix

A.1 A Review of Dual-RL

In this section, we aim to give a self-contained review for Dual Reinforcement Learning. For a more thorough read, refer to [40].

A.1.1 Convex conjugates and f-divergence

We first review the basics of duality in reinforcement learning. Let $f : (0, \infty) \to \mathbb{R}$ be a convex function. The convex conjugate f^* of f is defined by:

$$f^*(y) = \sup_{x \in \mathbb{R}} [\langle x, y \rangle - f(x)] \tag{16}$$

where $\langle\ \rangle$ denotes the dot product. The convex conjugates have the important property that f^* is also convex and the convex conjugate of f^* retrieves back the original function f. Going forward, we would be dealing extensively with f-divergences. Informally, f-divergences [] are a measure of distance between two probability distributions. Here's a more formal definition:

Let $f : (0, \infty) \to \mathbb{R}$ be a convex lower semi-continuous function with $f(1) = 0$. Let P and Q be two probability distributions, then the f-divergence is defined as:

$$D_{\mathrm{f}}(P \,||\, Q) = \mathbb{E}_{z \sim Q} \left[f \left(\frac{P(z)}{Q(z)} \right) \right] \tag{17}$$

Now, we will do a simple exercise in finding the convex conjugate for this f-divergence (where f is a convex function) which will also give us the well-known variational representation of f-divergence. We will use it frequently in the subsequent sections.

Using the definition of convex conjugate and the fact that convex conjugate of f^* gives back f, we have:

$$\begin{aligned} D_{\mathrm{f}}(P \,||\, Q) &= \mathbb{E}_{z \sim Q} \left[f \left(\frac{P(z)}{Q(z)} \right) \right] && (18) \\ &= \sup_{y} \mathbb{E}_{z \sim Q} \left[\frac{P(z)}{Q(z)} y(z) \right] - \mathbb{E}_{Q}[f^*(y(z))] && (19) \\ &= \sup_{y : \mathcal{Z} \to \mathbb{R}} \mathbb{E}_{z \sim P}[y(z)] - \mathbb{E}_{z \sim Q}[f^*(y(z))] && (20) \end{aligned}$$

Thus Eq 20 derives the variational form for f-divergence. Although deriving the analytical form of f^* is not complicated for most common f-divergences — set the derivative for Eq 16 to zero and find out the stationary point, it might be useful to list some common f-divergences and their conjugates f^*. We also note an important relation regarding f and f^*: $(f^*)' = (f')^{-1}$, where the $'$ notation denotes first derivative.

Table 2: List of f-divergences and their convex conjugates

Divergence	$f(t)$	$f^*(u)$
Forward KL	$-\log t$	$-1 - \log(-u)$
Reverse KL	$t \log t$	$e^{(u-1)}$
Squared Hellinger	$(\sqrt{t} - 1)^2$	$\frac{u}{1-u}$
Pearson χ^2	$(t-1)^2$	$u + \frac{u^2}{4}$
Total variation	$\frac{1}{2}\lvert t - 1 \rvert$	u
Jensen-Shannon	$-(t+1)\log(\frac{t+1}{2}) + t \log t$	$-\log(2 - e^u)$

A.1.2 Duality in Reinforcement Learning

Duality in reinforcement learning allows a different perspective for solving RL problems, often giving off-policy alternatives to typical on-policy approaches. We consider a regularized policy optimization objective below:

$$\max_{\pi} \mathbb{E}_{d^\pi(s,a)}[r(s,a)] - \alpha D_{\mathrm{f}}(d^\pi(s,a) \mid\mid d^o(s,a)) \tag{21}$$

where d^o is a known state-action visitation distribution. Optimizing over π, at first sight, gives us a non-convex problem further complicating the analysis. We can rewrite the problem as a linear program (LP) by considering optimization over *valid* state-action visitations by adding a constraint for the optimization:

$$\max_{\pi, d \geq 0} \mathbb{E}_{d(s,a)}[r(s,a)] - \alpha D_{\mathrm{f}}(d(s,a) \mid\mid d^O(s,a)) \tag{22}$$

$$\text{s.t } d(s,a) = (1-\gamma) d_0(s).\pi(a|s) + \gamma \sum_{s',a'} d(s',a') p(s|s',a') \pi(a|s) \tag{23}$$

where α allows us to weigh policy improvement against conservatism from staying close to the state-action distribution d^O.

A careful reader may notice that the above problem is overconstrained. The solution to the inner maximization with respect to d is uniquely determined by the $|\mathcal{S}| \times |\mathcal{A}|$ constraints from the formulation. The inner optimization, using only the constraints, uniquely determines the visitation d^π - the state action visitation of policy π and is independent of the term being optimized 22. The gradient with respect to policy π when d is optimized can be shown to be equivalent to the on-policy policy gradient (see Section 5.1 from [40]).

The constraints above are the probability flow equations that a stationary state-action distribution must satisfy. Now, how can we go about solving it? Here is where duality comes into play. First, we form the lagrangian dual of our original optimization problem, transforming our constrained optimization into an unconstrained form. This introduces additional optimization variables - the Lagrange multipliers.

$$\begin{aligned}
&\max_{\pi, d \geq 0} \min_{Q(s,a)} \mathbb{E}_{s,a \sim d(s,a)}[r(s,a)] - \alpha D_{\mathrm{f}}(d(s,a) \mid\mid d^o(s,a)) \\
&+ \sum_{s,a} Q(s,a) \left((1-\gamma) d_0(s).\pi(a|s) + \gamma \sum_{s',a'} d(s',a') p(s|s',a') \pi(a|s) - d(s,a) \right)
\end{aligned}$$

where $Q(s,a)$ are the Lagrange multipliers for enforcing the equality constraints. We can now do some algebraic manipulation on the above equation to further simplify it:

$$\begin{aligned}
&\max_{\pi, d \geq 0} \min_{Q(s,a)} \mathbb{E}_{s,a \sim d(s,a)}[r(s,a)] - \alpha D_{\mathrm{f}}(d(s,a) \mid\mid d^O(s,a)) \\
&+ \sum_{s,a} Q(s,a) \left((1-\gamma) d_0(s).\pi(a|s) + \gamma \sum_{s',a'} d(s',a') p(s|s',a') \pi(a|s) - d(s,a) \right)
\end{aligned} \tag{24}$$

$$= \max_{\pi, d \geq 0} \min_{Q(s,a)} (1-\gamma)\mathbb{E}_{d_0(s),\pi(a|s)}[Q(s,a)]$$
$$+ \mathbb{E}_{s,a\sim d}\left[r(s,a) + \gamma \sum_{s'} p(s'|s,a)\pi(a'|s')Q(s',a') - Q(s,a)\right] - \alpha D_{\mathrm{f}}(d(s,a) \mid\mid d^O(s,a)) \tag{25}$$

$$= \max_{\pi(a|s)} \min_{Q(s,a)} \max_{d(s,a)\geq 0} (1-\gamma)\mathbb{E}_{d_0(s),\pi(a|s)}[Q(s,a)]$$
$$+ \mathbb{E}_{s,a\sim d}\left[r(s,a) + \gamma \sum_{s'} p(s'|s,a)\pi(a'|s')Q(s',a') - Q(s,a)\right] - \alpha D_{\mathrm{f}}(d(s,a) \mid\mid d^O(s,a)) \tag{26}$$

$$= \max_{\pi(a|s)} \min_{Q(s,a)} \max_{d(s,a)\geq 0} \frac{(1-\gamma)}{\alpha}\mathbb{E}_{d_0(s),\pi(a|s)}[Q(s,a)]$$
$$+ \mathbb{E}_{s,a\sim d}\left[(r(s,a) + \gamma \sum_{s'} p(s'|s,a)\pi(a'|s')Q(s',a') - Q(s,a))/\alpha\right] - D_{\mathrm{f}}(d(s,a) \mid\mid d^O(s,a)) \tag{27}$$

$$= \max_{\pi(a|s)} \min_{Q(s,a)} \frac{(1-\gamma)}{\alpha}\mathbb{E}_{d_0(s),\pi(a|s)}[Q(s,a)]$$
$$+ \mathbb{E}_{s,a\sim d^O}\left[f^*((r(s,a) + \gamma \sum_{s'} p(s'|s,a)\pi(a'|s')Q(s',a') - Q(s,a))/\alpha)\right] \tag{28}$$

The last step is due to the application of Eq 16 (convex conjugate definition). To see this more clearly let $y(s,a) = r(s,a) + \gamma \sum_{s'} p(s'|s,a)\pi(a'|s')Q(s',a') - Q(s,a)$. Then,

$$\max_{d\geq 0} \mathbb{E}_{s,a\sim d}[y(s,a)] - D_{\mathrm{f}}(d(s,a) \mid\mid d^o(s,a)) \tag{29}$$
$$= \max_{d\geq 0} \mathbb{E}_{s,a\sim d^o}\left[\frac{d(s,a)}{d^O(s,a)}y(s,a) - f\left(\frac{d(s,a)}{d^o(s,a)}\right)\right] \tag{30}$$
$$= \mathbb{E}_{d^o}[f^*(y(s,a))] \tag{31}$$

Finally, the policy optimization problem is reduced to the solving the following min-max optimization, which we will refer to as `dual-Q`:

$$\max_{\pi(a|s)} \min_{Q(s,a)} \frac{(1-\gamma)}{\alpha}\mathbb{E}_{d_0(s),\pi(a|s)}[Q(s,a)] + \mathbb{E}_{s,a\sim d^O}\left[f^*((r(s,a) + \gamma \sum_{s'} p(s'|s,a)\pi(a|s')Q(s',a') - Q(s,a))/\alpha)\right] \tag{32}$$

For common f-divergences, table 2 lists the corresponding convex conjugates f^*. Also, note that the primal RL problem is convex and due to slater's condition [3] we can interchange the min-max between the Lagrange variable Q and visitation distribution d to max-min.

In the case of deterministic policy and deterministic dynamics, the above-obtained optimization takes a simpler form:

$$\min_{Q(s,a)} \max_{\pi(a|s)} \frac{(1-\gamma)}{\alpha}\mathbb{E}_{\rho_0(s)}[Q(s,\pi(s))] + \mathbb{E}_{s,a\sim d^o}[f^*((r(s,a) + \gamma Q(s',\pi(s')) - Q(s,a))/\alpha)] \tag{33}$$

Now, we have seen how we can transform a regularized RL problem into its `dual-Q` form which uses Lagrange variables in the form of state-action functions. Interestingly, we can go further to transform the regularized RL problem into Lagrange variables (V) that only depend on the state, and in doing so we also get rid of the min-max optimization in the `dual-Q`.

Consider the regularized RL problem again (Eq 21). This time, we formulate the visitation constraints to depend solely on states rather than state-action pairs. We consider $\alpha = 1$ for sake of exposition. Interested readers can derive the result for $\alpha \neq 1$ as in the `dual-Q` case above. Hence, we are solving the following constrained optimization problem:

$$\max_{d \geq 0} \mathbb{E}_{d(s,a)}[r(s,a)] - D_{\mathrm{f}}(d(s,a) \mid\mid d^O(s,a)) \tag{34}$$

$$\text{s.t} \sum_{a \in \mathcal{A}} d(s,a) = (1-\gamma) d_0(s) + \gamma \sum_{s',a'} d(s',a') p(s|s',a') \tag{35}$$

As before we construct the Lagrangian dual to this problem. Note that our constraints now solely depend on s.

$$\max_{d \geq 0} \min_{V(s)} \mathbb{E}_{s \sim d(s,a)}[r(s,a)] - D_{\mathrm{f}}(d(s,a) \mid\mid d^o(s,a)) \tag{36}$$

$$+ \sum_s V(s) \left((1-\gamma) d_0(s) + \gamma \sum_{s',a'} d(s',a') p(s|s',a') - d(s,a) \right) \tag{37}$$

Using similar algebraic manipulations we used to obtain `dual-Q`, we get the `dual-V` formulation for policy optimization:

$$\begin{aligned}
&\max_{d(s,a) \geq 0} \min_{V(s)} \mathbb{E}_{s,a \sim d(s,a)}[r(s,a)] - D_{\mathrm{f}}(d(s,a) \mid\mid d^O(s,a)) \\
&\quad + \sum_s V(s) \left((1-\gamma) d_0(s) + \gamma \sum_{s',a'} d(s',a') p(s|s',a') - d(s,a) \right) \qquad (38) \\
&= \min_{V(s)} \max_{d(s,a) \geq 0} (1-\gamma) \mathbb{E}_{d_0(s)}[V(s)] \\
&\quad + \mathbb{E}_{s,a \sim d} \left[r(s,a) + \gamma \sum_{s'} p(s'|s,a) V(s') - V(s) \right] - D_{\mathrm{f}}(d(s,a) \mid\mid d^O(s,a)) \qquad (39)
\end{aligned}$$

$$= \min_{V(s)} \max_{d(s,a) \geq 0} (1-\gamma) \mathbb{E}_{d_0(s)}[V(s)] \tag{40}$$

$$+ \mathbb{E}_{s,a \sim d} \left[r(s,a) + \gamma \sum_{s'} p(s'|s,a) V(s') - V(s) \right] - D_{\mathrm{f}}(d(s,a) \mid\mid d^O(s,a)) \tag{41}$$

$$= \min_{V(s)} (1-\gamma) \mathbb{E}_{d_0(s)}[V(s)] + \mathbb{E}_{s,a \sim d^O} \left[f^*(r(s,a) + \gamma \sum_{s'} p(s'|s,a) V(s') - V(s)) \right] \tag{42}$$

In summary, we have two methods for policy optimization given by:

`dual-Q`: $\max_\pi \min_Q (1-\gamma) \mathbb{E}_{d_0(s),\pi(a|s)}[Q(s,a)]$
$+ \mathbb{E}_{s,a \sim d^o}[f^*(r(s,a) + \gamma \sum_{s'} p(s'|s,a) \pi(a'|s') Q(s',a') - Q(s,a))]$

and,

`dual-V`: $\min_{V(s)} (1-\gamma) \mathbb{E}_{d_0(s)}[V(s)] + \mathbb{E}_{s,a \sim d^o}[f^*(r(s,a) + \gamma \sum_{s'} p(s'|s,a) V(s') - V(s))]$

The above derivations for dual of RL CoP - `dual-Q` and `dual-V` brings out some important observations

- `dual-Q` and `dual-V` present off-policy policy optimization solutions for regularized RL problems which requires sampling transitions only from the distribution the policy state-action visitation is being regularized against.
- The above property allows us to solve not only RL problems but also imitation problems by setting the reward function to be zero everywhere and d^o to be the expert dataset, and also offline RL problems where we want to maximize reward with the constraint that our state-action visitation should not deviate too much from the replay buffer ($d^o =$ replay-buffer).
- `dual-V` formulation presents a way to solve the RL problem using a single optimization rather than a min-max optimization of the Q-CoP or standard RL formulation. V-CoP implicitly subsumes greedy policy maximization.

A.1.3 Recovering the optimal policy in V-CoP

In the above derivations for dual-Q and dual-V we leveraged the fact that the closed form solution for optimizing d is known and it could be written in the form of Eq 29. The value of d^* can found using KKT conditions on Eq 29:

$$\frac{d^*(s,a)}{d^o(s,a)} = \max\left(0, (f')^{-1}\left(\frac{y(s,a)}{\alpha}\right)\right) \tag{43}$$

Using this ratio there are two ways to recover the optimal policy:

Method 1: Maximum likelihood on expert visitation distribution

Policy learning can be treated as maximizing the likelihood of optimal actions:

$$\max \mathbb{E}_{s,a\sim d^*}[\pi_\theta(a|s)] \tag{44}$$

Using importance sampling we can rewrite the optimization above in a more tractable form:

$$\max_\theta \mathbb{E}_{s,a\sim d^o}[w^*(s,a)\pi_\theta(a|s)] \tag{45}$$

This way of policy learning is similar to weighted behavior cloning, but suffers from the issue that policy is not optimized at state-actions where the expert does not visit, i.e $w^*(s,a) = 0$

Method 2: Reverse KL matching on offline data distribution (Information Projection)

To allow the policy to be optimized at all that states in the offline dataset we consider an alternate objective:

$$\min_\theta D_{\mathrm{KL}}(d^o(s)\pi_\theta(a|s) \;||\; d^o(s)\pi^*(a|s)) \tag{46}$$

The objective can be written in a form suitable for optimization as follows:

$$\min_\theta D_{\mathrm{KL}}(d^o(s)\pi_\theta(a|s) \;||\; d^o(s)\pi^*(a|s)) = \min_\theta \mathbb{E}_{s\sim d^o(s),a\sim\pi_\theta}\left[\log \frac{\pi_\theta(a|s)}{\pi^*(a|s)}\right] \tag{47}$$

$$= \min_\theta \mathbb{E}_{s\sim d^o(s),a\sim\pi_\theta}\left[\log \frac{\pi_\theta(a|s)d^*(s)d^o(s)\pi^o(a|s)}{\pi^*(a|s)d^*(s)d^o(s)\pi^o(a|s)}\right] \tag{48}$$

$$= \min_\theta \mathbb{E}_{s\sim d^o(s),a\sim\pi_\theta}\left[\log \frac{\pi_\theta(a|s)}{\pi^o(a|s)} - \log(w^*(s,a)) + \log\frac{d^*(s)}{d^o(s)}\right] \tag{49}$$

$$= \min_\theta \mathbb{E}_{s\sim d^o(s),a\sim\pi_\theta}[\log(\pi_\theta(a|s)) - \log(\pi^o(a|s)) - \log(w^*(s,a))] \tag{50}$$

This method recovers the optimal policy at the states present in the dataset but requires learning another policy $\pi^o(a|s)$ which can be obtained by behavior cloning the replay buffer.

A.2 Positivity constraints in Dual RL

We have ignored an important consideration in the derivation of dual-RL methods in Section A.1.2 – the constraint that the distribution d we are optimizing for in Q-CoP and V-CoP must be positive. Although this does not affect derivation for `dual-Q` as it is overconstrained and the distribution is guaranteed to be unique, it is imperative we consider this constraint in the `dual-V` setting. We will now modify the derivation for `dual-V` to incorporate these constraints.

$$\max_{d \geq 0} \mathbb{E}_{d(s,a)}[r(s,a)] - D_{\mathrm{f}}(d(s,a) \mid\mid d^O(s,a)) \tag{51}$$

$$\text{s.t} \sum_{a \in \mathcal{A}} d(s,a) = (1-\gamma) d_0(s) + \gamma \sum_{s',a'} d(s',a') p(s|s',a') \tag{52}$$

We arrive at the following equation using the steps in Section A.1.2 (see Equation 39).

$$\begin{aligned}
&= \min_{V(s)} \max_{d(s,a) \geq 0} (1-\gamma) \mathbb{E}_{d_0(s)}[V(s)] \\
&\quad + \mathbb{E}_{s,a \sim d} \left[r(s,a) + \gamma \sum_{s'} p(s'|s,a) V(s') - V(s) \right] - D_{\mathrm{f}}(d(s,a) \mid\mid d^O(s,a)) \qquad (53) \\
&= \min_{V(s)} \max_{d(s,a) \geq 0} (1-\gamma) \mathbb{E}_{d_0(s)}[V(s)] \\
&\quad + \mathbb{E}_{s,a \sim d^O} \left[\frac{d(s,a)}{d^O(s,a)} (r(s,a) + \gamma \sum_{s'} p(s'|s,a) V(s') - V(s)) \right] - \mathbb{E}_{s,a \sim d^O} \left[f(\frac{d(s,a)}{d^O(s,a)}) \right] \qquad (54)
\end{aligned}$$

Let $w(s,a) = \frac{d(s,a)}{d^O(s,a)}$ and $r(s,a) + \gamma \sum_{s'} p(s'|s,a) V(s') - V(s)$ be denoted by $y(s,a)$. We have,

$$\min_{V(s)} \max_{d(s,a) \geq 0} (1-\gamma) \mathbb{E}_{d_0(s)}[V(s)] + \mathbb{E}_{s,a \sim d^O}[w(s,a)(y(s,a))] - \mathbb{E}_{s,a \sim d^O}[f(w(s,a))] \tag{55}$$

We now direct the attention to the inner maximization and find a closed-form solution under the constraint that $d(s,a) \geq 0$.

$$\max_{d(s,a)} \max_{\lambda \geq = 0} \mathbb{E}_{s,a \sim d^O}[w(s,a)(y(s,a))] - \mathbb{E}_{s,a \sim d^O}[f(w(s,a))] + \sum_{s,a} \lambda(s,a) w(s,a) \tag{56}$$

where λ is the Lagrangian dual parameter than ensures the positivity constraint. Since strong duality holds, we can use the KKT constraints to find the solutions $w^*(s,a)$ and $\lambda^*(s,a)$.

Primal feasibility: $w^* \geq 0 \ \ \forall \, s, a$
Dual feasibility: $\lambda^* \geq 0 \ \ \forall \, s, a$
Stationarity: $d^O(s,a)(f'(w^*(s,a)) + y(s,a) + \lambda^*(s,a)) = 0 \ \ \forall \, s, a$
Complementary Slackness: $w^*(s,a)\lambda^*(s,a) = 0 \ \ \forall \, s, a$

Using stationarity we have the following:

$$f'(w^*(s,a)) = y(s,a) + \lambda^*(s,a) \ \ \forall \, s, a \tag{57}$$

Now using complementary slackness only two cases are possible $w^*(s,a) > 0$ or $\lambda^*(s,a) > 0$. Combining both cases we arrive at the following solution for this constrained optimization:

$$w^*(s,a) = \max \left(0, {f'}^{-1}(y(s,a)) \right) \tag{58}$$

We refer to the resulting function after plugging the solution for w^* back as f_p^*.

$$f_p^*(s,a) = w^*(s,a)(y(s,a)) - f(w^*(s,a)) \tag{59}$$

Note that we get the original conjugate f^* back if we do not consider the positivity constraints. i.e

$$f^*(s,a) = {f'}^{-1}(y(s,a))(y(s,a)) - f({f'}^{-1}(y(s,a))) \tag{60}$$

Finally, we have the following optimization to solve for `dual-V` when considering the positivity constraints:

`dual-V` (with positivity constraints): $\min_{V(s)}(1-\gamma)\mathbb{E}_{d_0(s)}[V(s)] + \mathbb{E}_{s,a\sim d^O}\left[f_p^*(y(s,a))\right]$

A.3 Dual Connections to Reinforcement Learning

Our first result shows that Conservative Q-learning [34], an offline RL method primarily understood to prevent overestimation by learning a lower bounded Q function is actually a `dual-Q` method. The lemma below formalizes the above statement:

Lemma 5. *Conservative Q-Learning (CQL) is the dual of Q-CoP with the generator function $f = (t-1)^2$ (Pearson χ^2) and when the regularization distribution is the replay buffer ($d^O = d^R$).*

In other words, CQL eventually solves the regularized RL problem (Q-CoP) in its dual form where the regularization is a particular form of f-divergence. This unification indicates that its better performance compared to the family of behavior-regularized offline RL methods [42, 12, 53], which solve the Q-CoP using approximate dynamic programming is likely due to the choice of f-divergence and more amenable optimization afforded by the dual formulation. The dual-Q formulation has been previously studied for online RL by the name AlgaeDICE [41] but not evaluated in the context of offline RL. Lemma 5 also suggests that CQL is a special case of AlgaeDICE.

Leveraging the dual form of **V-CoP** converts the policy improvement problem from a min-max two-player game to a single optimization, thus potentially making the optimization easier to solve [40]. We also note that an additional step needs to be performed to recover policies in `dual-V` which requires solving a supervised learning problem (see Appendix A.1.3). We first show that Extreme Q-Learning (X-QL) [15], a method for both online and offline RL based on the principle of *implicit maximization* in the value function space using Gumbel regression, can be reduced to a `dual-V` problem with a semi-gradient update rule (i.e `stop-gradient` $(r(s,a) + \gamma \sum_{s'} p(s'|s,a)V(s'))$) when f is set to be the reverse-KL regularization. Here, *implicit maximization* refers to finding the extreme values of a distribution using only samples from the distribution. This insight obtained through duality, allows us to propose a class of algorithms extending X-QL, by choosing different functions f which we show below to result in a family of implicit maximizers.

Lemma 6. *Extreme Q-Learning (X-QL) is the dual of V-CoP with f-divergence set to be the reverse Kullback-Liebler divergence with a semi-gradient update rule.*

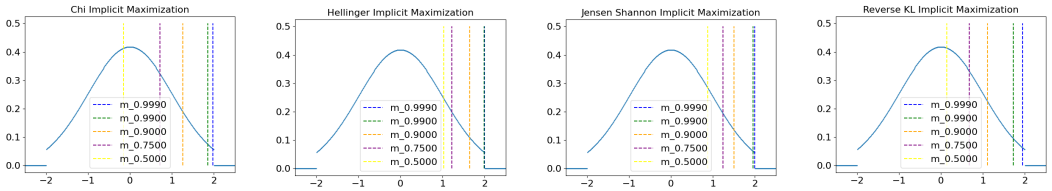

Figure 2: A family of implicit maximizers arising from semi-gradient dual reinforcement learning corresponding to different f-divergences. 10000 datapoints are sampled from 1-D bounded gaussian distribution D and v is inferred using Equation 62. As $\tau \to 1$ (see legend) we obtain more accurate estimates for the supremum of the support.

A family of implicit maximizers: Consider the λ-parameterized semi-gradient `dual-V` objective below:

$$\min_{V(s)}(1-\lambda)\mathbb{E}_{d_O(s)}[V(s)] + \lambda\mathbb{E}_{s,a\sim d^O}\left[f_p^*\left(\left[\hat{Q}(s,a) - V(s)\right],\right)\right] \tag{61}$$

where $\hat{Q}(s,a) = r(s,a) + \gamma\sum_{s'} p(s'|s,a)V(s')$ with hat denoting stop-gradient. More generally for any random variable X with distribution D,

$$\min_{v}(1-\lambda)\mathbb{E}_{x\sim D}[v] + \lambda\mathbb{E}_{x\sim D}\left[f_p^*\left(x - v\right)\right]. \tag{62}$$

We show through Lemma 7 below and a simple example 2 that the semi-gradient form of `dual-V` optimization naturally gives rise to a family of implicit maximizers. Intuitively, this is because the second term in Eq 62 is minimized in value as v increases and saturates once $v = \max(x \in support(D))$ while the first term is minimized for smaller v. This is opposed to specially curated implicit maximizers found in offline RL methods [32]. Gumbel regression becomes a special case of this family. We list some of the loss functions for value function updates with different f-divergences in Appendix A.3.1. We also highlight that the full-gradient variant of the `dual-V` framework for offline RL has been studied extensively in OptiDICE [35].

Lemma 7. *Let X be a real-valued random variable with bounded support and the supremum of the support is x^*. Then optimizing equation 62, the solution v_λ satisfies the following properties*

$$\lim_{\lambda \to 1} v_\lambda = x^* \text{ and } \forall\ \lambda_1 < \lambda_2 \in (0, 1),\ \ v_{\lambda_1} \leq v_{\lambda_2}. \tag{63}$$

A generalized policy iteration view of semi-gradient `dual-V`: `dual-V` framework presents optimization difficulties when using full-gradients [4]. X-QL shows that stable learning can be achieved using the semi-gradient form. Our insight into implicit maximizers suggests that using semi-gradients brings the `dual-V` framework closer to generalized policy-iteration framework. The update of V in its semi-gradient dual form acts as an implicit policy optimizer and the estimation of $\hat{Q}(s, a)$ by regressing to $r(s, a) + \gamma(V(s'))$ is akin to a policy evaluation step, bridging the connection to generalized policy iteration.

Proofs for this section:

Lemma 5. *Conservative Q-Learning (CQL) is the dual of Q-CoP with the generator function $f = (t-1)^2$ (Pearson χ^2) and when the regularization distribution is the replay buffer ($d^O = d^R$).*

Proof. We show that CQL [34], a popular offline RL method is also a special case of `dual-Q` for offline RL. Consider an f-divergence with the generator function $f = (t-1)^2$. The dual function f^* is given by $f^* = (\frac{t^2}{4} + t)$. With the f-divergence our Q-CoP can be written as:

$$\frac{(1-\gamma)}{\alpha}\mathbb{E}_{d_0,\pi(a|s)}[Q(s,a)] + \mathbb{E}_{s,a\sim d^O}\left[\frac{y(s,a,r,s')^2}{4\alpha^2} + \frac{y(s,a,r,s')}{\alpha}\right] \tag{64}$$

$$= \frac{(1-\gamma)}{\alpha}\mathbb{E}_{d_0,\pi(a|s)}[Q(s,a)] + \mathbb{E}_{s,a\sim d^O}\left[\frac{y(s,a,r,s')}{\alpha}\right] + \mathbb{E}_{s,a\sim d^O}\left[\frac{y(s,a,r,s')^2}{4\alpha^2}\right] \tag{65}$$

Let's simplify the first two terms:

$$\frac{1}{\alpha}\left[(1-\gamma)\mathbb{E}_{d_0,\pi(a|s)}[Q(s,a)] + \mathbb{E}_{s,a\sim d^O}\left[r(s,a) + \gamma\sum_{s',a'} p(s'|s,a)\pi(a'|s')Q(s',a') - Q(s,a)\right]\right] \tag{66}$$

$$= \frac{1}{\alpha}\left[(1-\gamma)\mathbb{E}_{d_0,\pi(a|s)}[Q(s,a)] + \mathbb{E}_{s,a\sim d^O}\left[\gamma\sum_{s',a'} p(s'|s,a)\pi(a'|s')Q(s',a')\right] - \mathbb{E}_{s,a\sim d^O}[Q(s,a)] + \cancel{\mathbb{E}_{s,a\sim d^O}[r(s,a)]}\right] \tag{67}$$

$$= \frac{1}{\alpha}\left[(1-\gamma)\sum_{s,a} d_0(s)\pi(a|s)Q(s,a) + \gamma\sum_{s,a} d^O(s,a)\sum_{s'} p(s'|s,a)\pi(a'|s')Q(s',a') - \mathbb{E}_{s,a\sim d^O}[Q(s,a)]\right] \tag{68}$$

$$= \frac{1}{\alpha}\left[(1-\gamma)\sum_{s,a} d_0(s)\pi(a|s)Q(s,a) + \gamma\langle d^O, P^\pi Q\rangle - \mathbb{E}_{s,a\sim d^O}[Q(s,a)]\right] \tag{69}$$

$$= \frac{1}{\alpha}\left[(1-\gamma)\sum_{s,a} d_0(s)\pi(a|s)Q(s,a) + \gamma\langle P^\pi_* d^o, Q\rangle - \mathbb{E}_{s,a\sim d^o}[Q(s,a)]\right] \tag{70}$$

$$= \frac{1}{\alpha}\left[(1-\gamma)\sum_{s,a} d_0(s)\pi(a|s)Q(s,a) + \gamma\sum_{s,a}\pi(a|s)Q(s,a)\sum_{s',a'} p(s|s',a')d(s',a') - \mathbb{E}_{s,a\sim d^o}[Q(s,a)]\right] \tag{71}$$

$$= \frac{1}{\alpha}\left[\sum_{s,a}(d_0(s) + \gamma\sum_{s'a,'} p(s|s',a')d(s',a'))\pi(a|s)Q(s,a) - \mathbb{E}_{s,a\sim d^o}[Q(s,a)] + \mathbb{E}_{s,a\sim d^o}[r(s,a)]\right] \tag{72}$$

$$= \frac{1}{\alpha}\left[\sum_{s,a} d^o(s)\pi(a|s)Q(s,a) - \mathbb{E}_{s,a\sim d^o}[Q(s,a)] + \mathbb{E}_{s,a\sim d^o}[r(s,a)]\right] \tag{73}$$

$$= \frac{1}{\alpha}\left[\mathbb{E}_{s\sim d^o,a\sim\pi}[Q(s,a)] - \mathbb{E}_{s,a\sim d^o}[Q(s,a)]\right] \tag{74}$$

where P^π denotes the policy transition operator, P^π_* denotes the adjoint policy transition operator. Removing constant terms (Equation 67) with respect to optimization variables we end up with the following form for `dual-Q`:

$$\frac{1}{\alpha}\left[\mathbb{E}_{s\sim d^o,a\sim\pi}[Q(s,a)] - \mathbb{E}_{s,a\sim d^o}[Q(s,a)]\right] + \mathbb{E}_{s,a\sim d^o}\left[\frac{y(s,a,r,s')^2}{4\alpha^2}\right] \tag{75}$$

Hence the `dual-Q` optimization reduces to:

$$\max_\pi \min_Q \alpha\left[\mathbb{E}_{s\sim d^o,a\sim\pi}[Q(s,a)] - \mathbb{E}_{s,a\sim d^o}[Q(s,a)]\right] + \mathbb{E}_{s,a\sim d^o}\left[\frac{y(s,a,r,s')^2}{4}\right] \tag{76}$$

This equation matches the unregularized CQL objective (Equation 3 in [34]). □

Lemma 6. *Extreme Q-Learning (X-QL) is the dual of V-CoP with f-divergence set to be the reverse Kullback-Liebler divergence with a semi-gradient update rule.*

Proof. We show that the Extreme Q-Learning [] framework is a special case of the dual framework, specifically the `dual-V` using the semi-gradient update rule.

Consider setting the f-divergence to be the KL divergence in the dual V framework, the regularization distribution and the initial state distribution to be the replay buffer distribution ($d^O = d^R$ and $d_0 = d^R$). The conjugate of the generating function for KL divergence is given by $f^*(t) = e^{t-1}$.

$$\min_{V(s)}(1-\gamma)\mathbb{E}_{d_0(s)}[V(s)] + \mathbb{E}_{s,a\sim d^R}\left[f^*\left(\left[r(s,a) + \gamma\sum_{s'} p(s'|s,a)V(s') - V(s))\right]/\alpha\right)\right] \tag{77}$$

$$\min_{V(s)}(1-\gamma)\mathbb{E}_{d_0(s)}[V(s)] + \mathbb{E}_{s,a\sim d^R}\left[\exp(\left(\left[r(s,a) + \gamma\sum_{s'} p(s'|s,a)V(s') - V(s))\right]/\alpha - 1\right)\right] \tag{78}$$

A popular approach for stable optimization in temporal difference learning is the semi-gradient update rule which has been studied in previous works []. In this update strategy, we fix the targets for the temporal difference backup. Our target in the above optimization is given by:

$$\hat{Q}(s,a) = r(s,a) + \gamma \sum_{s'} p(s'|s,a)V(s') \tag{79}$$

The update equation for V is now given by:

$$\min_{V(s)}(1-\gamma)\mathbb{E}_{d_0(s)}[V(s)] + \mathbb{E}_{s,a\sim d^R}\Big[\exp(\Big(\Big[\hat{Q}(s,a) - V(s))\Big]/\alpha - 1\Big)\Big] \tag{80}$$

where hat denotes the `stop-gradient` operation. We approximate this target by using mean-squared regression with the single sample unbiased estimate as follows:

$$\min_{Q} \mathbb{E}_{s,a,s'\sim d^R}\big[(Q(s,a) - (r(s,a) + V(s')))^2\big] \tag{81}$$

The procedure is now equivalent to Extreme-Q learning and is a special case of the `dual-V` framework.

□

A.3.1 A family of implicit maximizers

Lemma 7. *Let X be a real-valued random variable with bounded support and the supremum of the support is x^*. Then optimizing equation 62, the solution v_λ satisfies the following properties*

$$\lim_{\lambda\to 1} v_\lambda = x^* \text{ and } \forall\ \lambda_1 < \lambda_2 \in (0,1),\ \ v_{\lambda_1} \leq v_{\lambda_2}. \tag{63}$$

Proof. We analyze the behavior of the following optimization of interest.

$$\min_{v}(1-\lambda)\mathbb{E}_{x\sim D}[v] + \lambda\mathbb{E}_{x\sim D}\big[f_p^*\left(x-v\right)\big] \tag{82}$$

$f_p^*(t)$ is given by (using the result derived in A.2):

$$f_p^*(t) = -f\left(\max(f^{'-1}(t),0)\right) + t\max\left(f^{'-1}(t),0\right) \tag{83}$$

The function f_p^* admits two different behaviors given by:

$$f_p^* = \begin{cases} -f(f^{'-1}(t)) + tf^{'-1}(t) = f^*(t), & \text{if } f^{'-1}(t) > 0 \\ -f(0), & \text{otherwise} \end{cases}$$

where f^* is the convex conjugate of f-divergence and is strictly increasing with t. We note other properties related to f function for f-divergences: f^*, f', $(f')^{-1}$ is strictly increasing, $f(0_+) > 0$ and $(f')^{-1}(t) > 0$ when $t > 0$ and 0 otherwise.

We analyze the second term in Eq 82. It can be expanded as follows:

$$\lambda\int_{x:(f')^{-1}(x-v)>0} p(x)f^*(x-v)dx - \lambda\int_{x:(f')^{-1}(x-v)<0} f(0)p(x)dx \tag{84}$$

From the properties of f, we use the fact that $(f^{'})^{-1}(x-v) > 0$ when $x - v > 0$ or equivalently $x > v$.

$$\lambda\int_{x>v} p(x)f^*(x-v)dx - \lambda\int_{x\leq v} f(0)p(x)dx \tag{85}$$

The first term in the above equation decreases monotonically and the second term increases monotonically (thus the combined terms decrease) as v increases until $v = x^*$ (supremum of the support of the distribution) after which the equation assumes a constant value of $-\lambda f(0)$.

Going back to our original optimization in Equation 82, the first term decreases monotonically with v. As $\lambda \to 1$, the minimization of the second term takes precedence, with increasing v until saturation ($v = x^*$). We can go further to characterize the effect of λ on solution v_λ of the equation. The solution of the optimization can be written in closed form as:

$$\frac{(1-\lambda)}{\lambda} = \mathbb{E}_{x\sim D}\Big[f_p^{*'}(x-v)\Big] \tag{86}$$

Using the fact that $f_p^{*'}$ is non-decreasing, we can show that the right-hand term in the equation above increases as v decreases. This in turn implies that for all λ_1, λ_2 such that $\lambda_1 \leq \lambda_2$ we have that $v_{\lambda_1} \leq v_{\lambda_2}$.

□

A.4 Dual Connections to Imitation Learning

A.4.1 Offline imitation learning with expert data only

A new method for offline imitation learning: Analogous to `dual-Q` (offline imitation), we can leverage the `dual-V` (offline imitation) setting which avoids the min-max optimization given by:

`IV-Learn` or `dual-V` (offline imitation from expert-only data):

$$\min_{V(s)}(1-\gamma)\mathbb{E}_{d_0(s)}[V(s)] + \mathbb{E}_{s,a\sim d^E}[f^*\left([\mathcal{T}_0 V(s,a) - V(s))\right]/\alpha)] \tag{87}$$

We propose `dual-V` (offline imitation) to be a new method arising out of this framework which we leave for future exploration.

Proofs for this section:

Corollary 1. *`dual-Q` is equivalent to Implicit Behavior Cloning [7] when $r(s,a) = 0 \ \forall\ (\mathcal{S}, \mathcal{A})$ and $d^O(s,a) = d^E(s,a)$ and f is set to be the total variation divergence.*

Equation 10 suggests that intuitively IQ-learn trains an energy-based model in the form of Q where it pushes down the Q-values for actions predicted by current policy and pushes up the Q-values at the expert state-action pairs. This becomes more clear when the divergence f is chosen to be Total-Variation ($f^* = \mathbb{I}$), IQ learn reduces to:

$$(1-\gamma)\mathbb{E}_{d_0(s),\pi(a|s)}[Q(s,a)] + \mathbb{E}_{s,a\sim d^E}\left[\gamma\sum_{s',a'}p(s'|s,a)\pi(a'|s')Q(s',a') - Q(s,a)\right] \tag{88}$$

$$= \left[(1-\gamma)\mathbb{E}_{d_0(s),\pi(a|s)}[Q(s,a)] + \mathbb{E}_{s,a\sim d^E}\left[\gamma\sum_{s',a'}p(s'|s,a)\pi(a'|s')Q(s',a')\right]\right] - \mathbb{E}_{s,a\sim d^E}[Q(s,a)] \tag{89}$$

Let's simplify the first two terms:

$$(1-\gamma)\mathbb{E}_{d_0(s),\pi(a|s)}[Q(s,a)] + \mathbb{E}_{s,a\sim d^E}\left[\gamma\sum_{s'}p(s'|s,a)\pi(a'|s')Q(s',a')\right] \tag{90}$$

$$= (1-\gamma)\sum_{s,a}d_0(s)\pi(a|s)Q(s,a) + \gamma\sum_{s,a}d^E(s,a)\sum_{s',a'}p(s'|s,a)\pi(a'|s')Q(s',a') \tag{91}$$

$$= (1-\gamma)\sum_{s,a} d_0(s)\pi(a|s)Q(s,a) + \gamma \sum_{s',a'}\sum_{s,a} d^E(s,a)p(s'|s,a)\pi(a'|s')Q(s',a') \tag{92}$$

$$= (1-\gamma)\sum_{s,a} d_0(s)\pi(a|s)Q(s,a) + \gamma \sum_{s',a'} \pi(a'|s')Q(s',a')(\sum_{s,a} d^E(s,a)p(s'|s,a)) \tag{93}$$

$$= (1-\gamma)\sum_{s,a} d_0(s)\pi(a|s)Q(s,a) + \gamma \sum_{s',a'} \pi(a'|s')Q(s',a')(\sum_{s,a} d^E(s,a)p(s'|s,a)) \tag{94}$$

$$= (1-\gamma)\sum_{s,a} d_0(s)\pi(a|s)Q(s,a) + \gamma \sum_{s,a} \pi(a|s)Q(s,a)(\sum_{s',a'} d^E(s',a')p(s|s',a')) \tag{95}$$

$$= \sum_{s,a}(1-\gamma)d_0(s)\pi(a|s)Q(s,a) + \pi(a|s)Q(s,a)(\sum_{s',a'} d^E(s',a')p(s|s',a')) \tag{96}$$

$$= \sum_{s,a} \pi(a|s)Q(s,a)\left[(1-\gamma)d_0(s) + \gamma \sum_{s',a'} d^E(s',a')p(s|s',a')\right] \tag{97}$$

$$= \sum_{s,a} \pi(a|s)Q(s,a)d^E(s) \tag{98}$$

where the last step is due to the steady state property of the MDP (Bellman flow constraint).

Therefore IQ-Learn/`dual-Q` for offline imitation (in the special case of TV divergence) simplifies to (from Equation 89):

$$\left[(1-\gamma)\mathbb{E}_{d_0(s),\pi(a|s)}[Q(s,a)] + \mathbb{E}_{s,a\sim d^E}\left[\gamma \sum_{s',a'} p(s'|s,a)\pi(a'|s')Q(s',a')\right]\right] - \mathbb{E}_{s,a\sim d^E}[Q(s,a)] \tag{99}$$

$$= \min_Q \mathbb{E}_{d_E(s),\pi(a|s)}[Q(s,a)] - \mathbb{E}_{s,a\sim d^E}[Q(s,a)] \tag{100}$$

The gradient w.r.t for the above optimization matches the gradient update of Implicit Behavior Cloning [7] with Q as the energy-based model.

A.5 Off-policy imitation learning (under coverage assumption)

First, we show that is easy to see why choosing the f-divergence to be reverse KL makes it possible to get an off-policy objective for imitation learning in the dual framework. We start with the Q-CoP for imitation learning using the reverse KL-divergence ($r(s,a) = 0$ and $d^o = d^E$):

$$\max_{d(s,a)\geq 0, \pi(a|s)} -D_{\text{KL}}(d(s,a) \,||\, d^E(s,a))$$
$$\text{s.t } d(s,a) = (1-\gamma)\rho_0(s).\pi(a|s) + \gamma\pi(a|s)\sum_{s',a'} d(s',a')p(s|s',a'). \tag{101}$$

Under the assumption that the replay buffer visitation (denoted by d^R) covers the expert visitation ($d^R > 0$ wherever d^E>0) [37], which we refer to as the **coverage assumption**, the reverse KL divergence can be expanded as follows:

$$D_{\text{KL}}(d(s,a) \,||\, d^E(s,a)) = \mathbb{E}_{s,a\sim d(s,a)}\left[\log \frac{d(s,a)}{d^E(s,a)}\right] = \mathbb{E}_{s,a\sim d(s,a)}\left[\log \frac{d(s,a)}{d^E(s,a)}\frac{d^R(s,a)}{d^R(s,a)}\right] \tag{102}$$

$$= \mathbb{E}_{s,a\sim d(s,a)}\left[\log \frac{d(s,a)}{d^R(s,a)} + \log \frac{d^R(s,a)}{d^E(s,a)}\right] \tag{103}$$

$$= \mathbb{E}_{s,a\sim d(s,a)}\left[\log \frac{d^R(s,a)}{d^E(s,a)}\right] + D_{\text{KL}}(d(s,a) \,||\, d^R(s,a)). \tag{104}$$

Hence the Q-CoP can now be written as:

$$\max_{d(s,a)\geq 0, \pi(a|s)} \mathbb{E}_{s,a\sim d(s,a)}\left[-\log \frac{d^R(s,a)}{d^E(s,a)}\right] - D_{\mathrm{KL}}(d(s,a) \,||\, d^R(s,a)) \tag{105}$$

$$\text{s.t } d(s,a) = (1-\gamma)\rho_0(s).\pi(a|s) + \gamma \sum_{s',a'} d(s',a')p(s|s',a')\pi(a|s). \tag{106}$$

Now, in the optimization above the first term resembles the reward function and the second term resembles the divergence constraint with a new distribution $d^R(s,a)$ in the original regularized RL primal (Eq 22). Hence we can obtain respective `dual-Q` and `dual-V` in the setting for off-policy imitation learning using the reward function as $r^{imit}(s,a) = -\log \frac{d^R(s,a)}{d^E(s,a)}$ and the new regularization distribution as $d^R(s,a)$. Using $\mathcal{T}^{\pi}_{r^{imit}}$ and $\mathcal{T}_{r^{imit}}$ to denote backup operators under new reward function r^{imit}, we have

`dual-Q` for off-policy imitation (coverage assumption):

$$\max_{\pi(a|s)} \min_{Q(s,a)} (1-\gamma)\mathbb{E}_{\rho_0(s),\pi(a|s)}[Q(s,a)] + \mathbb{E}_{s,a\sim d^R}[f^*(\mathcal{T}^{\pi}_{r^{imit}}Q(s,a) - Q(s,a))]. \tag{107}$$

This choice of KL divergence leads us to a reduction of another method, OPOLO [57] for off-policy imitation learning to `dualQ` which we formalize in the lemma below:

Lemma 8. *`dual-Q` for off-policy imitation learning reduces to OPOLO [57], with the f-divergence set to the reverse KL divergence when $r(s,a) = 0\ \forall \mathcal{S}, \mathcal{A}$, $d^O = d^E$ and under the assumption that the replay data distribution covers the expert data distribution.*

Analogously we have `dual-V` for off-policy imitation (coverage assumption):

$$\min_{V(s)} (1-\gamma)\mathbb{E}_{\rho_0(s)}[V(s)] + \mathbb{E}_{s,a\sim d^R}[f^*(\mathcal{T}_{r^{imit}}V(s,a) - V(s))]. \tag{108}$$

We note that the `dual-V` framework for off-policy imitation learning under coverage assumptions was studied in the imitation learning work SMODICE [37].

B Off-policy imitation learning with relaxed coverage

We now derive our proposed method for imitation learning with arbitrary data. The derivation for the `dual-Q` setting is shown below. `dual-V` derivation can be done similarly.

Lemma 3. ***(`dual-Q` for off-policy imitation (relaxed coverage assumption))*** *Imitation learning using off-policy data can be solved by optimizing the following modified dual objective for Q-CoP with $r(s,a) = 0\ \forall \mathcal{S}, \mathcal{A}$ and f-divergence considered between distributions $d^R_{mix} := \beta d(s,a) + (1-\beta)d^R(s,a)$ and $d^{E,R}_{mix} := \beta d^E(s,a) + (1-\beta)d^R(s,a)$, and is given by:*

$$\max_{\pi(a|s)} \min_{Q(s,a)} \beta(1-\gamma)\mathbb{E}_{d_0(s),\pi(a|s)}[Q(s,a)] + \mathbb{E}_{s,a\sim d^{E,R}_{mix}}\left[f^*_p(\mathcal{T}^{\pi}_0 Q(s,a) - Q(s,a))\right] - (1-\beta)\mathbb{E}_{s,a\sim d^R}[\mathcal{T}^{\pi}_0 Q(s,a) - Q(s,a)] \tag{14}$$

Proof.

$$\max_{\pi, d\geq 0} \min_{Q(s,a)} \alpha\mathbb{E}_{s,a\sim d}[r(s,a)] - D_{\mathrm{f}}(d^R_{mix} \,||\, d^{E,R}_{mix}) + \alpha \sum_{s,a} Q(s,a)\left((1-\gamma)d_0(s).\pi(a|s) + \gamma \sum_{s',a'} d(s',a')p(s|s',a')\pi(a|s) - d(s,a)\right)$$

We can use the same algebraic machinery as before (Section A.1.2) to get an unconstrained tractable optimization problem:

$$\max_{\pi, d\geq 0} \min_{Q(s,a)} \alpha \mathbb{E}_{s,a\sim d(s,a)}[r(s,a)] - D_{\mathrm{f}}(d^R_{mix} \,||\, d^{E,R}_{mix})$$
$$+ \alpha \sum_{s,a} Q(s,a) \left((1-\gamma) d_0(s).\pi(a|s) + \gamma \sum_{s',a'} d(s',a') p(s|s',a') \pi(a|s) - d(s,a) \right) \tag{109}$$

$$= \max_{\pi, d\geq 0} \min_{Q(s,a)} \alpha(1-\gamma) \mathbb{E}_{d_0(s),\pi(a|s)}[Q(s,a)]$$
$$+ \alpha \mathbb{E}_{s,a\sim d} \left[r(s,a) + \gamma \sum_{s'} p(s'|s,a) \pi(a'|s') Q(s',a') - Q(s,a) \right] - D_{\mathrm{f}}(d^R_{mix} \,||\, d^{E,R}_{mix}) \tag{110}$$

$$= \max_{\pi, d\geq 0} \min_{Q(s,a)} \alpha(1-\gamma) \mathbb{E}_{d_0(s),\pi(a|s)}[Q(s,a)]$$
$$+ \alpha \mathbb{E}_{s,a\sim d} \left[r(s,a) + \gamma \sum_{s'} p(s'|s,a) \pi(a'|s') Q(s',a') - Q(s,a) \right]$$
$$+ (1-\alpha) \mathbb{E}_{s,a\sim d^R} \left[r(s,a) + \gamma \sum_{s'} p(s'|s,a) \pi(a'|s') Q(s',a') - Q(s,a) \right]$$
$$- (1-\alpha) \mathbb{E}_{s,a\sim d^R} \left[r(s,a) + \gamma \sum_{s'} p(s'|s,a) \pi(a'|s') Q(s',a') - Q(s,a) \right] - D_{\mathrm{f}}(d^R_{mix} \,||\, d^{E,R}_{mix}) \tag{111}$$

Imitation from Arbitrary data (dualQ, no positivity constraints)

$$= \max_{\pi(a|s)} \min_{Q(s,a)} \max_{d\geq 0} \alpha(1-\gamma) \mathbb{E}_{d_0(s),\pi(a|s)}[Q(s,a)]$$
$$+ \mathbb{E}_{s,a\sim d^R_{mix}} \left[r(s,a) + \gamma \sum_{s'} p(s'|s,a) \pi(a'|s') Q(s',a') - Q(s,a) \right] - D_{\mathrm{f}}(d^R_{mix} \,||\, d^{E,R}_{mix})$$
$$- (1-\alpha) \mathbb{E}_{s,a\sim d^R} \left[r(s,a) + \gamma \sum_{s'} p(s'|s,a) \pi(a'|s') Q(s',a') - Q(s,a) \right] \tag{112}$$

Note that the inner maximization with respect to d has the constraint that $d \geq 0$. This constraint was not necessary for the previous settings for `dual-Q` problems we have discussed. In this setting, to get a tractable closed form we replace the optimization variable from d to d^R_{mix} with the constraint that $d \geq 0$. This prevents the optimization to result in values for d^R_{mix} which has $d < 0$. Ignoring this constraint ($d \geq 0$) results in the following dual-optimization for imitation from arbitrary data.

$$\max_{\pi(a|s)} \min_{Q(s,a)} \alpha(1-\gamma) \mathbb{E}_{d_0(s),\pi(a|s)}[Q(s,a)]$$
$$+ \mathbb{E}_{s,a\sim d^{E,R}_{mix}} \left[f^*(r(s,a) + \gamma \sum_{s'} p(s'|s,a) \pi(a'|s') Q(s',a') - Q(s,a)) \right]$$
$$- (1-\alpha) \mathbb{E}_{s,a\sim d^R} \left[r(s,a) + \gamma \sum_{s'} p(s'|s,a) \pi(a'|s') Q(s',a') - Q(s,a) \right] \tag{113}$$

To incorporate the positivity constraints we begin on the inner maximization w.r.t d^R_{mix} and consider the terms dependent on d^R_{mix} below.

$$\max_{d^R_{mix}, d \geq 0} \mathbb{E}_{s,a \sim d^R_{mix}} \left[r(s,a) + \gamma \sum_{s'} p(s'|s,a) \pi(a'|s') Q(s',a') - Q(s,a) \right] - D_{\mathrm{f}}(d^R_{mix} \,||\, d^{E,R}_{mix}) \tag{114}$$

Let $p(s,a) = \frac{(1-\alpha)\rho^R(s,a)}{\alpha\rho^E(s,a) + (1-\alpha)\rho^R(s,a)}$, $y(s,a) = r(s,a) + \gamma \sum_{s'} p(s'|s,a)\pi(a'|s')Q(s',a') - Q(s,a)$ and $w(s,a) = \frac{d^R_{mix}(s,a)}{d^E_{mix}(s,a)}$. We construct the lagrangian dual to incorporate the constraint $d \geq 0$ in its equivalent form $w(s,a) \geq p(s,a)$ and obtain the following:

$$\max_{w(s,a)} \max_{\lambda \geq 0} \mathbb{E}_{s,a \sim d^{E,R}_{mix}}[w(s,a)y(s,a)] - \mathbb{E}_{d^{E,R}_{mix}}[f(w(s,a))] + \sum_{s,a} \lambda(w(s,a) - p(s,a)) \tag{115}$$

Since strong duality holds, we can use the KKT constraints to find the solutions $w^*(s,a)$ and $\lambda^*(s,a)$.

Primal feasibility: $w^*(s,a) \geq p(s,a) \;\; \forall \; s,a$
Dual feasibility: $\lambda^* \geq 0 \;\; \forall \; s,a$
Stationarity: $d^{E,R}_{mix}(s,a)(f'(w^*(s,a)) + y(s,a) + \lambda^*(s,a)) = 0 \;\; \forall \; s,a$
Complementary Slackness: $(w^*(s,a) - p(s,a))\lambda^*(s,a) = 0 \;\; \forall \; s,a$

Using stationarity we have the following:

$$f'(w^*(s,a)) = y(s,a) + \lambda^*(s,a) \;\; \forall \; s,a \tag{116}$$

Now using complementary slackness only two cases are possible $w^*(s,a) > p(s,a)$ or $\lambda^*(s,a) > 0$. Combining both cases we arrive at the following solution for this constrained optimization:

$$w^*(s,a) = \max\left(p(s,a), {f'}^{-1}(y(s,a))\right) \tag{117}$$

We can still find a closed-form solution for the inner optimization, in the case when $d \geq 0$, although a bit more involved (See Appendix for the proof). Let $y(s,a) = r(s,a) + \gamma \sum_{s'} p(s'|s,a)\pi(a'|s')Q(s',a') - Q(s,a)$. Also let $p(s,a) = \frac{(1-\alpha)\rho^R(s,a)}{\alpha\rho^E(s,a) + (1-\alpha)\rho^R(s,a)}$.

$$\begin{aligned} &\max_{\pi(a|s)} \min_{Q(s,a)} \alpha(1-\gamma)\mathbb{E}_{d_0(s),\pi(a|s)}[Q(s,a)] \\ &+ \mathbb{E}_{s,a \sim d^{E,R}_{mix}}\left[\max\left(p(s,a), (f')^{-1}\left(y(s,a)\right)\right) y(s,a) - \alpha f\left(\max\left(p(s,a), (f')^{-1}\left(y(s,a)\right)\right)\right)\right] \end{aligned} \tag{118}$$

$$- (1-\alpha)\mathbb{E}_{s,a \sim d^R}\left[r(s,a) + \gamma \sum_{s'} p(s'|s,a)\pi(a'|s')Q(s',a') - Q(s,a) \right] \tag{119}$$

Thus, the closed-form solution with the positivity constraints requires us to estimate the ratio $p(s,a)$ which is possible by learning a discriminator. We observed in our experiments that ignoring the positivity constraints still resulted in a performant method while having the benefits of being simple. A similar derivation can be done in V-space to obtain an analogous result. □

C Implementation and Experiment Details

Environments: In this work for benchmarking we use 4 MuJoCo (licensed under CC BY 4.0) locomotion environments: Hopper, Walker2d, HalfCheetah, and Ant. .

Offline datasets: In this task, we use offline dataset of environments interactions from D4RL [8]. We consider the following MuJoCo environments: Our dataset composition for 'random+expert'

is similar to SMODICE [37] where we use a mixture of a small number of expert trajectories ($\leq$ 200 trajectories) and a large number of low-quality trajectories from the "random-v2" dataset (1 million transitions). We similarly create another offline dataset 'medium+expert' consisting of 200 expert trajectories and 1 million medium-quality transitions from the "medium-v2". The 'random+few-expert' dataset is similar to the 'random+expert' dataset except that only 30 expert trajectories are present in the offline dataset.

Expert dataset The offline dataset for imitation consists of 1000 transitions obtained from the "expert-v2" dataset for the respective environment.

Baselines: We compare our proposed methods against 4 representative methods for offline imitation learning with suboptimal data – SMODICE [37], RCE [6], ORIL [59] and IQLearn [14]. We do not compare to DEMODICE [29] as SMODICE was shown to be competitive in [37]. SMODICE is an imitation method emerging from the dual framework but under an restrictive coverage assumption. ORIL adapts GAIL [21] to the offline setting by using an offline RL algorithm for policy optimization. RCE baseline in the paper combine RCE (Eysenbach et al., 2021), the state-of-art online example-based RL method, and TD3-BC. ORIL and RCE share the same state-action based discriminator as in SMODICE, and TD3-BC [11] as the offline RL algorithm. All the approaches only have access to expert state-action trajectory.

We use the author's open-source implementations of baselines SMODICE, RCE, ORIL available at `https://github.com/JasonMa2016/SMODICE`. We use the author-provided hyperparameters (similar to those used in [37]) for all MuJoCo locomotion environments. IQ-Learn was tested on our expert dataset by following authors implementation found here: `https://github.com/Div99/IQ-Learn`. We tested two IQ-Learn loss variants: 'v0' and 'value' as found in their hyperparameter configurations and took the best out of the two runs.

Policy Optimization: We use Method 1 in Section A.1.3 for policy update.

C.1 Hyperparameters

Hyperparameters for our proposed offpolicy imitation learning method `ReCOIL` are shown in Table 3.

Hyperparameter	Value
Policy updates n_{pol}	1
Policy learning rate	3e-5
Value learning rate	3e-4
Temperature α	0.1
f-divergence	χ^2

Table 3: Hyperparameters for `ReCOIL`.

D Experimental Results

D.1 The failure of ADP-based traditional off-policy algorithms

Figure 3 shows that methods like SAC, SACfD deteriorate increasingly with increasing action dimension when bootstrapped with off-policy data. Figure 4 shows that traditional ADP methods suffer from overestimation during training.

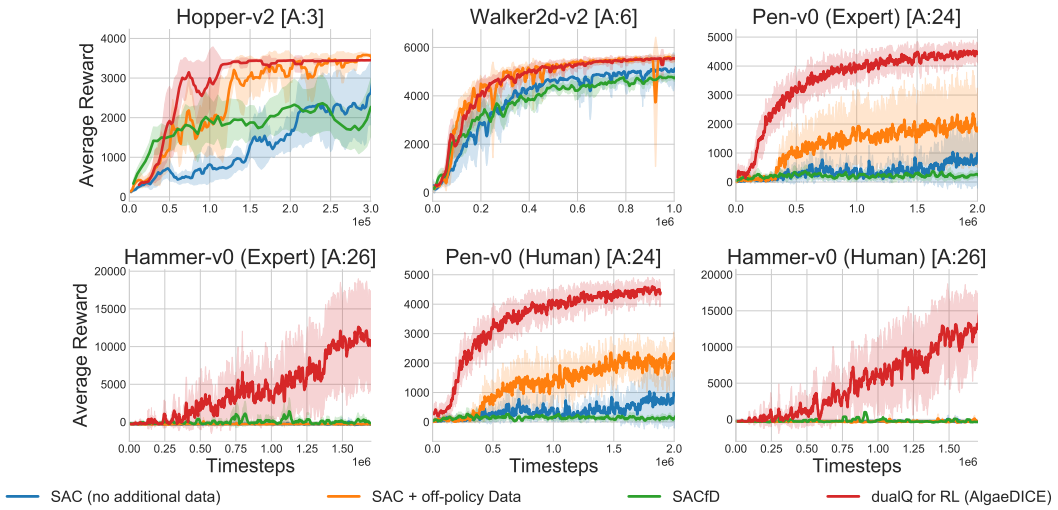

Figure 3: Despite the promise of off-policy methods, current methods based on ADP such as SAC fail when the dimension of action space, denoted by A, increases even when helpful data is added to their replay buffer. On other hand, dual-Q methods are able to leverage off-policy data to increase their learning performance

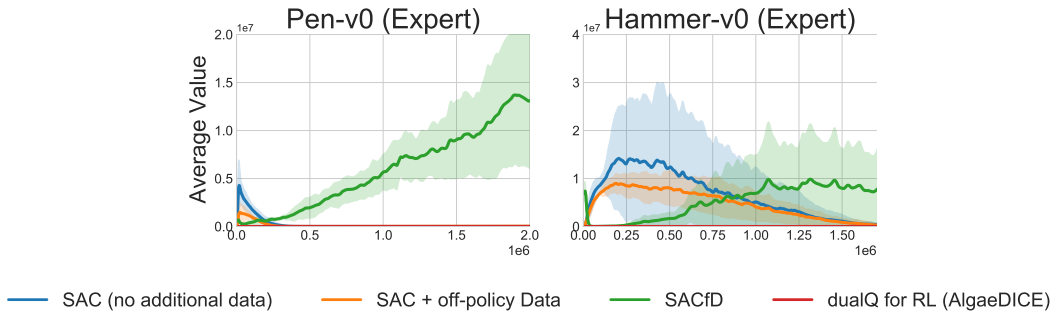

Figure 4: SAC and SACfD suffer from overestimation when off-policy data is added to the replay buffer. We hypothesize this to cause instabilities during training while dualQ has no overestimation.

D.2 Does `ReCOIL` allow for better estimation of agent visitation distribution?

We consider two didactic environments which demonstrate the failures of method that either do not utilize off-policy data (IQ-Learn) or relies on a coverage assumption (SMODICE). ReCOIL is able to perfectly infer agent's visitation when replay buffer covers agent ground truth visitation perfectly (Fig 5) and is able to outperform baselines when the replay buffer has imperfect coverage over the agent's ground truth visitation (Fig 6).

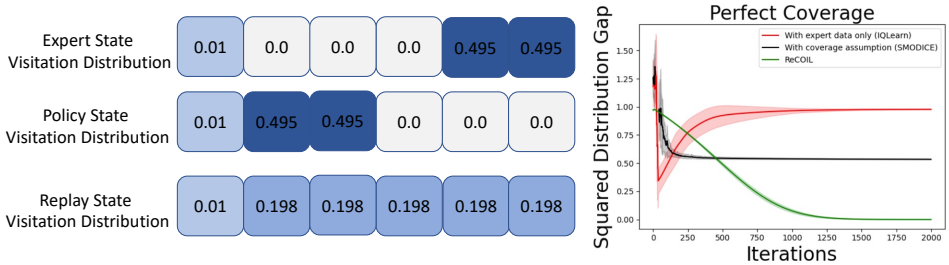

Figure 5: The replay buffer distribution covers the agent policy visitation distribution. Using `ReCOIL`, we are perfectly able to infer the agent policy visitation whereas a method that only relies on expert data or the replay data with the coverage assumption fails. Results are averaged over 100 seeds.

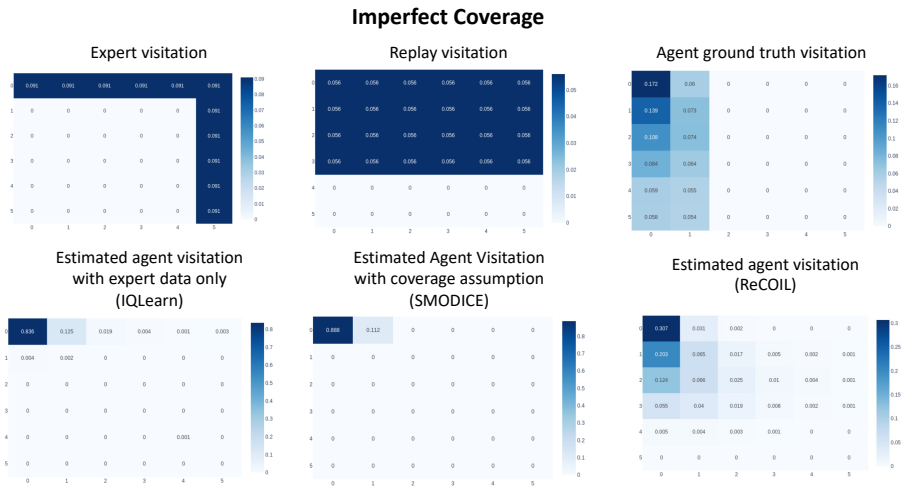

Figure 6: Replay buffer consists of data that visits near the initial state (0,0), a setting commonly observed when training RL agents. We estimate agent's policy visitation and observe `ReCOIL` to outperform both methods which rely on expert data only or use the replay data with coverage assumption

D.3 Benchmarking performance of ReCOIL on MuJoCo tasks

We show learning curves for ReCOIL in Figure 7 below.

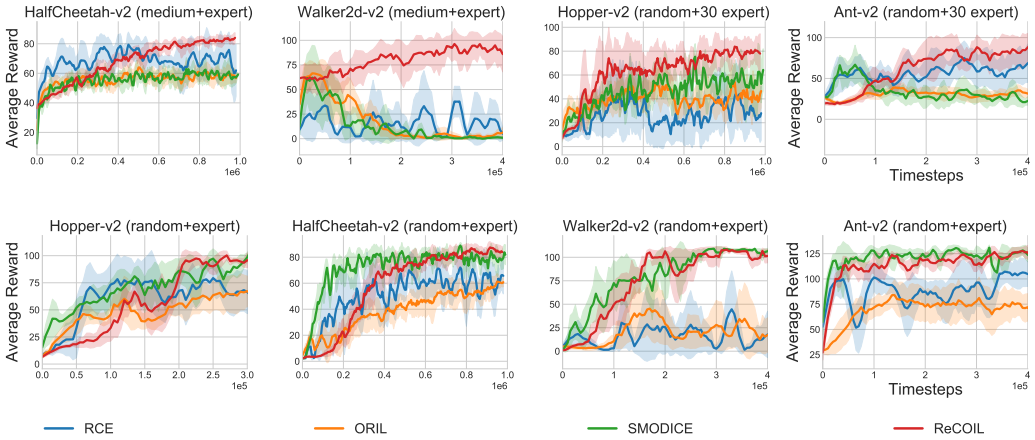

Figure 7: ReCOIL performs competitively in the setting of learning to imitate from diverse offline data. The results are averaged over 5 seeds