The Importance of Pessimism in Fixed-Dataset Policy Optimization

Jacob Buckman (Mila; McGill University)*
Carles Gelada (OpenAI)
Marc G. Bellemare (Google Research; Mila; McGill University; CIFAR Fellow)

This work **provides formal justification** for the properties of **every "Offline RL" algorithm** in the literature, including:

BCQ, CRR, SPIBB, BEAR, CQL, KLC, BRAC, MBS-QI, MoREL, MOPO, and more.

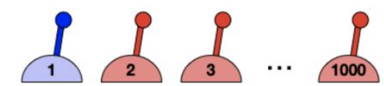
Our goal is an algo with minimal worst-case suboptimality,

$$\textsc{SubOpt}(\mathcal{O}(D)) = \mathbb{E}_\rho[\mathbf{v}^{\pi^*_\mathcal{M}}_\mathcal{M}] - \mathbb{E}_\rho[\mathbf{v}^{\mathcal{O}(D)}_\mathcal{M}].$$

We consider "value-based" algorithms,

$$\mathcal{O}^{\text{VB}}_{\text{sub}}(D) := \arg\max_\pi \mathbb{E}_\rho[\mathcal{E}_{\text{sub}}(D, \pi)].$$

These algorithms can be characterized by the choice of fixed point of E_{sub}.
Suboptimality of these algos permits an "over/under decomposition",



μ	99%	1%	1%		1%
n_D	10000	1	1		1
μ_D	98%	0%	100%		0%

Theorem 1. *For any space $\mathcal{X}$, objective $f : \mathcal{X} \to \mathbb{R}$, and proxy objective $\hat{f} : \mathcal{X} \to \mathbb{R}$,*

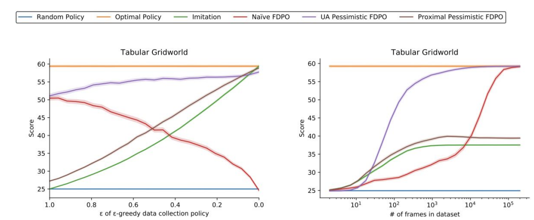
$$f(x^*) - f(\hat{x}^*) \leq \inf_{x \in \mathcal{X}} \Big([f(x^*) - f(x)] + [f(x) - \hat{f}(x)] \Big) + \sup_{x \in \mathcal{X}} \Big(\hat{f}(x) - f(x) \Big)$$

where $x^* := \arg\max_{x \in \mathcal{X}} f(x)$ and $\hat{x}^* := \arg\max_{x \in \mathcal{X}} \hat{f}(x)$. Furthermore, this bound is tight.

One important type of algo is "naive":

$$f_{\textit{naïve}}(\mathbf{v}^\pi) := A^\pi(\mathbf{r}_D + \gamma P_D \mathbf{v}^\pi).$$

$$\textit{SubOpt}(\mathcal{O}^{VB}_{\textit{naïve}}(D)) \leq \inf_\pi \Big(\mathbb{E}_\rho[\mathbf{v}^{\pi^*_\mathcal{M}}_\mathcal{M} - \mathbf{v}^\pi_\mathcal{M}] + \mathbb{E}_\rho[\boldsymbol{\mu}^\pi_{D,\delta}] \Big) + \sup_\pi \mathbb{E}_\rho[\boldsymbol{\mu}^\pi_{D,\delta}]$$

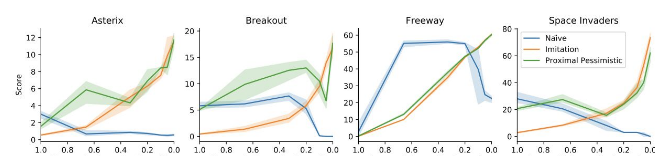


This often leads to a large "sup" term. We can fix this by finding pessimistic fixed points, which let us choose the relative size of the two terms:

$$\textit{SubOpt}(\underline{\mathcal{O}}^{VB}_{ua}(D)) \leq \inf \Big(\mathbb{E}_\rho[\mathbf{v}^{\pi^*_\mathcal{M}}_\mathcal{M} - \mathbf{v}^\pi_\mathcal{M}] + (1+\alpha) \cdot \mathbb{E}_\rho[\boldsymbol{\mu}^\pi_{D,\delta}] \Big) + (1-\alpha) \cdot \Big(\sup_\pi \mathbb{E}_\rho[\boldsymbol{\mu}^\pi_{D,\delta}] \Big)$$

$$f_{ua}(\mathbf{v}^\pi) = A^\pi(\mathbf{r}_D + \gamma P_D \mathbf{v}^\pi) - \alpha \mathbf{u}^\pi_{D,\delta}$$

Implementing this algorithm requires implementing a valid uncertainty measure, which we don't know how to do right now with NNs. If we take "trivial uncertainty" of V_{max}, we get proximal algorithms:



$$f_{\textit{proximal}}(\mathbf{v}^\pi) = A^\pi(\mathbf{r}_D + \gamma P_D \mathbf{v}^\pi) - \alpha \left(\frac{TV_\mathcal{S}(\pi, \hat{\pi}_D)}{(1-\gamma)^2} \right)$$

The trivial uncertainty is the "worst" uncertainty, leading to a much looser bound; but it is, at least, implementable.