A. Proofs

A.1. Proofs for Section 3

Proof of Theorem 3. The proof is similar to that of Corollary 1 in (Mathé, 2006). Define $\tilde{i} = \max\{i : \textsc{b}(i) \leq \textsc{cnf}(i)\}$. The proof is composed of two steps: first we show that we are competitive with $\tilde{i}$ and then we show that $\tilde{i}$ is competitive with the best index.

Competing with $\tilde{i}$. Observe that since $\textsc{b}$ is monotonically increasing, and $\textsc{cnf}$ is monotonically decreasing, for $i \leq \tilde{i}$ we have

$$\textsc{b}(i) \leq \textsc{b}(\tilde{i}) \leq \textsc{cnf}(\tilde{i}) \leq \textsc{cnf}(i).$$

Therefore, for $i \leq \tilde{i}$

$$\left|\hat{\theta}_i - \theta^\star\right| \leq \textsc{b}(i) + \textsc{cnf}(i) \leq 2\textsc{cnf}(i).$$

This implies that $\theta^\star \in I_i$ for all $i \leq \tilde{i}$.

As a consequence, the definition of our chosen index $\hat{i}$ implies that $\hat{i} \geq \tilde{i}$, which in turn implies that $I_{\tilde{i}} \cap I_{\hat{i}} \neq \emptyset$. So, there exists $x \in I_{\tilde{i}} \cap I_{\hat{i}}$ such that $|x - \hat{\theta}_{\tilde{i}}| \leq 2\textsc{cnf}(\tilde{i})$ and $|x - \hat{\theta}_{\hat{i}}| \leq 2\textsc{cnf}(\hat{i})$. As we know that $\theta^\star \in I_{\tilde{i}}$, we get

$$|\hat{\theta}_{\hat{i}} - \theta^\star| \leq |\hat{\theta}_{\hat{i}} - x| + |x - \hat{\theta}_{\tilde{i}}| + |\hat{\theta}_{\tilde{i}} - \theta^\star| \leq 2\textsc{cnf}(\hat{i}) + 2\textsc{cnf}(\tilde{i}) + 2\textsc{cnf}(\tilde{i}) \leq 6\textsc{cnf}(\tilde{i}). \tag{5}$$

Comparing $\tilde{i}$ to $i^\star$. Define $i^\star := \operatorname{argmin}_i\{\textsc{b}(i) + \textsc{cnf}(i)\}$ which is the index we actually want to compete with in our guarantee. If we compare $\tilde{i}$ with $i^\star$, then by the above argument we can translate to $\hat{i}$. For this, we consider two cases:

If $i^\star \leq \tilde{i}$, then by definition of $\tilde{i}$, we have

$$\textsc{b}(\tilde{i}) + \textsc{cnf}(\tilde{i}) \leq 2\textsc{cnf}(\tilde{i}) \leq 2\textsc{cnf}(i^\star) \leq 2(\textsc{cnf}(i^\star) + \textsc{b}(i^\star)),$$

so we are a factor of 2 worse.

On the other hand, if $i^\star > \tilde{i}$ then by Assumption 2 and the optimality condition for $\tilde{i}$

$$\kappa\textsc{cnf}(\tilde{i}) \leq \textsc{cnf}(\tilde{i}+1) \leq \textsc{b}(\tilde{i}+1) \leq \textsc{b}(i^\star).$$

This implies

$$\textsc{b}(\tilde{i}) + \textsc{cnf}(\tilde{i}) \leq (1 + 1/\kappa)\textsc{b}(i^\star).$$

As $\kappa \leq 1$, this bound dominates the previous case, and together, with (5) we have

$$\left|\hat{\theta}_{\hat{i}} - \theta^\star\right| \leq 6 \times \textsc{cnf}(\tilde{i}) \leq 6(1 + 1/\kappa) \min_{i \in [M]} \left\{\textsc{b}(i) + \textsc{cnf}(i)\right\}. \qquad \square$$

Proof of Corollary 4. For the MSE calculation, we simply need to translate from the high probability guarantee to the MSE, which is not difficult under Assumption 1. In particular, fix δ and let $\mathcal{E}$ be the event that all confidence bounds are valid, which holds with probability $1 - \delta$, then we have

$$\begin{aligned}
\mathbb{E}(\hat{\theta}_{\hat{i}} - \theta^\star)^2 &= \mathbb{E}(\hat{\theta}_{\hat{i}} - \theta^\star)^2 \mathbf{1}\left\{\mathcal{E}\right\} + \mathbb{E}(\hat{\theta}_{\hat{i}} - \theta^\star)^2 \mathbf{1}\left\{\bar{\mathcal{E}}\right\} \\
&\leq \mathbb{E}(\hat{\theta}_{\hat{i}} - \theta^\star)^2 \mathbf{1}\left\{\mathcal{E}\right\} + R^2\delta \\
&\leq \mathbb{E}\mathbf{1}\left\{\mathcal{E}\right\} \left(6(1 + 1/\kappa)\mathbb{E} \min_{i \in [M]} \left\{\textsc{b}(i) + \textsc{cnf}(i;\delta)\right\}\right)^2 + R^2\delta \\
&\leq 72(1 + 1/\kappa)^2 \min_{i \in [M]} \left\{\textsc{b}(i)^2 + \textsc{cnf}(i;\delta)^2\right\} + R^2\delta.
\end{aligned}$$

Here in the first line we are introducing the event $\mathcal{E}$ and its complement. In the second, we use that $\hat{\theta}_i, \theta^\star \in [0, R]$ almost surely and that $\mathbb{P}[\bar{\mathcal{E}}] \leq \delta$ according to Assumption 1. In the third line, we apply Theorem 3, which holds under event $\mathcal{E}$. The final step uses the simplification that $(a + b)^2 \leq 2a^2 + 2b^2$. $\square$

A.2. Proofs for Section 4

Proof of Theorem 5. Let us first verify that the confidence function specified in (2) satisfy Assumption 1. We will apply Bernstein's inequality, which requires variance and range bounds. For the variance, a single sample satisfies

$$\mathrm{Var}\left(\frac{K(|\pi_{\mathrm{T}}(x)-a|/h)r(a)}{h\times\pi_{\mathrm{L}}(a\mid x)}\right)\le\frac{1}{h}\mathbb{E}\left[\frac{K(|\pi_{\mathrm{T}}(x)-a|/h)^2}{h\times\pi_{\mathrm{L}}^2(a\mid x)}\mid a\sim\pi_{\mathrm{L}}(\cdot\mid x)\right]\le\frac{1}{2h},$$

where we first use that the variance is upper bounded by the second moment, and then we use that π_{L} is uniform and the boxcar kernel is at most $1/2$. Finally, we use that by a change of variables $K(\cdot/h)/h$ integrates to 1. Note that we are using that $\pi_{\mathrm{T}}\in[\gamma_0,1-\gamma_0]$, as we are integrating over the support of π_{L}.

For the range, we have

$$\sup\frac{K(|\pi_{\mathrm{T}}(x)-a|/h)r(a)}{h\times\pi_{\mathrm{L}}(a\mid x)}\le\frac{1}{2h}.$$

Therefore, Bernstein's inequality gives that with probability $1-\delta$ we have

$$\left|\hat{V}_h(\pi_{\mathrm{T}})-\mathbb{E}\hat{V}_h(\pi_{\mathrm{T}})\right|\le\sqrt{\frac{\log(2/\delta)}{nh}}+\frac{\log(2/\delta)}{3nh},$$

and the first claim follows by a union bound.

Monotonicity is also easy to verify with this definition of CNF. In particular, since $h_i=\gamma h_{i+1}$ and $\gamma<1$, we immediately have that

$$\gamma\textsc{cnf}(i)=\gamma\sqrt{\frac{\log(2M/\delta)}{nh_i}}+\gamma\frac{\log(2M/\delta)}{3nh_i}=\gamma\sqrt{\frac{\log(2M/\delta)}{n\gamma h_{i+1}}}+\frac{\log(2M/\delta)}{3nh_{i+1}}\le\textsc{cnf}(i+1)$$

Clearly $\textsc{cnf}(i+1)\le\textsc{cnf}(i)$, and so Assumption 2 holds. This verifies that we may apply Theorem 3.

For the last claim, if the rewards are L-Lipschitz, then we claim we can set $\textsc{b}(i)=Lh_i$. To see why, observe that

$$\begin{aligned}|\mathbb{E}V_h(\pi_{\mathrm{T}})-V(\pi_{\mathrm{T}})|&=\left|\mathbb{E}_{(x,r),a\sim\pi_{\mathrm{L}}(x)}\frac{K(|\pi_{\mathrm{T}}(x)-a|/h)r(a)}{h}-r(\pi_{\mathrm{T}}(x))\right|\\&=\left|\mathbb{E}_{(x,r)}\int_{a'}\frac{\mathbf{1}\left\{|\pi_{\mathrm{T}}(x)-a|\le h\right\}r(a)}{2h}-r(\pi_{\mathrm{T}}(x))\right|\\&=\left|\mathbb{E}_{(x,r)}\int_{a'}\frac{\mathbf{1}\left\{|\pi_{\mathrm{T}}(x)-a|\le h\right\}(r(a)-r(\pi_{\mathrm{T}}(x)))}{2h}\right|\le Lh.\end{aligned}$$

Clearly this bias bound is monotonic. To apply Theorem 3, it is better to first simplify the confidence function. Observe that as $\theta^\star,\bar{\theta}_i\in[0,1]$, it is always better to clip the estimates $\hat{\theta}_i$ to lie in $[0,1]$. This has no bearing on the bias and only improves the deviation term, and in particular allows us to replace $\textsc{cnf}(i)$ with $\min\{\textsc{cnf}(i),1\}$. This leads to a further simplification:

$$\sqrt{\frac{\log(2M/\delta)}{nh_i}}+\frac{\log(2M/\delta)}{3nh_i}\le1\Rightarrow\sqrt{\frac{\log(2M/\delta)}{nh_i}}+\frac{\log(2M/\delta)}{3nh_i}\le\frac{4}{3}\sqrt{\frac{\log(2M/\delta)}{nh_i}}$$

Therefore we may replace CNF with this latter function and by Theorem 3 we guarantee that with probability at least $1-\delta$

$$\left|\hat{\theta}_i-\theta^\star\right|\le6(1+\gamma^{-1})\min_i\left\{Lh_i+\frac{4}{3}\sqrt{\frac{\log(2M/\delta)}{nh_i}}\right\}.$$

The optimal choice for h is $h^\star:=\left(\frac{4}{3L}\sqrt{\frac{\log(2M/\delta)}{n}}\right)^{2/3}$, which will in general not be in our set $\mathcal{H}$. However, if we use this choice for h, the error rate is $O((L/n)^{1/3})$, and since we know that $\theta^\star\in[0,1]$, if $L>n$ then this error guarantee is trivial.

In other words, the maximum value of L that we are interested in adapting to is $L_{\max} = n$. This will be useful in setting the number of models to search over M.

To set M, we want to ensure that there exists some h_i such that $h_i \le h^\star \le h_{i+1}$. We first verify the first inequality, which requires that

$$\gamma_0 \gamma^M \le h^\star := \left(\frac{4}{3L} \sqrt{\frac{\log(2M/\delta)}{n}} \right)^{2/3}$$

We will always take $M \ge 2$, which implies that $\log(2M/\delta) \ge 1$. Then, since we are only interested in $L \le n$, a sufficient condition here is

$$\gamma_0 \gamma^M \le \frac{4}{3}^{2/3} n^{-1} \Rightarrow M \ge \frac{C_{\gamma_0} \log(n)}{\log(1/\gamma_0)},$$

where C_{γ_0} is a constant that only depends on γ_0. The upper bound $h^\star \le h_{i+1}$ is satisfied as soon as n is large enough, provided that $L \ge \omega(\sqrt{\log(\log(n))/n})$, which we are assuming. Thus we know that there is $i^\star$ such that $h_{i^\star} \le h^\star \le h_{i^\star}/\gamma$, and using this choice, we have

$$\begin{aligned}
\left| \hat{\theta}_{\hat{i}} - \theta^\star \right| &\le 6(1+\gamma^{-1}) \left\{ L h_{i^\star} + \frac{4}{3} \sqrt{\frac{\log(2M/\delta)}{n h_{i^\star}}} \right\} \le 6(1+\gamma^{-1}) \left\{ L h^\star + \frac{4}{3} \sqrt{\frac{\log(2M/\delta)}{\gamma n h^\star}} \right\} \\
&\le 6(1+\gamma^{-1}) \cdot \frac{c_1}{\sqrt{\gamma}} (L \log(2M/\delta)/n)^{1/3} \le C_{\gamma,\gamma_0} (L \log(\log(n)/\delta)/n)^{1/3},
\end{aligned}$$

where C_{γ,γ_0} is a constant that depends only on γ, γ_0. □

Note that if π_{L} is non-uniform, but satisfies $\inf_{x,a} \pi_{\mathrm{L}}(a \mid x) \ge p_{\min}$, then very similar arguments apply. In particular, we have that both variance and range are bounded by $\frac{1}{2hp_{\min}}$, and some Bernstein's inequality in this case yields

$$\left| \hat{V}_h(\pi_{\mathrm{T}}) - \mathbb{E}\hat{V}_h(\pi_{\mathrm{T}}) \right| \le \sqrt{\frac{\log(2/\delta)}{nhp_{\min}}} + \frac{\log(2/\delta)}{3nhp_{\min}},$$

Monotonicity follows from the same calculation as before and the clipping trick yields a more interpretable final bound, which holds with probability at least $1-\delta$, of

$$\left| \hat{\theta}_{\hat{i}} - \theta^\star \right| \le 6(1+\gamma^{-1}) \min_i \left\{ L h_i + \frac{4}{3} \sqrt{\frac{\log(2M/\delta)}{n h_i p_{\min}}} \right\}.$$

The remaining calculation for M is analogous, since this bound is identical to the previous one with n replaced by $np_{\min}$. Thus, we obtain a final bound of $C_{\gamma,\gamma_0}(L \log(\log(n/\delta))/(np_{\min}))^{1/3}$.

A.3. Proofs for Section 5

We first develop and state the more precise version of Theorem 6. We introduce the doubly robust version of the partial importance weighting estimator. As it is the empirical average over n trajectories, here we will focus on a single trajectory $(x_1, a_1, r_1, \ldots, x_H, a_H, r_H)$ sampled by following the logging policy π_{L}.

Define $\hat{V}^0_{\mathrm{DR}} := 0$ and

$$\hat{V}^{H+1-h}_{\mathrm{DR}} := \hat{V}(x_h) + p_h(r_h + \gamma \hat{V}^{H-t}_{\mathrm{DR}} - \hat{Q}(x_h, a_h)), \qquad p_h := \frac{\pi_{\mathrm{T}}(a_h \mid x_h)}{\pi_{\mathrm{L}}(a_h \mid x_h)}.$$

where $\hat{Q}$ is the direct model, trained via supervised learning, and $\hat{V}(x) = \mathbb{E}_{a \sim \pi_{\mathrm{T}}(x)} \hat{Q}(x,a)$. The full horizon doubly-robust estimator is $\hat{V}_{\mathrm{DR}} := \hat{V}^H_{\mathrm{DR}}$. To define the η-step partial estimator, let $\hat{V}^\eta_{\mathrm{DM}} := \rho_\eta \hat{Q}(x_{\eta+1}, \pi_{\mathrm{T}}(x_{\eta+1}))$, which is an estimate

of $\mathbb{E}_{\pi_{\mathrm{T}}}[V(x_{\eta+1})]$. Set $\hat{V}_{\mathrm{DM}}^{H} := 0$. Then for a false horizon η, we define a similar recursion

$$\hat{V}_{\eta}^{H+1-h} := \begin{cases} \hat{V}(x_h) + p_h(r_h + \gamma V_{\eta}^{H-h} - \hat{Q}(x_h, a_h)) \text{ if } 1 \le h < \eta \\ \hat{V}(x_h) + p_h(r_h + \gamma \hat{V}_{\mathrm{DM}}^{\eta} - \hat{Q}(x_h, a_h)) \text{ if } h = \eta. \end{cases}$$

The doubly robust variant of the η-step partial importance weighted estimator is $\hat{V}_{\eta}^{H}$. We also define $\hat{V}_{0}^{H} = \hat{V}_{\mathrm{DM}}^{0}$ which estimates $\mathbb{E}[V(x_1)]$. Observe that if in the definition of $\hat{V}_{\mathrm{DR}}$, we take $\hat{V}, \hat{Q} \equiv 0$ then we obtain the estimator in (4).

Define $\Delta := \log(2(H+1)/\delta)$, $V_{\max} := (1-\gamma)^{-1}$ and recall that $p_{\max} := \max_{x,a} \frac{\pi_{\mathrm{T}}(a|x)}{\pi_{\mathrm{L}}(a|x)}$. Then define

$$\textsc{b}(i) := \frac{\gamma^{H-i+1} - 1}{1-\gamma}$$

$$\textsc{cnf}(i) := \sqrt{\frac{6V_{\max}^2 \left(1 + \sum_{h=1}^{H-i+1} \gamma^{2(h-1)} p_{\max}^h\right) \Delta}{n}} + \frac{6V_{\max}\left(1 + \sum_{h=1}^{H-i+1} \gamma^{h-1} p_{\max}^h\right)\Delta}{3n}$$

With these definitions, we know state the theorem

Theorem 7 (Formal version of Theorem 6)**.** *In the episodic reinforcement learning setting with discount factor γ, consider the doubly robust partial importance weighting estimators $\hat{\theta}_i := \hat{V}_{H-i+1}^{H}(\pi_{\mathrm{T}})$ for $i \in \{1, \dots, H+1\}$. Then* b *and* cnf *are valid and monotone, with $\kappa := (1 + \gamma p_{\max})^{-1}$.*

Proof of Theorem 7. We now turn to the proof.

Bias analysis. By repeatedly applying the tower property, the expectation for $\hat{V}_{\eta}^{H}$ is

$$\begin{aligned}
\mathbb{E}\left[\hat{V}_{\eta}^{H}\right] &= \mathbb{E}_{\pi_{\mathrm{L}}}\left[\hat{V}(x_1) + p_1(r_1 + \gamma \hat{V}_{\eta}^{H-1} - \hat{Q}(x_1, a_1)\right] \\
&= \mathbb{E}_{x_1}\left[\hat{V}(x_1) + \mathbb{E}_{a_1 \sim \pi_{\mathrm{L}}(x_1), a_{2:H} \sim \pi_{\mathrm{L}}}\left[p_1(r_1 + \gamma \hat{V}_{\eta}^{H-1} - \hat{Q}(x_1, a_1)) \mid x_1\right]\right] \\
&= \mathbb{E}_{x_1}\left[\hat{V}(x_1) + \mathbb{E}_{a_1 \sim \pi_{\mathrm{T}}(x_1), a_{2:H} \sim \pi_{\mathrm{L}}}\left[r_1 + \gamma \hat{V}_{\eta}^{H-1} - \hat{Q}(x_1, a_1) \mid x_1\right]\right] \\
&= \mathbb{E}_{x_1, a_1 \sim \pi_{\mathrm{T}}(x_1)}[r] + \gamma \mathbb{E}_{x_2 \sim \pi_{\mathrm{T}}, a_{2:H} \sim \pi_{\mathrm{L}}}\left[\hat{V}_{\eta}^{H-1}\right] \\
&= \dots \\
&= \mathbb{E}_{\pi_{\mathrm{T}}}\left[\sum_{h=1}^{\eta} \gamma^{h-1} r\right] + \gamma^{\eta} \mathbb{E}_{x_{\eta+1} \sim \pi_{\mathrm{T}}}\left[\hat{V}_{\mathrm{DM}}^{\eta}\right].
\end{aligned}$$

Here, we use that p_1 is the one-step importance weight, so it changes the action distribution from π_{L} to π_{T}. We also use the relationship between the direct models $\hat{Q}$ and $\hat{V}$. Therefore, the bias is

$$\left|\mathbb{E}\left[\hat{V}_{\eta}^{H}\right] - V(\pi_{\mathrm{T}})\right| = \gamma^{\eta}\left|\hat{V}_{\mathrm{DM}}^{\eta} - \mathbb{E}_{\pi_{\mathrm{T}}}[V(x_{\eta+1})]\right| \le \frac{\gamma^{\eta} - \gamma^{H}}{1-\gamma} =: \textsc{b}(H - \eta + 1) \tag{6}$$

The first identity justifies are choice of $\hat{V}_{\mathrm{DM}}^{\eta}$ which attempts to minimize this bias using the direct model. The inequality here follows from the fact that rewards are in $[0,1]$, which implies that values at time $\eta+1$ are in $\left[0, \frac{1-\gamma^{H-\eta}}{1-\gamma}\right]$. As $\gamma \in (0,1)$, clearly we have that $\textsc{b}(i)$ is monotonically increasing with i increasing. Thus this bias bound is valid.

Variance analysis. For the variance calculation, let $\mathbb{E}_h[\cdot], \mathrm{Var}_h(\cdot)$ denote expectation and variance conditional on all randomness *before* time step h. Adapting Theorem 1 of Jiang & Li (2016) the variance for $1 \le h < \eta$ is given by the recursive formula:

$$\operatorname*{Var}_h(\hat{V}_{\eta}^{H+1-h}) = \operatorname*{Var}_h(\mathbb{E}\left[\hat{V}_{\eta}^{H+1-h} \mid x_h\right]) + \mathbb{E}_h\left[\mathrm{Var}(p_h \Delta(x_h, a_h) \mid x_h)\right] + \mathbb{E}_h\left[p_h^2 \mathrm{Var}(r_h)\right] + \mathbb{E}_h\left[\gamma^2 p_h^2 \operatorname*{Var}_{h+1}(\hat{V}_{\eta}^{H-h})\right],$$

where $\Delta(x_h, a_h) := \hat{Q}(x_h, a_h) - Q(x_h, a_h)$. For $h = \eta$ it is identical, except that in the last term we use $\hat{V}^{\eta}_{\mathrm{DM}}$ instead of $\hat{V}^{H-\eta}_{\eta}$ (which is not defined).

Unrolling the recursion, the full expression for the variance is

$$\begin{aligned}
\mathrm{Var}(\hat{V}^H_\eta) = &\sum_{h=1}^{\eta} \mathbb{E}\left[\gamma^{2(h-1)}\rho^2_{h-1} \operatorname*{Var}_h(\mathbb{E}\left[\hat{V}^{H+1-h}_\eta \mid x_h\right])\right] \\
&+ \sum_{h=1}^{\eta} \mathbb{E}\left[\gamma^{2(h-1)}\rho^2_{h-1}\mathbb{E}_h\left[\mathrm{Var}(p_h\Delta(x_h, a_h) \mid x_h)\right]\right] \\
&+ \sum_{h=1}^{\eta} \mathbb{E}\left[\gamma^{2(h-1)}\rho^2_{h-1}\mathbb{E}_h\left[p_h^2\,\mathrm{Var}(r_h)\right]\right] \\
&+ \mathbb{E}\left[\gamma^{2\eta}\rho^2_\eta \operatorname*{Var}_{\eta+1}(\hat{V}^{\eta}_{\mathrm{DM}})\right].
\end{aligned}$$

For the variance bound, we do not attempt to obtain the sharpest bound possible. Instead, we use the following facts: (1) rewards are in $[0, 1]$, (2) all values, value estimates, and Q are at most $(1-\gamma)^{-1} =: V_{\max}$, and (3) for a random variable X that is bounded by B almost surely, we have $\mathrm{Var}(X) \le B^2$. Using these facts in each term gives

$$\begin{aligned}
\mathrm{Var}(\hat{V}^H_\eta) &\le \sum_{h=1}^{\eta} \mathbb{E}\left[\gamma^{2(h-1)}\rho^2_{h-1}V^2_{\max}\right] + \sum_{h=1}^{\eta} \mathbb{E}\left[\gamma^{2(h-1)}\rho^2_h V^2_{\max}\right] + \sum_{h=1}^{\eta} \mathbb{E}\left[\gamma^{2(h-1)}\rho^2_h\right] + \mathbb{E}\left[\gamma^{2\eta}\rho^2_\eta V^2_{\max}\right] \\
&= \sum_{h=1}^{\eta+1} \mathbb{E}\left[\gamma^{2(h-1)}\rho^2_{h-1}V^2_{\max}\right] + \sum_{h=1}^{\eta} \mathbb{E}\left[\gamma^{2(h-1)}\rho^2_h(V^2_{\max}+1)\right] \\
&\le 3V^2_{\max}\sum_{h=1}^{\eta} \mathbb{E}\left[\gamma^{2(h-1)}\rho^2_h\right] + V^2_{\max}
\end{aligned}$$

Here in the first line we use the three facts we stated above. In the second line we collect the terms. In the third line we note that $\gamma^{2(h-1)}\rho^2_{h-1} \le \gamma^{2(h-2)}\rho^2_{h-1}$ since $\gamma \in (0, 1)$, so we can re-index the first summation and group terms again.

To simplify further, let $p_{\max} := \sup_{x,a} \frac{\pi_{\mathrm{T}}(a|x)}{\pi_{\mathrm{L}}(a|x)}$ denote the largest importance weight and note that as $\mathbb{E}_h[p_h] = 1$, we have

$$\sum_{h=1}^{\eta} \mathbb{E}\left[\gamma^{2(h-1)}\rho^2_h\right] \le \sum_{h=1}^{\eta} \mathbb{E}\left[\gamma^{2(h-1)}p_{\max}\rho^2_{h-1}\mathbb{E}_h[w_h]\right] \le \ldots \le \sum_{h=1}^{\eta} \gamma^{2(h-1)}p^h_{\max}.$$

Therefore, our variance bound will be

$$\mathrm{Var}(\hat{V}^H_\eta) \le 3V^2_{\max}\left(1 + \sum_{h=1}^{\eta} \gamma^{2(h-1)}p^h_{\max}\right).$$

For the range, we obtain the recursion (for $1 \le h < \eta$):

$$\left|\hat{V}^{H+1-h}_\eta\right| \le V_{\max} + p_{\max}(1 + V_{\max}) + p_{\max}\gamma\left|\hat{V}^{H-h}_\eta\right|,$$

with the terminal condition $\left|\hat{V}^{\eta}_{\mathrm{DM}}\right| \le V_{\max}$. A somewhat crude upper bound is

$$\left|\hat{V}^H_\eta\right| \le 3V_{\max}\left(1 + \sum_{h=1}^{\eta} \gamma^{h-1}p^h_{\max}\right),$$

which has a similar form to the variance expression.

Therefore, Bernstein's inequality reveals that with probability $1-\delta$, we have that the n-trajectory empirical averages satisfy

$$\left|\hat{V}_\eta^H - \mathbb{E}\hat{V}_\eta^H\right| \leq \sqrt{\frac{6V_{\max}^2\left(1+\sum_{h=1}^{\eta}\gamma^{2(h-1)}p_{\max}^h\right)\log(2/\delta)}{n}} + \frac{6V_{\max}\left(1+\sum_{h=1}^{\eta}\gamma^{h-1}p_{\max}^h\right)\log(2/\delta)}{3n}. \tag{7}$$

This bound is clearly seen to be montonically increasing in η, which is montonically decreasing with i as required. The reason is that when we increase η we add one additional non-negative term to both the variance and range expressions.

Finally, we must verify that the bound does not decrease too quickly. For this, we first verify the following elementary fact

Fact 8. *Let $z \geq 0$ and $t \geq 0$ then*

$$\frac{1+\sum_{\tau=1}^{t} z^\tau}{1+\sum_{\tau=1}^{t-1} z^\tau} \leq 1+z.$$

Proof. Using the geometric series formula, we can rewrite

$$\frac{1+\sum_{\tau=1}^{t} z^\tau}{1+\sum_{\tau=1}^{t-1} z^\tau} = 1 + \frac{z^t}{1+\sum_{\tau=1}^{t-1} z^\tau} \leq 1+z. \qquad \square$$

Using the above fact, we can see that the variance bound decreases at rate $(1+\gamma^2 p_{\max})$ and the range bound decreases at rate $(1+\gamma p_{\max})$. The range bound dominates here, since

$$\sqrt{1+\gamma^2 p_{\max}} \leq \sqrt{1+\gamma^2 p_{\max}^2} \leq 1+\gamma p_{\max}$$

Therefore, we may take the decay constant to be $1/(1+\gamma p_{\max})$ to verify Assumption 2. $\square$

B. Details for continuous contextual bandits experiments

B.1. The simulation environment.

Here, we explain some of the important details of the simulation environment. The simulator is initialized with a d_x dimensional context space and action space $[0,1]^d$ for some parameter d. For our experiments we simply take $d=1$. There is also a hidden parameter matrix $\beta^\star \sim \mathcal{N}(0,I)$ with $\beta^\star \in \mathbb{R}^{d\times d_x}$. In each round, contexts are sampled iid from $\mathcal{N}(0,I)$, then the optimal action $a^\star(x) := \sigma(\beta^\star x)$, where $\sigma(z) = \frac{e^z}{e^z+1}$ is the standard sigmoid, and the function is applied component-wise. This optimal action $a^\star$ is used in the design of the reward functions.

We consider two different reward functions called "absolute value" and "quadratic." The first is simply $\ell(a) := 1 - \min(L\left\|a-a^\star(x)\right\|_1, 1)$, while the latter is $\ell(a) := 1-\min(L/4\sum_{j=1}^{d}(a_j - a_j^\star(x))^2, 1)$. Here L is the Lipschitz constant, which is also a configurable.

For policies, the uniform logging policy simply chooses $a \sim \text{Unif}([0,1]^d)$ on each round. Other logging and target policies are trained via regression on 10 vector-valued regression samples $(x, a^\star(x)+\mathcal{N}(0, 0.5\cdot I))$ where $x \sim \mathcal{N}(0,I)$. We use two different regression models: linear + sigmoid implemented in PyTorch, and a decision tree implemented in scikit-learn. Both regression procedures yield deterministic policies, and in our experiments we take this policies to be π_{T}.

For π_{L} we implement two softening techniques following Farajtabar et al. (2018), called "friendly" and "adversarial," and both techniques take two parameters α, β. Both methods are defined for discrete action spaces, and to adapt to the continuous setting we partition the continuous action space into m bins (for one-dimensional spaces). We round the deterministic action chosen by the regression model to its associated bin, run the softening procedure to choose a (potentially different) bin, and then sample an action uniformly from this bin. For higher dimensional action spaces, we discretize each dimension individually, so the softening results in a product measure.

Friendly softening with discrete actions is implemented as follows. We sample $U \sim \text{Unif}([-0.5, 0.5])$ and then the updated action is $\pi_{\text{det,disc}}(x)$ with probability $\alpha+\beta U$ and it is uniform over the remaining discrete actions with the remaining probability. Here $\pi_{\text{det,disc}}$ is the deterministic policy obtained by the regression model, discretized to one of the m bins. Adversarial softening instead is uniform over all discrete actions with probability $1-(\alpha+\beta U)$ and it is uniform over all

but $\pi_{\text{det,disc}}(x)$ with the remaining probability. In both cases, once we have a discrete action, we sample a continuous action from the corresponding bin.

The simulator also supports two different kernel functions: Epanechnikov and boxcar. The boxcar kernel is given by $K(u) = \frac{1}{2}\mathbf{1}\{|u| \leq 1\}$, while Epanechnikov is $K(u) = 0.75 \cdot (1 - u^2)\mathbf{1}\{|u| \leq 1\}$. We address boundary bias by normalizing the kernel appropriately, as opposed to forcing the target policy to choose actions in the interior. This issue is also discussed in Kallus & Zhou (2018).

Finally, we also vary the number of logged samples and the Lipschitz constant of the loss functions.

B.2. Reproducibility Checklist

Data collection process. All data are synthetically generated as described above.

Dataset and Simulation Environment. We will make the simulation environment publicly available.

Excluded Data. No excluded data.

Training/Validation/Testing allocation. There is no training/validation/testing setup in off policy evaluation. Instead all logged data are used for evaluation.

Hyper-parameters. Hyperparameters used in the experimental conditions are: $n \in 10^{1:5}$, $h \in \{2^{-(1:7)}$, $L \in \{0.1, 0.3, 1, 3, 10\}$, in addition to the other configurable parameters (e.g., softening technique, kernel, logging policy, target policy).

Evaluation runs. There are 1000 conditions, each with 30 replicates with different random seeds.

Description of experiments. For each condition, determined by logging policy, softening technique, target policy, sample size, lipschitz constant, reward function, and kernel type, we generate n logged samples following π_{L}, and 100k samples from π_{T} to estimate the ground truth $V(\pi_{\mathrm{T}})$. All fixed-bandwidth estimators and SLOPE are calculated based on the same logged data. The MSE is estimated by averaging across the 30 replicates, each with different random seed.

For the learning curve in the right panel of Figure 3 the specific condition shown is: uniform logging policy, linear+sigmoid target policy, $L = 3$, absolute value reward, boxcar kernel. MSE estimates are measured at $n = \{1, 3, 7\} \times 10^{1:3} \cup \{10,000\}$. We perform 100 replicates for this experiment.

Measure and Statistics. Results are shown in Figure 3. Statistics are based on empirical CDF calculated by aggregating the 1000 conditions. Typically there are no error bars for such plots. Pairwise comparison is based on paired t-test over all pair of methods and conditions, with significance level 0.05. The learning curve is based on 100 replicates, with error bar corresponding to ± 2 standard errors shown in the plots.

Computing infrastructure. Experiments were run on Microsoft Azure.

C. Details for reinforcement learning experiments

C.1. Experiment Details

Environment Description. We provide brief environment description below. More details can be found in Thomas & Brunskill (2016); Voloshin et al. (2019); Brockman et al. (2016).

- Mountain car is a classical benchmark from OpenAI Gym. We make the same modification as Voloshin et al. (2019). The domain has 2-dimensional state space (position and velocity) and one-dimensional action {left,nothing,right}. The reward is $r = -1$ for each timestep before reaching the goal. The initial state has position uniformly distributed in the discrete set $\{-0.4, -0.5, -0.6\}$ with velocity 0. The horizon is set to be $H = 250$ and there is an absorbing state at $(0.5, 0)$. The domain has deterministic dynamics, as well as deterministic, dense reward.
- Graph and Graph-POMDP are adopted from Voloshin et al. (2019). The Graph domain has horizon 16, state space $\{0, 1, 2, \cdots, 31\}$ and action space $\{0, 1\}$. The initial state is $x_0 = 0$, and we have the state-independent stochastic transition model with $\mathbb{P}(x_{t+1} = 2t + 1|a = 0) = 0.75$, $\mathbb{P}(x_{t+1} = 2t + 2|a = 0) = 0.25$, $\mathbb{P}(x_{t+1} = 2t + 1|a = 1) = 0.25$, $\mathbb{P}(x_{t+1} = 2t + 2|a = 1) = 0.75$. In the dense reward configuration, we have $r(x_t, a_t, x_{t+1}) = 2(x_{t+1} \mod 2) - 1 \; \forall t \leq T$. The sparse reward setting has $r(x_t, a_t, x_{t+1}) = 0 \; \forall t < T - 1$ with reward only at the last time

step, according to the dense reward function. We also consider a stochastic reward setting, where we change the reward to be $r(x_t, a_t, x_{t+1}) \sim \mathcal{N}(2(x_{t+1} \mod 2) - 1, 1)$. Graph-POMDP is a modified version of Graph where states are grouped into 6 groups. Only the group information is observed, so the states are aliased.

- Gridworld is also from Voloshin et al. (2019). The state space is an 8×8 grid with four actions [up, down, left, right]. The initial state distribution is uniform over the left column and top row, while the goal is in the bottom right corner. The horizon length is 25. The states belongs to four categories: Field, Hole, Goal, Others. The reward at Field is -0.005, Hole is -0.5, Goal is 1 and Others is -0.01. The exact map can be found in Voloshin et al. (2019).
- Hybrid Domain is from Thomas & Brunskill (2016). It is a composition of two other domains from the same study, called ModelWin and ModelFail. The ModelFail domain has horizon 2, four states $\{s_0, s_1, s_2, s_a\}$ and two actions $\{0, 1\}$. The agent starts at s_0, goes to s_1 with reward 1 if $a = 0$, and goes to s_2 with reward if $a = 1$. Then it transitions to the absorbing state s_a. This environment has partial observability so that $\{s_0, s_1, s_2\}$ are aliased together.
  In the hybrid domain the absorbing state s_a is replaced with a new state s_1 in the ModelWin domain. This domain has four states $\{s_1, s_2, s_3, s_a\}$. The action space is $\{0, 1\}$. The agent starts from s_1. Upon taking action $a = 0$, it goes to s_2 with probability 0.6 and receives reward 1, and goes to s_3 with probability 0.4 and reward -1. If $a = 1$, it does the opposite. From s_2 and s_3 the agent deterministically transitions to s_1 with 0 reward. do a deterministic transition back to s_1 with 0 reward. The horizon here is 20 and $x_{20} = s_a$. The states are fully observable.

Models. Instead of experiment with all possible approaches for direct modeling, which is quite burdensome, we follow the high-level guidelines provided in Table 3 of Voloshin et al. (2019)'s paper: for Graph, PO-Graph, and Mountain Car we use FQE because these environments are stochastic and have severe mismatch between logging and target policy. In contrast, Gridworld has moderate policy mismatch, so we use $Q^\pi(\lambda)$. For the Hybrid domain, we use a simple maximum-likelihood approximate model to predict the full transition operator and rewards, and plan in the model to estimate the value function.

Policy. For Gridworld and Mountain Car, we use ϵ-Greedy polices as logging and target policies. To derive these, we first train a base policy using value iteration and then we take $\pi(a^\star|x) = 1 - \epsilon$ and $\pi(a \mid x) = \epsilon/(|\mathcal{A}| - 1)$ for $a \neq a^\star(x)$, where $a^\star(x) = \operatorname{argmax} \hat{Q}(x, a)$ for the learned $\hat{Q}$ function. In Gridworld, we take the following policy pairs: $[(1, 0.1), (0.6, 0.1), (0.2, 0.1), (0.1, 0.2), (0.1, 0.6)]$, where the first argument is the ϵ parameter for . For Mountain Car domain, we take the following policy pairs: $[(0.1, 0), (1, 0), (1, 0.1), (0.1, 1)]$ where the first argument is the parameter for π_{L} and the second is for π_{T}. For the Graph and Graph-POMDP domain, both logging and target policies are static polices with probability p going left (marked as $a = 0$) and probability $1 - p$ going right (marked as $a = 1$), i.e., $\pi(a = 0|x) = p$ and $\pi(a = 1|x) = 1 - p \; \forall x$. In both environments, we vary p of the logging policy to be 0.2 and 0.6, while setting p for target policy to be 0.8. For the Hybrid domain, we use the same policy as Thomas & Brunskill (2016). For the first ModelFail part, $\pi_{\mathrm{L}}(a = 0) = 0.88$ and $\pi_{\mathrm{L}}(a = 1) = 0.12$, while the target policy does the opposite. For the second ModelWin part, $\pi_{\mathrm{L}}(a = 0|s_1) = 0.73$ and $\pi_{\mathrm{L}}(a = 1|s_1) = 0.27$, and the target policy does the opposite. For both policies, they select actions uniformly when $s \in \{s_2, s_3\}$.

Other parameters. For both the Graph and Graph-POMDP, we use $\gamma = 0.98$ and $N \in 2^{7:10}$. For Gridworld, $\gamma = 0.99$ and $N \in 2^{7:9}$. For Mountain Car, $\gamma = 0.96$ and $N \in 2^{8:10}$. For Hybrid, $\gamma = 0.99$ and $N \in \{10, 20, 50, \cdots, 10000, 20000, 50000\}$. Each condition is averaged over 100 replicates.

C.2. Reproducibility Checklist

Data collection process. All data are synthetically generated as described above.

Dataset and Simulation Environment. The Mountain Car environment is downloadable from OpenAI (Brockman et al., 2016). Graph, Graph-POMDP, Gridworld, and the Hybrid domain are available at `https://github.com/clvoloshin/OPE-tools`, which is the supporting code for Voloshin et al. (2019).

Excluded Data. No excluded data.

Training/Validation/Testing allocation. There is no training/validation/testing setup in off policy evaluation. Instead all logged data are used for evaluation.

Hyper-parameters. Hyperparameters (apart from those optimized by SLOPE) are optimized followng the guidelines of Voloshin et al. (2019). For MountainCar, the direct model is trained using a 2-layer fully connected neural network with hidden units 64 and 32. The batch size is 32 and convergence is set to be $1e-4$, network weights are initialized with

truncated Normal$(0, 0.1)$. For tabular models, convergence of Graph and Graph-POMDP is $1e-5$ and Gridworld is $4e-4$.

Evaluation runs. All conditions have 100 replicates with different random seeds.

Description of experiments. For each condition, determined by the choice of environment, stochastic/deterministic reward, sparse/dense reward, stochastic/deterministic transition model, logging policy π_{L}, target policy π_{T} and sample size N. We generate N logged trajectories following π_{L}, and 10000 samples from π_{T} to compute the ground truth $V(\pi_{\mathrm{T}})$. All baselines and SLOPE are calculated based on the same logged data. The MSE is estimated by averaging across the 100 replicates, each with different random seed.

Measure and Statistics. Results are shown in Figure 4. Statistics are based on empirical CDF calculated by aggregating the 106 conditions. Typically there are no error bars in ECDF plots. Pairwise comparison is based on paired t-test over all pair of methods over all conditions. Each test has significance level 0.05. Learning curve is based on Hybrid domain with 128 replicates, with error bar corresponding to ± 2 standard errors shown in the plots.

Computing infrastructure. RL experiments were conducted in a Linux compute cluster.