HELLINGER DISTANCE CONSTRAINED REGRESSION

Anonymous authors
Paper under double-blind review

ABSTRACT

This paper introduces the off-policy reinforcement learning method that uses the Hellinger distance between sampling policy and current policy as a constraint. Hellinger distance squared multiplied by two is greater than or equal to total variation distance squared and less than or equal to Kullback-Leibler divergence, therefore lower bound for expected discounted return for the new policy is improved comparing to KL. Also, Hellinger distance is less than or equal to 1, so there is a policy-independent lower bound for expected discounted return. HDCR is capable of training with Experience Replay, a common setting for distributed RL when collecting trajectories using different policies and learning from this data centralized. HDCR shows results comparable to or better than Advantage-weighted Behavior Model and Advantage-Weighted Regression on MuJoCo tasks using offline datasets collected by random agents and datasets obtained during the first iterations of online training of HDCR agent.

1 INTRODUCTION

Policy gradient algorithms are methods of model-free reinforcement learning that optimize policy through differentiating expected discounted return. Despite the simplicity, to converge, these methods should stay on-policy because of the first-order approximation of state visitation frequencies. This issue makes agents learn through trial-and-error, using data only once.

To make policy gradient updates more off-policy, we can add a constraint on the update to decrease the step size if the current policy is too far from the sampling policy. One of the first methods was to add total variation distance squared as a constraint for mixture policies. Later it was proven that there is a lower bound for new policy's expected discounted return (Kakade & Langford, 2002). Recently it was proven that this lower bound exists for all types of updates (Schulman et al., 2015).

Next, total variation distance squared was replaced by Kullback-Leibler divergence that is greater than or equal to the previous one (Pinker's inequality (Levin & Peres, 2017)), so that the lower bound was decreased (Schulman et al., 2015). With the use of Lagrangian, have been derived off-policy method called Advantage-Weighted Regression (Peng et al., 2019), which also used KL as a constraint.

This article proposes a new method whose lower bound of expected discounted return is greater than or equal to the bound with KL. We achieve this by replacing total variation distance by Hellinger distance, which decreases lower bound. Therefore strictness stays the same. Then we derive an off-policy method called Hellinger Distance Constrained Regression using the new constraint. It can be used on discrete and continuous action spaces since derivation uses Lebesgue integrals rather than a summation or Riemann integrals.

2 PRELIMINARIES

To better present the problem, we start from basic definitions, go through the history of improvements, and then describe the disadvantages of using KL divergence as a constraint.

We consider an infinite-horizon discounted Markov decision process (MDP), defined by the tuple $(\mathcal{S},\ \mathcal{A},\ P,\ r,\ \rho_0,\ \gamma)$, where $\mathcal{S}$ is a set of states (finite or infinite), $\mathcal{A}$ is a set of actions (finite or infinite), $P:\ \mathcal{S}\ \times\ \mathcal{A}\ \times\ \mathcal{S}\ \rightarrow\ \mathbb{R}$ is the transition probability distribution, $r:\ \mathcal{S}\ \rightarrow\ \mathbb{R}$ is

the reward function, $\rho_0 : \mathcal{S} \rightarrow \mathbb{R}$ is the distribution of the initial state s_0, and $\gamma \in (0, 1)$ is the discount factor.

Let π denote a stochastic policy $\pi : \mathcal{S} \times \mathcal{A} \rightarrow [0, 1]$, and then its expected discounted return is:

$$\begin{aligned} \eta(\pi) &= \mathbb{E}_{s_0, a_0, \ldots} \left[\sum_{t=0}^{\infty} \gamma^t r(s_t) \right], \text{ where} \\ s_0 &\sim \rho_0(\cdot),\ a_t \sim \pi(\cdot|s_t),\ s_{t+1} \sim P(\cdot|s_t, a_t). \end{aligned} \tag{1}$$

This paper uses state-action value function Q_π, state value function V_π, and advantage function A_π with the following definitions:

$$\begin{aligned} Q_\pi(s_t, a_t) &= \mathbb{E}_{s_{t+1}, a_{t+1}, \ldots} \left[\sum_{l=0}^{\infty} \gamma^l r(s_{t+l}) \right] \\ V_\pi(s_t) &= \mathbb{E}_{a_t, s_{t+1}, a_{t+1}, \ldots} \left[\sum_{l=0}^{\infty} \gamma^l r(s_{t+l}) \right] \\ A_\pi(s_t, a_t) &= Q_\pi(s_t, a_t) - V_\pi(s_t). \end{aligned} \tag{2}$$

Let $\rho_\pi(s)$ be unnormalized visitation frequencies of state s where actions are chosen according to π:

$$\rho_\pi(s) = \sum_{t=0}^{\infty} \gamma^t P(s_t = s). \tag{3}$$

Following identity expresses the expected return of another policy $\tilde{\pi}$ in terms of the advantage over π, accumulated over states (see Schulman et al. (2015) for proof):

$$\eta(\tilde{\pi}) = \eta(\pi) + \int \rho_{\tilde{\pi}}(s) \int \tilde{\pi}(a|s) A_\pi(s, a)\ da\ ds. \tag{4}$$

In approximately optimal learning, we replace state visitation frequency $\rho_{\tilde{\pi}}$ by ρ_π, since this drastically decrease optimization complexity:

$$L_\pi(\tilde{\pi}) = \eta(\pi) + \int \rho_\pi(s) \int \tilde{\pi}(a|s) A_\pi(s, a)\ da\ ds. \tag{5}$$

Let π_{old} denote current policy, then the lower bound for the expected discounted return for the new policy π_{new} will be (see Schulman et al. (2015) for proof):

$$\begin{aligned} \eta(\pi_{new}) &\geq L_{\pi_{old}}(\pi_{new}) - \frac{4\epsilon\gamma}{(1-\gamma)^2}\alpha^2 \\ \text{where } \epsilon &= \max_{s,a} |A_\pi(s, a)|, \\ \alpha &= \max_s D_{TV}(\pi_{old}(\cdot|s)||\pi_{new}(\cdot|s)), \\ D_{TV}(\pi_{old}(\cdot|s)||\pi_{new}(\cdot|s)) &= \frac{1}{2} \int |\pi_{old}(a|s) - \pi_{new}(a|s)|\ da. \end{aligned} \tag{6}$$

Theoretical Trust-Region Policy Optimization algorithm relays on Pinsker's inequality (see (Tsybakov, 2009) for proof):

$$\begin{aligned} D_{KL}(\pi_{old}(\cdot|s)||\pi_{new}(\cdot|s)) &\geq D_{TV}(\pi_{old}(\cdot|s)||\pi_{new}(\cdot|s))^2 \\ \text{where } D_{KL}(\pi_{old}(\cdot|s)||\pi_{new}(\cdot|s)) &= \int \pi_{old}(a|s) \log \frac{\pi_{old}(a|s)}{\pi_{new}(a|s)}\ da. \end{aligned} \tag{7}$$

To retain strictness and decrease calculation complexity, total variation distance squared was replaced with Kullback-Leibler divergence $D_{KL}(\pi_{old}(\cdot|s)||\pi_{new}(\cdot|s))$:

$$\begin{aligned}\eta(\pi_{new}) &\geq L_{\pi_{old}}(\pi_{new}) - C D_{KL}^{max}(\pi_{old}||\pi_{new}) \\ \text{where } \epsilon &= \max_{s,a} |A_\pi(s,a)|, \\ C &= \frac{4\epsilon\gamma}{(1-\gamma)^2}, \\ D_{KL}^{max}(\pi_{old}||\pi_{new}) &= \max_s D_{KL}(\pi_{old}(\cdot|s)||\pi_{new}(\cdot|s)).\end{aligned} \tag{8}$$

However, this replacement greatly decreases the lower bound for the expected discounted return for the new policy. Moreover, Kullback-Leibler divergence has no upper bound. Therefore we have no policy-independent lower bound for this type of update.

3 Hellinger Distance in Policy Optimization

We can improve lower bound (comparing to KL) by replacing $D_{TV}(\pi_{old}(\cdot \mid s) \mid\mid \pi_{new}(\cdot \mid s))$ with Hellinger distance $H(\pi_{old}(\cdot|s) \mid\mid \pi_{new}(\cdot \mid s))$:

$$H(\pi_{old}(\cdot|s) \mid\mid \pi_{new}(\cdot|s))^2 = 1 - \int \sqrt{\pi_{old}(a|s)\, \pi_{new}(a|s)}\, da \tag{9}$$

Theorem 1 (see Appendix A or (Tsybakov, 2009, section 2.4) for proof) proves correctness and improvement (comparing to KL) to the lower bound.

Let $p(v)$ and $q(v)$ be two probability density functions then:

$$D_{TV}(p(\cdot) \mid\mid q(\cdot))^2 \leq 2H(p(\cdot) \mid\mid q(\cdot))^2 \leq D_{KL}(p(\cdot) \mid\mid q(\cdot)) \tag{10}$$

Replacing $p(v)$ and $q(v)$ with $\pi_{old}(\cdot \mid s)$ and $\pi_{new}(\cdot \mid s)$ respectively, new lower bound follows:

$$\begin{aligned}\eta(\pi_{new}) &\geq L_{\pi_{old}}(\pi_{new}) - \frac{8\epsilon\gamma}{(1-\gamma)^2}\alpha^2 \\ \text{where } \epsilon &= \max_{s,a} |A_\pi(s,a)|, \\ \alpha &= \max_s H(\pi_{old}(\cdot|s)||\pi_{new}(\cdot|s))\end{aligned} \tag{11}$$

It is worth to note that $H(\pi_{old}(\cdot \mid s) \mid\mid \pi_{new}(\cdot \mid s)) \;\leq\; 1$.

4 Hellinger Distance Constrained Regression (HDCR)

We could use presented lower bound as in TRPO, but instead, we derive an offline regression algorithm by introducing the following optimization problem where μ is the sampling policy:

$$\begin{aligned}\arg\max_\pi & \int \rho_\mu(s) \int \pi(a|s) A_\mu(s,a)\, da\, ds \\ \text{s.t.} & \int \rho_\mu(s) H(\pi(\cdot|s)||\mu(\cdot|s))\, ds \leq \epsilon, \\ & \int \pi(a|s)\, da = 1,\ \forall s \in \mathcal{S}.\end{aligned} \tag{12}$$

Constructing Lagrangian, differentiating it with respect to π, and solving for π gives us the following optimal policy (see Appendix B for derivation):

$$\pi^*(a|s) = \mu(a|s) \frac{\beta^2}{(\beta - 2A_\mu(s,a))^2} \tag{13}$$

Constructing a regression problem of KL divergence between optimal policy π^* and current policy π and simplifying gives us the following supervised regression problem (see Appendix B for derivation):

$$\arg\max_\pi \mathbb{E}_{s\sim\rho_\mu(\cdot)}\, \mathbb{E}_{a\sim\mu(\cdot|s)} \log \pi(a|s) \frac{1}{(\beta - A_\mu(s,a))^2} \tag{14}$$

Using notation of ABM paper (Siegel et al., 2020) "advantage-weighting" function is $f(A(s,a)) = \frac{1}{(\beta - A(s,a))^2}$.

If we use HDCR with Experience Replay, in equation 14, we replace $\mu(\cdot|s)$ in expectation $\mathbb{E}_{a \sim \mu(\cdot \mid s)}$ and advantage function $A_\mu(s,\ a)$. Let $\Pi = \{\pi_i,\ \pi_{i+1},\ ...,\ \pi_{i+N}\}$ be a set of sampling policies from which actions were sampled, $w(\pi_i)$ probability of selecting policy π_i, then:

$$\begin{aligned} \mu(s,a) &= \int_\Pi w(\pi)\rho_\pi(s)\pi(a|s)\ d\pi \\ A_\mu(s,a) &= \frac{\int_\Pi w(\pi)\rho_\pi(s)\left(Q_\pi(s,a) - V_\pi(s)\right)\ d\pi}{\int_\Pi w(\pi)\rho_\pi(s)\ d\pi} \\ \overline{V}(s) &= \frac{\int_\Pi w(\pi)\rho_\pi(s)V_\pi(s)\ d\pi}{\int_\Pi w(\pi)\rho_\pi(s)\ d\pi} \end{aligned} \tag{15}$$

The proof will repeat the proof for AWR with Experience Replay (Peng et al., 2019).

Practically we simply sample uniformly from the replay buffer and using a value function estimator. This type of sampling provides us an approximation of expectation and state value function.

Let $\mathcal{D}$ denote a set of stored trajectories (replay buffer), $A(s,a;\phi)$ is an advantage function parameterized by vector ϕ.

The most popular method to obtain $A(s,a;\phi)$ for offline learning is to use state-action function estimator $Q(s,a;\phi)$ parameterized by ϕ:

$$A(s,a;\phi) = Q(s,a;\phi) - \int \pi(a'|s;\theta_k)Q(s,a';\phi_k)da' \tag{16}$$

Despite $Q(s,a;\phi)$ being closer to the expectation of discounted return following policy π (because of "taking" the first action according to π rather then μ), we found Monte-Carlo return more efficient on tiny offline datasets. Greater performance using MC return can be explained by a lack of experience "produced" by certain actions.

Monte-Carlo estimation of $A(s,a;\phi)$ can be described as follows where $\mathcal{R}^{\mathcal{D}}_{s_t,a_t} = \sum_{l=0}^{T}\gamma^l r_{t+l}$ and $V(s;\phi)$ is a state value function estimator parameterized by vector ϕ:

$$A(s,a;\phi) = \mathcal{R}^{\mathcal{D}}_{s,a} - V(s;\phi) \tag{17}$$

Also, we can use Generalized Advantage Estimation (Schulman et al., 2016), where $\gamma \in [0,1]$ and $\delta_t^{\phi_k}$ is a one-step temporal difference for state s_t calculated using old vector ϕ_k:

$$\begin{aligned} \delta_t^{\phi_k} &= r_t + \gamma V(s_{t+1};\phi_k) - V(s_t;\phi_k) \\ \hat{A}(s_t,a_t;\phi_k) &= \sum_{l=0}^{T}(\gamma\lambda)^l\delta_{t+l}^{\phi_k} \\ A(s_t,a_t;\phi) &= \hat{A}(s_t,a_t;\phi_k) + V(s_t;\phi_k) - V(s_t;\phi) \end{aligned} \tag{18}$$

Finally, we propose the following reinforcement learning algorithm:

Algorithm 1: Hellinger Distance Constrained Regression

$\theta_1 \leftarrow$ random initial weights
$\mathcal{D} \leftarrow \emptyset$
for *iteration* $k = 1, ..., k_{max}$ **do**
  add trajectories $\{\tau_i\}$ sampled via π_{θ_k} to $\mathcal{D}$
  $\phi_{k+1} \leftarrow \arg\min_\phi \mathbb{E}_{s,a\sim\mathcal{D}} A^2(s,a;\phi)$
  $\theta_{k+1} \leftarrow \arg\max_\theta \mathbb{E}_{s,a\sim\mathcal{D}}\left[\log \pi_\theta(a|s)\frac{1}{(\beta - A(s,a;\phi_k))^2}\right]$
end

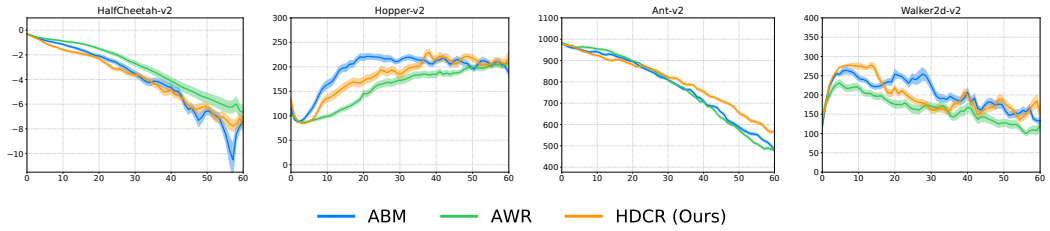

Figure 1: Learning curves of ABM, AWR, and HDCR averaged across results of learning from 10 different datasets of 10k timestamps (also 5 seeds used to generate each dataset).

Task	ABM	AWR	HDCR
Ant-v2	468 ± 72	495 ± 84	$\mathbf{569 \pm 64}$
HalfCheetah-v2	$\mathbf{-7 \pm 3}$	$\mathbf{-7 \pm 4}$	-8 ± 6
Hopper-v2	184 ± 72	209 ± 31	$\mathbf{223 \pm 141}$
Walker2d-v2	132 ± 76	125 ± 92	$\mathbf{145 \pm 117}$

Table 1: Final returns for different algorithms, with $\pm$ corresponding to one standard deviation of the average return across 10 datasets of 10k timestamps.

5 EXPERIMENTS

In our experiments, we evaluate the algorithm on MuJoCo (Todorov et al., 2012) tasks. We use setting inspired by Behavioral Modelling Priors for Offline Reinforcement Learning paper (Siegel et al., 2020). We have the following differences: on a tiny dataset, we use random actions sampled uniformly (random agent).

First, we collect 2048 timestamps or more, until episode termination (whichever occurred later), from each of 5 seeds using random actions. Then we load collected trajectories to a replay buffer for the agent training. Separate networks with the same architecture (except the last layer) represent the policy and value function and consist of 2 hidden layers of 256 ELU units. Each train iteration uses only old data obtained by random agents. Each iteration, the value function is updating with 5 gradient steps and policy with 50 steps using a uniformly sampled batches of 512 samples using all data from the replay buffer. Learning rates for Adam optimizer are 2×10^{-3} and 2×10^{-4} for critic and actor, respectively.

We compare 3 different "advantage-weight" functions $f(A(s, a))$:

- Hellinger Distance Constrained Regression, where $f(A(s, a)) = \frac{1}{(\beta - A(s,a))^2}$;
- Advantage-weighted Behavior Model, where $f(A(s, a)) = I_{A(s,a)>0}$, $I_{x>0} = 1$ if $x > 0$ otherwise $I_{x>0} = 0$;
- Advantage-Weighted Regression, where $f(A(s, a)) = \exp(\frac{1}{\beta} A(s, a))$.

AWR method uses $\beta = 1.0$ as it is in implementation released by authors. HDCR uses $\beta = 1.0$. For TD(λ) we use $\lambda = 0.95$.

On simple tasks as Hopper-v2, all methods are able to learn (Figure 1), and HDCR shows slightly better results (Table 1). While on difficult tasks as Ant-v2, all algorithms do not improve their results through iterations. Moreover, evaluation returns decrease.

Next, we perform training with the same setting on 16k first timestamps obtained from a single on-policy training by HDCR. Figure 2 shows that agents can learn on 2 out of 4 environments. On 3 out of 4 environments, HDCR shows (Table 2) better results than ABM and AWR.

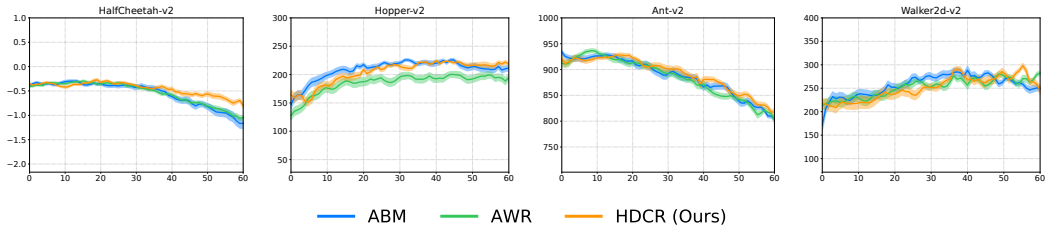

Figure 2: Learning curves of ABM, AWR, and HDCR averaged across results of learning from 10 different datasets each of 16k timestamps from single training of HDCR.

Task	ABM	AWR	HDCR
Ant-v2	805 ± 36	796 ± 45	$\mathbf{817 \pm 47}$
HalfCheetah-v2	$\mathbf{-1 \pm 1}$	$\mathbf{-1 \pm 0}$	$\mathbf{-1 \pm 1}$
Hopper-v2	212 ± 40	197 ± 48	$\mathbf{217 \pm 30}$
Walker2d-v2	250 ± 52	$\mathbf{284 \pm 44}$	242 ± 98

Table 2: Final returns for different algorithms, with $\pm$ corresponding to one standard deviation of the average return across 10 datasets of 16k timestamps.

6 Discussion

We theoretically proved that Hellinger distance improves the lower bound of expected discounted return compared to Kullback-Leibler divergence and proposed a simple off-policy reinforcement learning method that uses Hellinger distance as a constraint. The expected discounted return for a new policy now has a policy-independent lower bound. This bound guarantees that return will not decrease in "one" shot.

Experiments show that HDCR outperforms both ABM and AWR on tiny datasets obtained by random agents. This performance proves the efficiency of using Hellinger distance by allowing bigger step sizes, retaining lower bound.

On bigger datasets, HDCR shows comparable or better results compared to AWR and ABM training from offline data collected by HDCR on first on-policy train iterations.

References

Sham Kakade and John Langford. Approximately optimal approximate reinforcement learning. In *Proceedings of the Nineteenth International Conference on Machine Learning*, ICML '02, pp. 267–274, San Francisco, CA, USA, 2002. Morgan Kaufmann Publishers Inc. ISBN 1558608737.

David Levin and Yuval Peres. *Markov Chains and Mixing Times*. 2017. doi: 10.1090/mbk/107.

Xue Bin Peng, Aviral Kumar, Grace Zhang, and Sergey Levine. Advantage-weighted regression: Simple and scalable off-policy reinforcement learning, 2019.

John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. Trust region policy optimization. volume 37 of *Proceedings of Machine Learning Research*, pp. 1889–1897, Lille, France, 07–09 Jul 2015. PMLR. URL `http://proceedings.mlr.press/v37/schulman15.html`.

John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2016.

Noah Siegel, Jost Tobias Springenberg, Felix Berkenkamp, Abbas Abdolmaleki, Michael Neunert, Thomas Lampe, Roland Hafner, Nicolas Heess, and Martin Riedmiller. Keep doing what

worked: Behavior modelling priors for offline reinforcement learning. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=rke7geHtwH.

E. Todorov, T. Erez, and Y. Tassa. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5026–5033, 2012.

Alexandre B. Tsybakov. *Introduction to Nonparametric Estimation*. Springer New York, 2009. doi: 10.1007/b13794. URL https://doi.org/10.1007%2Fb13794.

A THEOREM 1 PROOF

Let $(\Omega,\ \mathcal{A})$ be a measurable space, P and Q be two probability measures on that space, v be a σ-finite measure on $(\Omega,\ \mathcal{A})$ such that $P \ll v$ ($P(A) = 0$ for any $A \in \mathcal{A}$ such that $\mu(A) = 0$) and $Q \ll v$. And let us denote Radon–Nikodym derivatives (in our derivations we can assume them as probability density functions) as follows:

$$\begin{aligned} p &= \frac{dP}{dv} \\ q &= \frac{dQ}{dv} \end{aligned} \tag{19}$$

So, p and q satisfy following conditions:

$$\begin{aligned} P(A) &= \int_A p(v)\ dv\ \ \forall A \in \mathcal{A} \\ Q(A) &= \int_A q(v)\ dv\ \ \forall A \in \mathcal{A} \end{aligned} \tag{20}$$

Then we define distances between probability measures:

$$\begin{aligned} D_{TV}(P\ ||\ Q) &= \sup_{A \in \mathcal{A}} |P(A) - Q(A)| \\ H(P\ ||\ Q)^2 &= \frac{1}{2}\int (\sqrt{p(v)} - \sqrt{q(v)})^2\ dv = 1 - \int \sqrt{p(v)q(v)}\ dv \\ D_{KL}(P\ ||\ Q) &= \int \log \frac{dP}{dQ}\ dP = \int p(v) \log \frac{p(v)}{q(v)}\ dv \end{aligned} \tag{21}$$

Following Lemmas will be used in Theorem 1 proof:

Lemma 1. *Given two probability distributions p and q total variance divergence can be calculated as follows:*

$$D_{TV}(P\ ||\ Q) = \sup_{A \in \mathcal{A}} |P(A) - Q(A)| = \frac{1}{2}\int |p(v) - q(v)|\ dv \tag{22}$$

Proof. Let $B = \{v : P(v) > Q(v)\}$, $B^c = \{v : P(v) \leq Q(v)\}$ and $A \in \mathcal{A}$:

$$\begin{aligned} P(A) - Q(A) &\leq P(A \cap B) - Q(A \cap B) \leq P(B) - Q(B) \\ P(A) - Q(A) &\leq Q(A \cap B^c) - P(A \cap B^c) \leq Q(B^c) - P(B^c) \\ \sup_{A \in \mathcal{A}} |P(A) - Q(A)| &= P(B) - Q(B) = Q(B^c) - P(B^c) \\ D_{TV}(P\ ||\ Q) &= \frac{1}{2}[P(B) - Q(B) + Q(B^c) - P(B^c)] \\ &= \frac{1}{2}\int |p(v) - q(v)|\ dv \end{aligned} \tag{23}$$

□

Lemma 2.

$$D_{TV}(P\ ||\ Q) = 1 - \int \min(p(v), q(v))\ dv \tag{24}$$

Proof.

$$\begin{aligned} D_{TV}(P\ ||\ Q) &= \frac{1}{2}\int |p(v) - q(v)|\ dv = \int_{\{v:\ p(v) > q(v)\}} [p(v) - q(v)]\ dv \\ &= 1 - \int_{\{v:\ p(v) < q(v)\}} p(v)\ dv - \int_{\{v:\ p(v) > q(v)\}} q(v)\ dv \\ &= 1 - \int \min(p(v), q(v))\ dv \end{aligned} \tag{25}$$

□

Lemma 3.

$$\int \max(p(v), q(v))\, dv + \int \min(p(v), q(v))\, dv = 2 \tag{26}$$

Proof. Rewriting left part in 4 integrals over following sets $\{\max(p(v), q(v)) = p(v)\}$, $\{\max(p(v), q(v)) = q(v)\}$, $\{\min(p(v), q(v)) = q(v)\}$ and $\{\min(p(v), q(v)) = q(v)\}$ and stacking integrals back gives us $P(\Omega) + Q(\Omega) = 2$. □

Theorem 1. *For any two probability density functions p and q, the following double inequality is true:*

$$D_{TV}(p \,||\, q)^2 \leq 2H(p \,||\, q)^2 \leq D_{KL}(p \,||\, q) \tag{27}$$

Proof. First inequality can be proved as follows:

$$\begin{aligned}
(1 - H(P \,||\, Q)^2)^2 &= \left[\int \sqrt{p(v)q(v)}\, dv\right]^2 \\
&= \left[\int \sqrt{\min(p(v), q(v)) \max(p(v), q(v))}\, dv\right]^2 \\
&\leq \int \min(p(v), q(v))\, dv \int \max(p(v), q(v))\, dv \\
&= \int \min(p(v), q(v))\, dv \left[2 - \int \min(p(v), q(v))\, dv\right] \\
&= (1 - D_{TV}(P \,||\, Q))(1 + D_{TV}(P \,||\, Q)) \\
&= 1 - D_{TV}(P \,||\, Q)^2 \\
D_{TV}(P \,||\, Q)^2 &\leq H(P \,||\, Q)^2(2 - H(P \,||\, Q)^2) \\
D_{TV}(P \,||\, Q)^2 &\leq 2H(P \,||\, Q)^2
\end{aligned} \tag{28}$$

Proof of the second inequality:

$$\begin{aligned}
D_{KL}(P \,||\, Q) &= \int p(v) \log \frac{p(v)}{q(v)}\, dv = 2 \int p(v) \log \sqrt{\frac{p(v)}{q(v)}}\, dv \\
&= -2 \int p(v) \log \left(\left[\sqrt{\frac{q(v)}{p(v)}} - 1\right] + 1\right) dv \\
&\geq -2 \int p(v) \left[\sqrt{\frac{q(v)}{p(v)}} - 1\right]\, dv = -2 \int \left[\sqrt{q(v)p(v)} - 1\right]\, dv \\
&= 2\left[1 - \int \sqrt{p(v)q(v)}\, dv\right] = 2H(P \,||\, Q)^2 \\
D_{KL}(P \,||\, Q) &\geq 2H(P \,||\, Q)^2
\end{aligned} \tag{29}$$

□

B HDCR DERIVATION

Given optimization problem:

$$\begin{aligned}
&\arg\max_{\pi} \int \rho_\mu(s) \int \pi(a|s) A_\mu(s, a)\, da\, ds \\
&\text{s.t.} \int \rho_\mu(s) H(\pi(\cdot|s) || \mu(\cdot|s))\, ds \leq \epsilon, \\
&\int \pi(a|s)\, da = 1,\ \forall s \in \mathcal{S}.
\end{aligned} \tag{30}$$

Constructing Lagrangian where $\alpha : \mathcal{S} \rightarrow \mathbb{R}$ is a function for obtaining Lagrange multiplier for every state, β is also a Lagrange multiplier:

$$\begin{aligned}\mathcal{L}(\pi, \beta, \alpha) = &\int_s \rho_\mu(s) \int_a \pi(a|s) A_\mu(s,a)\, da\, ds \\ &+ \beta \left(\epsilon - \int_s \rho_\mu(s) \left[1 - \int \sqrt{\pi(a|s)\mu(a|s)}\, da \right] ds \right) \\ &+ \int_s \alpha_s \left(1 - \int_a \pi(a|s)\, da \right) ds\end{aligned} \tag{31}$$

Differentiating with respect to $\pi(a|s)$ gives us following result:

$$\frac{\partial \mathcal{L}}{\partial \pi(a|s)} = \rho_\mu(s) A_\mu(s,a) + \beta \rho_\mu(s) \frac{\mu(a|s)}{2\sqrt{\pi(a|s)\mu(a|s)}} - \alpha_s = 0 \tag{32}$$

Solving for $\pi(a|s)$:

$$\begin{aligned}\beta \rho_\mu(s) \frac{\mu(a|s)}{2\sqrt{\pi(a|s)\mu(a|s)}} &= \alpha_s - \rho_\mu(s) A_\mu(s,a) \\ \frac{\sqrt{\mu(a|s)}}{\sqrt{\pi(a|s)}} &= 2\frac{\frac{\alpha_s}{\rho_\mu(s)} - A_\mu(s,a)}{\beta}\end{aligned} \tag{33}$$

Substituting $\pi(a|s) = \mu(\cdot|s)$ and taking expectation over actions taken according to μ gives us expression for α_s:

$$\begin{aligned}1 &= 2\frac{\frac{\alpha_s}{\rho_\mu(s)} - A_\mu(s,a)}{\beta} \\ \mathbb{E}_{a\sim\mu(\cdot|s)} \alpha_s = \alpha_s &= \rho_\mu(s) \mathbb{E}_{a\sim\mu(\cdot|s)} \left[\frac{1}{2}\beta + A_\mu(s,a) \right] = \frac{1}{2}\beta\rho_\mu(s)\end{aligned} \tag{34}$$

Then optimal policy $\pi^*(a|s)$ can be written as follows:

$$\pi^*(a|s) = \mu(a|s) \frac{\beta^2}{(\beta - 2A_\mu(s,a))^2} \tag{35}$$

To obtain regression problem we construct Kullback–Leibler divergence between optimal policy π^* and current policy π:

$$\begin{aligned}&\arg\min_\pi \mathbb{E}_{s\sim\rho_\mu(\cdot)} D_{KL}(\pi^*(\cdot|s) \,||\, \pi(\cdot|s)) \\ &= \arg\min_\pi \mathbb{E}_{s\sim\rho_\mu(\cdot)} \int \pi^*(a|s) \log \frac{\pi^*(a|s)}{\pi(a|s)}\, da \\ &= \arg\min_\pi \mathbb{E}_{s\sim\rho_\mu(\cdot)} \int \mu(a|s) \frac{\beta^2}{(\beta - 2A_\mu(s,a))^2} \log \frac{\mu(a|s)\frac{\beta^2}{(\beta-2A_\mu(s,a))^2}}{\pi(a|s)}\, da \\ &= \arg\min_\pi \mathbb{E}_{s\sim\rho_\mu(\cdot)} \mathbb{E}_{a\sim\mu(\cdot|s)} \frac{1}{(\beta - 2A_\mu(s,a))^2} \left[\log \left(\mu(a|s) \frac{\beta^2}{(\beta - 2A_\mu(s,a))^2} \right) - \log \pi(a|s) \right] \\ &= \arg\max_\pi \mathbb{E}_{s\sim\rho_\mu(\cdot)} \mathbb{E}_{a\sim\mu(\cdot|s)} \log \pi(a|s) \frac{1}{(\beta - 2A_\mu(s,a))^2} \\ &= \arg\max_\pi \mathbb{E}_{s\sim\rho_\mu(\cdot)} \mathbb{E}_{a\sim\mu(\cdot|s)} \log \pi(a|s) \frac{1}{(\beta - A_\mu(s,a))^2}\end{aligned} \tag{36}$$

Regression problem follows:

$$\arg\max_\pi \mathbb{E}_{s\sim\rho_\mu(\cdot)} \mathbb{E}_{a\sim\mu(\cdot|s)} \log \pi(a|s) \frac{1}{(\beta - A_\mu(s,a))^2} \tag{37}$$