A Proofs from Section 4

In this section, we provide proofs for theoretical results in Section 4. Before the proofs, we note that all statements are proven in the case of finite state space (i.e., $|\mathcal{S}| < \infty$) and finite action space (i.e., $|\mathcal{A}| < \infty$) we define some commonly appearing notation symbols appearing in the proof:

- $P_{\mathcal{M}}$ and $r_{\mathcal{M}}$ (or P and r with no subscript for notational simplicity) denote the dynamics and reward function of the actual MDP $\mathcal{M}$
- $P_{\overline{\mathcal{M}}}$ and $r_{\overline{\mathcal{M}}}$ denote the dynamics and reward of the empirical MDP $\overline{\mathcal{M}}$ generated from the transitions in the dataset
- $P_{\widehat{\mathcal{M}}}$ and $r_{\widehat{\mathcal{M}}}$ denote the dynamics and reward of the MDP induced by the learned model $\widehat{\mathcal{M}}$

We also assume that whenever the cardinality of a particular state or state-action pair in the offline dataset $\mathcal{D}$, denoted by $|\mathcal{D}(\mathbf{s}, \mathbf{a})|$, appears in the denominator, we assume it is non-zero. For any non-existent $(\mathbf{s}, \mathbf{a}) \notin \mathcal{D}$, we can simply set $|\mathcal{D}(\mathbf{s}, \mathbf{a})|$ to be a small value < 1, which prevents any bound from producing trivially ∞ values.

A.1 A Useful Lemma and Its Proof

Before proving our main results, we first show that the penalty term in equation 4 is positive in expectation. Such a positive penalty is important to combat any overestimation that may arise as a result of using $\widehat{\mathcal{B}}$.

Lemma A.1 ((Interpolation Lemma)**.** *For any $f \in [0, 1]$, and any given $\rho(\mathbf{s}, \mathbf{a}) \in \Delta^{|\mathcal{S}||\mathcal{A}|}$, let d_f be an f-interpolation of ρ and $\mathcal{D}$, i.e., $d_f(\mathbf{s}, \mathbf{a}) := fd(\mathbf{s}, \mathbf{a}) + (1 - f)\rho(\mathbf{s}, \mathbf{a})$. For a given iteration k of Equation 4, we restate the definition of the expected penalty under $\rho(\mathbf{s}, \mathbf{a})$ in Eq. 5:*

$$\nu(\rho, f) := \mathbb{E}_{\mathbf{s},\mathbf{a}\sim\rho(\mathbf{s},\mathbf{a})}\left[\frac{\rho(\mathbf{s}, \mathbf{a}) - d(\mathbf{s}, \mathbf{a})}{d_f(\mathbf{s}, \mathbf{a})}\right].$$

Then $\nu(\rho, f)$ satisfies, (1) $\nu(\rho, f) \geq 0, \ \ \forall \rho, f$, (2) $\nu(\rho, f)$ is monotonically increasing in f for a fixed ρ, and (3) $\nu(\rho, f) = 0$ iff $\forall\, \mathbf{s}, \mathbf{a}, \ \rho(\mathbf{s}, \mathbf{a}) = d(\mathbf{s}, \mathbf{a})$ or $f = 0$.

Proof. To prove this lemma, we use algebraic manipulation on the expression for quantity $\nu(\rho, f)$ and show that it is indeed positive and monotonically increasing in $f \in [0, 1]$.

$$\begin{aligned}
\nu(\rho, f) &= \sum_{\mathbf{s},\mathbf{a}} \rho(\mathbf{s}, \mathbf{a}) \left(\frac{\rho(\mathbf{s}, \mathbf{a}) - d(\mathbf{s}, \mathbf{a})}{fd(\mathbf{s}, \mathbf{a}) + (1 - f)\rho(\mathbf{s}, \mathbf{a})}\right) \\
&= \sum_{\mathbf{s},\mathbf{a}} \rho(\mathbf{s}, \mathbf{a}) \left(\frac{\rho(\mathbf{s}, \mathbf{a}) - d(\mathbf{s}, \mathbf{a})}{\rho(\mathbf{s}, \mathbf{a}) + f(d(\mathbf{s}, \mathbf{a}) - \rho(\mathbf{s}, \mathbf{a}))}\right) \qquad (6) \\
\implies \frac{d\nu(\rho, f)}{df} &= \sum_{\mathbf{s},\mathbf{a}} \rho(\mathbf{s}, \mathbf{a}) \left(\rho(\mathbf{s}, \mathbf{a}) - d(\mathbf{s}, \mathbf{a})\right)^2 \cdot \left(\frac{1}{(\rho(\mathbf{s}, \mathbf{a}) + f(d(\mathbf{s}, \mathbf{a}) - \rho(\mathbf{s}, \mathbf{a}))}\right)^2 \geq 0 \\
&\qquad \forall f \in [0, 1]. \qquad (7)
\end{aligned}$$

Since the derivative of $\nu(\rho, f)$ with respect to f is always positive, it is an increasing function of f for a fixed ρ, and this proves the second part (2) of the Lemma. Using this property, we can show the part (1) of the Lemma as follows:

$$\begin{aligned}
\forall f \in (0, 1], \ \nu(\rho, f) \geq \nu(\rho, 0) = \sum_{\mathbf{s},\mathbf{a}} \rho(\mathbf{s}, \mathbf{a}) \frac{\rho(\mathbf{s}, \mathbf{a}) - d(\mathbf{s}, \mathbf{a})}{\rho(\mathbf{s}, \mathbf{a})} &= \sum_{\mathbf{s},\mathbf{a}} \left(\rho(\mathbf{s}, \mathbf{a}) - d(\mathbf{s}, \mathbf{a})\right) \\
&= 1 - 1 = 0. \qquad (8)
\end{aligned}$$

Finally, to prove the third part (3) of this Lemma, note that when $f = 0$, $\nu(\rho, f) = 0$ (as shown above), and similarly by setting $\rho(\mathbf{s}, \mathbf{a}) = d(\mathbf{s}, \mathbf{a})$ note that we obtain $\nu(\rho, f) = 0$. To prove the only if side of (3), assume that $f \neq 0$ and $\rho(\mathbf{s}, \mathbf{a}) \neq d(\mathbf{s}, \mathbf{a})$ and we will show that in this case $\nu(\rho, f) \neq 0$. When $d(\mathbf{s}, \mathbf{a}) \neq \rho(\mathbf{s}, \mathbf{a})$, the derivative $\frac{d\nu(\rho,f)}{df} > 0$ (i.e., strictly positive) and hence the function $\nu(\rho, f)$ is a strictly increasing function of f. Thus, in this case, $\nu(\rho, f) > 0 = \nu(\rho, 0)\ \forall f > 0$. Thus we have shown that if $\rho(\mathbf{s}, \mathbf{a}) \neq d(\mathbf{s}, \mathbf{a})$ and $f > 0$, $\nu(\rho, f) \neq 0$, which completes our proof for the only if side of (3). □

A.2 Proof of Proposition 4.1

Before proving this proposition, we provide a bound on the Bellman backup in the empirical MDP, $\mathcal{B}_{\overline{\mathcal{M}}}$. To do so, we formally define the standard concentration properties of the reward and transition dynamics in the empirical MDP, $\overline{\mathcal{M}}$, that we assume so as to prove Proposition A.1. Following prior work [42, 19, 29], we assume:

Assumption A1. $\forall\, \mathbf{s}, \mathbf{a} \in \mathcal{M}$, *the following relationships hold with high probability,* $\geq 1 - \delta$

$$|r_{\overline{\mathcal{M}}}(\mathbf{s}, \mathbf{a}) - r(\mathbf{s}, \mathbf{a})| \leq \frac{C_{r,\delta}}{\sqrt{|\mathcal{D}(\mathbf{s}, \mathbf{a})|}}, \quad ||P_{\overline{\mathcal{M}}}(\mathbf{s}'|\mathbf{s}, \mathbf{a}) - P(\mathbf{s}'|\mathbf{s}, \mathbf{a})||_1 \leq \frac{C_{P,\delta}}{\sqrt{|\mathcal{D}(\mathbf{s}, \mathbf{a})|}}.$$

Under this assumption and assuming that the reward function in the MDP, $r(\mathbf{s}, \mathbf{a})$ is bounded, as $|r(\mathbf{s}, \mathbf{a})| \leq R_{\max}$, we can bound the difference between the empirical Bellman operator, $\mathcal{B}_{\overline{\mathcal{M}}}$ and the actual MDP, $\mathcal{B}_{\mathcal{M}}$,

$$\begin{aligned}
\left|\left(\mathcal{B}_{\overline{\mathcal{M}}}{}^{\pi}\hat{Q}^k\right) - \left(\mathcal{B}^{\pi}_{\mathcal{M}}\hat{Q}^k\right)\right| &= |(r_{\overline{\mathcal{M}}}(\mathbf{s}, \mathbf{a}) - r_{\mathcal{M}}(\mathbf{s}, \mathbf{a})) \\
&\quad + \gamma \sum_{\mathbf{s}'} \left(P_{\overline{\mathcal{M}}}(\mathbf{s}'|\mathbf{s}, \mathbf{a}) - P_{\mathcal{M}}(\mathbf{s}'|\mathbf{s}, \mathbf{a})\right) \mathbb{E}_{\pi(\mathbf{a}'|\mathbf{s}')}\left[\hat{Q}^k(\mathbf{s}', \mathbf{a}')\right]\Big| \\
&\leq |r_{\overline{\mathcal{M}}}(\mathbf{s}, \mathbf{a}) - r_{\mathcal{M}}(\mathbf{s}, \mathbf{a})| \\
&\quad + \gamma \left|\sum_{\mathbf{s}'} \left(P_{\overline{\mathcal{M}}}(\mathbf{s}'|\mathbf{s}, \mathbf{a}) - P_{\mathcal{M}}(\mathbf{s}'|\mathbf{s}, \mathbf{a})\right) \mathbb{E}_{\pi(\mathbf{a}'|\mathbf{s}')}\left[\hat{Q}^k(\mathbf{s}', \mathbf{a}')\right]\right| \\
&\leq \frac{C_{r,\delta} + \gamma C_{P,\delta} 2R_{\max}/(1-\gamma)}{\sqrt{|\mathcal{D}(\mathbf{s}, \mathbf{a})|}}.
\end{aligned}$$

Thus the overestimation due to sampling error in the empirical MDP, $\overline{\mathcal{M}}$ is bounded as a function of a bigger constant, $C_{r,P,\delta}$ that can be expressed as a function of $C_{r,\delta}$ and $C_{P,\delta}$, and depends on δ via a $\sqrt{\log(1/\delta)}$ dependency. For the purposes of proving Proposition A.1, we assume that:

$$\forall \mathbf{s}, \mathbf{a}, \quad \left|\left(\mathcal{B}_{\overline{\mathcal{M}}}{}^{\pi}\hat{Q}^k\right) - \left(\mathcal{B}^{\pi}_{\mathcal{M}}\hat{Q}^k\right)\right| \leq \frac{C_{r,T,\delta} R_{\max}}{(1-\gamma)\sqrt{|\mathcal{D}(\mathbf{s}, \mathbf{a})|}}. \tag{9}$$

Next, we provide a bound on the error between the bellman backup induced by the learned dynamics model and the learned reward, $\mathcal{B}_{\widehat{\mathcal{M}}}$, and the actual Bellman backup, $\mathcal{B}_{\mathcal{M}}$. To do so, we note that:

$$\begin{aligned}
\left|\left(\mathcal{B}_{\widehat{\mathcal{M}}}{}^{\pi}\hat{Q}^k\right) - \left(\mathcal{B}^{\pi}_{\mathcal{M}}\hat{Q}^k\right)\right| &= \left|\left(r_{\widehat{\mathcal{M}}}(\mathbf{s}, \mathbf{a}) - r_{\mathcal{M}}(\mathbf{s}, \mathbf{a})\right)\right. \qquad (10) \\
&\quad \left. + \gamma \sum_{\mathbf{s}'} \left(P_{\widehat{\mathcal{M}}}(\mathbf{s}'|\mathbf{s}, \mathbf{a}) - P_{\mathcal{M}}(\mathbf{s}'|\mathbf{s}, \mathbf{a})\right) \mathbb{E}_{\pi(\mathbf{a}'|\mathbf{s}')}\left[\hat{Q}^k(\mathbf{s}', \mathbf{a}')\right]\right| \\
&\leq |r_{\widehat{\mathcal{M}}}(\mathbf{s}, \mathbf{a}) - r_{\mathcal{M}}(\mathbf{s}, \mathbf{a})| + \gamma \frac{2R_{\max}}{1-\gamma} D(P, P_{\widehat{\mathcal{M}}}), \qquad (11)
\end{aligned}$$

where $D(P, P_{\widehat{\mathcal{M}}})$ is the total-variation divergence between the learned dynamics model and the actual MDP. Now, we show that the asymptotic Q-function learned by COMBO lower-bounds the actual Q-function of any policy π with high probability for a large enough $\beta \geq 0$. We will use Equations 9 and 11 to prove such a result.

Proposition A.1 (Asymptotic lower-bound)**.** *Let P^{π} denote the Hadamard product of the dynamics P and a given policy π in the actual MDP and let $S^{\pi} := (I - \gamma P^{\pi})^{-1}$. Let D denote the total-variation divergence between two probability distributions. For any $\pi(\mathbf{a}|\mathbf{s})$, the Q-function obtained by recursively applying Equation 4, with $\hat{\mathcal{B}}^{\pi} = f\mathcal{B}^{\pi}_{\overline{\mathcal{M}}} + (1-f)\mathcal{B}^{\pi}_{\widehat{\mathcal{M}}}$, with probability at least $1 - \delta$, results in $\hat{Q}^{\pi}$ that satisfies:*

$$\begin{aligned}
\forall \mathbf{s}, \mathbf{a},\ \hat{Q}^{\pi}(\mathbf{s}, \mathbf{a}) \leq Q^{\pi}(\mathbf{s}, \mathbf{a}) - \beta \cdot \left[S^{\pi}\left[\frac{\rho - d}{d_f}\right]\right](\mathbf{s}, \mathbf{a}) + f\left[S^{\pi}\left[\frac{C_{r,T,\delta} R_{\max}}{(1-\gamma)\sqrt{|\mathcal{D}|}}\right]\right](\mathbf{s}, \mathbf{a}) \\
+ (1-f)\left[S^{\pi}\left[|r - r_{\widehat{\mathcal{M}}}| + \frac{2\gamma R_{\max}}{1-\gamma} D(P, P_{\widehat{\mathcal{M}}})\right]\right](\mathbf{s}, \mathbf{a}).
\end{aligned}$$

Proof. We first note that the Bellman backup $\hat{\mathcal{B}}^\pi$ induces the following Q-function iterates as per Equation 4,

$$\begin{aligned}
\hat{Q}^{k+1}(\mathbf{s},\mathbf{a}) &= \left(\hat{\mathcal{B}}^\pi \hat{Q}^k\right)(\mathbf{s},\mathbf{a}) - \beta \frac{\rho(\mathbf{s},\mathbf{a}) - d(\mathbf{s},\mathbf{a})}{d_f(\mathbf{s},\mathbf{a})} \\
&= f\left(\mathcal{B}^\pi_{\mathcal{M}} \hat{Q}^k\right)(\mathbf{s},\mathbf{a}) + (1-f)\left(\mathcal{B}^\pi_{\widehat{\mathcal{M}}} \hat{Q}^k\right)(\mathbf{s},\mathbf{a}) - \beta \frac{\rho(\mathbf{s},\mathbf{a}) - d(\mathbf{s},\mathbf{a})}{d_f(\mathbf{s},\mathbf{a})} \\
&= \left(\mathcal{B}^\pi \hat{Q}^k\right)(\mathbf{s},\mathbf{a}) - \beta \frac{\rho(\mathbf{s},\mathbf{a}) - d(\mathbf{s},\mathbf{a})}{d_f(\mathbf{s},\mathbf{a})} + (1-f)\left(\mathcal{B}_{\widehat{\mathcal{M}}}{}^\pi \hat{Q}^k - \mathcal{B}^\pi \hat{Q}^k\right)(\mathbf{s},\mathbf{a}) \\
&+ f\left(\mathcal{B}_{\overline{\mathcal{M}}}{}^\pi \hat{Q}^k - \mathcal{B}^\pi \hat{Q}^k\right)(\mathbf{s},\mathbf{a}) \\
\forall \mathbf{s},\mathbf{a},\ \hat{Q}^{k+1} &\leq \left(\mathcal{B}^\pi \hat{Q}^k\right) - \beta \frac{\rho - d}{d_f} + (1-f)\left[|r_{\widehat{\mathcal{M}}} - r_{\mathcal{M}}| + \frac{2\gamma R_{\max}}{1-\gamma} D(P, P_{\widehat{\mathcal{M}}})\right] + f \frac{C_{r,T,\delta} R_{\max}}{(1-\gamma)\sqrt{|\mathcal{D}|}}
\end{aligned}$$

Since the RHS upper bounds the Q-function pointwise for each $(\mathbf{s},\mathbf{a})$, the fixed point of the Bellman iteration process will be pointwise smaller than the fixed point of the Q-function found by solving for the RHS via equality. Thus, we get that

$$\begin{aligned}
\hat{Q}^\pi(\mathbf{s},\mathbf{a}) \leq \underbrace{S^\pi r_{\mathcal{M}}}_{=Q^\pi(\mathbf{s},\mathbf{a})} - \beta\left[S^\pi\left[\frac{\rho - d}{d_f}\right]\right](\mathbf{s},\mathbf{a}) + f\left[S^\pi\left[\frac{C_{r,T,\delta} R_{\max}}{(1-\gamma)\sqrt{|\mathcal{D}|}}\right]\right](\mathbf{s},\mathbf{a}) \\
+ (1-f)\left[S^\pi\left[|r - r_{\widehat{\mathcal{M}}}| + \frac{2\gamma R_{\max}}{1-\gamma} D(P, P_{\widehat{\mathcal{M}}})\right]\right](\mathbf{s},\mathbf{a}),
\end{aligned}$$

which completes the proof of this proposition. □

Next, we use the result and proof technique from Proposition A.1 to prove Corollary 4.1, that in expectation under the initial state-distribution, the expected Q-value is indeed a lower-bound.

Corollary A.1 (Corollary 4.1 restated)**.** *For a sufficiently large β, we have a lower-bound that $\mathbb{E}_{\mathbf{s}\sim\mu_0,\mathbf{a}\sim\pi(\cdot|\mathbf{s})}[\hat{Q}^\pi(\mathbf{s},\mathbf{a})] \leq \mathbb{E}_{\mathbf{s}\sim\mu_0,\mathbf{a}\sim\pi(\cdot|\mathbf{s})}[Q^\pi(\mathbf{s},\mathbf{a})]$, where $\mu_0(\mathbf{s})$ is the initial state distribution. Furthermore, when ϵ_s is small, such as in the large sample regime; or when the model bias ϵ_m is small, a small β is sufficient along with an appropriate choice of f.*

Proof. To prove this corollary, we note a slightly different variant of Proposition A.1. To observe this, we will deviate from the proof of Proposition A.1 slightly and will aim to express the inequality using $\mathcal{B}_{\widehat{\mathcal{M}}}$, the Bellman operator defined by the learned model and the reward function. Denoting $(I - \gamma P_{\widehat{\mathcal{M}}})^{-1}$ as $S^\pi_{\widehat{\mathcal{M}}}$, doing this will intuitively allow us to obtain $\beta\left(\mu(\mathbf{s})\pi(\mathbf{a}|\mathbf{s})\right)^T\left(S^\pi_{\widehat{\mathcal{M}}}\left[\frac{\rho-d}{d_f}\right]\right)(\mathbf{s},\mathbf{a})$ as the conservative penalty which can be controlled by choosing β appropriately so as to nullify the potential overestimation caused due to other terms. Formally,

$$\begin{aligned}
\hat{Q}^{k+1}(\mathbf{s},\mathbf{a}) &= \left(\hat{\mathcal{B}}^\pi \hat{Q}^k\right)(\mathbf{s},\mathbf{a}) - \beta \frac{\rho(\mathbf{s},\mathbf{a}) - d(\mathbf{s},\mathbf{a})}{d_f(\mathbf{s},\mathbf{a})} = \left(\mathcal{B}^\pi_{\widehat{\mathcal{M}}} \hat{Q}^k\right)(\mathbf{s},\mathbf{a}) - \beta \frac{\rho(\mathbf{s},\mathbf{a}) - d(\mathbf{s},\mathbf{a})}{d_f(\mathbf{s},\mathbf{a})} \\
&+ f \underbrace{\left(\mathcal{B}^\pi_{\mathcal{M}} - \mathcal{B}^\pi_{\widehat{\mathcal{M}}} \hat{Q}^k\right)(\mathbf{s},\mathbf{a})}_{:=\Delta(\mathbf{s},\mathbf{a})}
\end{aligned}$$

By controlling $\Delta(\mathbf{s},\mathbf{a})$ using the pointwise triangle inequality:

$$\forall \mathbf{s},\mathbf{a},\ \left|\mathcal{B}^\pi_{\mathcal{M}} \hat{Q}^k - \mathcal{B}^\pi_{\widehat{\mathcal{M}}} \hat{Q}^k\right| \leq \left|\mathcal{B}^\pi \hat{Q}^k - \mathcal{B}^\pi_{\widehat{\mathcal{M}}} \hat{Q}^k\right| + \left|\mathcal{B}^\pi_{\mathcal{M}} \hat{Q}^k - \mathcal{B}^\pi \hat{Q}^k\right|, \tag{12}$$

and then iterating the backup $\mathcal{B}^\pi_{\widehat{\mathcal{M}}}$ to its fixed point and finally noting that $\rho(\mathbf{s},\mathbf{a}) = \left((\mu\cdot\pi)^T S^\pi_{\widehat{\mathcal{M}}}\right)(\mathbf{s},\mathbf{a})$, we obtain:

$$\mathbb{E}_{\mu,\pi}[\hat{Q}^\pi(\mathbf{s},\mathbf{a})] \leq \mathbb{E}_{\mu,\pi}[Q^\pi_{\widehat{\mathcal{M}}}(\mathbf{s},\mathbf{a})] - \beta\, \mathbb{E}_{\rho(\mathbf{s},\mathbf{a})}\left[\frac{\rho(\mathbf{s},\mathbf{a}) - d(\mathbf{s},\mathbf{a})}{d_f(\mathbf{s},\mathbf{a})}\right] + \text{terms independent of } \beta. \tag{13}$$

The terms marked as "terms independent of β" correspond to the additional positive error terms obtained by iterating $\left|\mathcal{B}^\pi \hat{Q}^k - \mathcal{B}^\pi_{\widehat{\mathcal{M}}} \hat{Q}^k\right|$ and $\left|\mathcal{B}^\pi_{\widehat{\mathcal{M}}} \hat{Q}^k - \mathcal{B}^\pi \hat{Q}^k\right|$, which can be bounded similar to the proof of Proposition A.1 above. Now by replacing the model Q-function, $\mathbb{E}_{\mu,\pi}[Q^\pi_{\widehat{\mathcal{M}}}(\mathbf{s},\mathbf{a})]$ with the actual Q-function, $\mathbb{E}_{\mu,\pi}[Q^\pi(\mathbf{s},\mathbf{a})]$ and adding an error term corresponding to model error to the bound, we obtain that:

$$\mathbb{E}_{\mu,\pi}[\hat{Q}^\pi(\mathbf{s},\mathbf{a})] \leq \mathbb{E}_{\mu,\pi}[Q^\pi(\mathbf{s},\mathbf{a})] + \text{terms independent of } \beta - \beta \underbrace{\mathbb{E}_{\rho(\mathbf{s},\mathbf{a})}\left[\frac{\rho(\mathbf{s},\mathbf{a}) - d(\mathbf{s},\mathbf{a})}{d_f(\mathbf{s},\mathbf{a})}\right]}_{=\nu(\rho,f)>0}. \tag{14}$$

Hence, by choosing β large enough, we obtain the desired lower bound guarantee. □

Remark 1 (COMBO does not underestimate at every $\mathbf{s} \in \mathcal{D}$ unlike CQL.). *Before concluding this section, we discuss how the bound obtained by COMBO (Equation 14) is tighter than CQL. CQL learns a Q-function such that the value of the policy under the resulting Q-function lower-bounds the true value function at each state $\mathbf{s} \in \mathcal{D}$ individually (in the absence of no sampling error), i.e., $\forall \mathbf{s} \in \mathcal{D}, \hat{V}^\pi_{CQL}(\mathbf{s}) \leq V^\pi(\mathbf{s})$, whereas the bound in COMBO is only valid in expectation of the value function over the initial state distribution, i.e., $\mathbb{E}_{\mathbf{s}\sim\mu_0(\mathbf{s})}[\hat{V}^\pi_{COMBO}(\mathbf{s})] \leq \mathbb{E}_{\mathbf{s}\sim\mu_0(\mathbf{s})}[V^\pi(\mathbf{s})]$, and the value function at a given state may not be a lower-bound. For instance, COMBO can overestimate the value of a state more frequent in the dataset distribution $d(\mathbf{s},\mathbf{a})$ but not so frequent in the $\rho(\mathbf{s},\mathbf{a})$ marginal distribution of the policy under the learned model $\widehat{\mathcal{M}}$. To see this more formally, note that the expected penalty added in the effective Bellman backup performed by COMBO (Equation 4), in expectation under the dataset distribution $d(\mathbf{s},\mathbf{a})$, $\widetilde{\nu}(\rho,d,f)$ is actually* ***negative****:*

$$\widetilde{\nu}(\rho,d,f) = \sum_{\mathbf{s},\mathbf{a}} d(\mathbf{s},\mathbf{a}) \frac{\rho(\mathbf{s},\mathbf{a}) - d(\mathbf{s},\mathbf{a})}{d_f(\mathbf{s},\mathbf{a})} = -\sum_{\mathbf{s},\mathbf{a}} d(\mathbf{s},\mathbf{a}) \frac{d(\mathbf{s},\mathbf{a}) - \rho(\mathbf{s},\mathbf{a})}{f d(\mathbf{s},\mathbf{a}) + (1-f)\rho(\mathbf{s},\mathbf{a})} < 0,$$

where the final inequality follows via a direct application of the proof of Lemma A.1. Thus, COMBO actually overestimates *the values at atleast some states (in the dataset) unlike CQL.*

A.3 Proof of Proposition 4.2

In this section, we will provide a proof for Proposition 4.2, and show that the COMBO can be less conservative in terms of the estimated value. To recall, let $\Delta^\pi_{\text{COMBO}} := \mathbb{E}_{\mathbf{s},\mathbf{a}\sim d_{\widehat{\mathcal{M}}}(\mathbf{s}),\pi(\mathbf{a}|\mathbf{s})}\left[\hat{Q}^\pi(\mathbf{s},\mathbf{a}\right]$ and let $\Delta^\pi_{\text{CQL}} := \mathbb{E}_{\mathbf{s},\mathbf{a}\sim d_{\widehat{\mathcal{M}}},\pi(\mathbf{a}|\mathbf{s})}\left[\hat{Q}^\pi_{\text{CQL}}(\mathbf{s},\mathbf{a})\right]$. From Kumar et al. [29], we obtain that $\hat{Q}^\pi_{\text{CQL}}(\mathbf{s},\mathbf{a}) := Q^\pi(\mathbf{s},\mathbf{a}) - \beta\frac{\pi(\mathbf{a}|\mathbf{s}) - \pi_\beta(\mathbf{a}|\mathbf{s})}{\pi_\beta(\mathbf{a}|\mathbf{s})}$. We shall derive the condition for the real data fraction $f = 1$ for COMBO, thus making sure that $d_f(\mathbf{s}) = d^{\pi_\beta}(\mathbf{s})$. To derive the condition when $\Delta^\pi_{\text{COMBO}} \geq \Delta^\pi_{\text{CQL}}$, we note the following simplifications:

$$\Delta^\pi_{\text{COMBO}} \geq \Delta^\pi_{\text{CQL}} \tag{15}$$

$$\implies \sum_{\mathbf{s},\mathbf{a}} d_{\widehat{\mathcal{M}}}(\mathbf{s})\pi(\mathbf{a}|\mathbf{s})\hat{Q}^\pi(\mathbf{s},\mathbf{a}) \geq \sum_{\mathbf{s},\mathbf{a}} d_{\widehat{\mathcal{M}}}(\mathbf{s})\pi(\mathbf{a}|\mathbf{s})\hat{Q}^\pi_{\text{CQL}}(\mathbf{s},\mathbf{a}) \tag{16}$$

$$\implies \beta \sum_{\mathbf{s},\mathbf{a}} d_{\widehat{\mathcal{M}}}(\mathbf{s})\pi(\mathbf{a}|\mathbf{s}) \left(\frac{\rho(\mathbf{s},\mathbf{a}) - d^{\pi_\beta}(\mathbf{s})\pi_\beta(\mathbf{a}|\mathbf{s})}{d^{\pi_\beta}(\mathbf{s})\pi_\beta(\mathbf{a}|\mathbf{s})}\right) \leq \beta \sum_{\mathbf{s},\mathbf{a}} d_{\widehat{\mathcal{M}}}(\mathbf{s})\pi(\mathbf{a}|\mathbf{s}) \left(\frac{\pi(\mathbf{a}|\mathbf{s}) - \pi_\beta(\mathbf{a}|\mathbf{s})}{\pi_\beta(\mathbf{a}|\mathbf{s})}\right). \tag{17}$$

Now, in the expression on the left-hand side, we add and subtract $d^{\pi_\beta}(\mathbf{s})\pi(\mathbf{a}|\mathbf{s})$ from the numerator inside the paranthesis.

$$\sum_{\mathbf{s},\mathbf{a}} d_{\overline{\mathcal{M}}}(\mathbf{s},\mathbf{a}) \left(\frac{\rho(\mathbf{s},\mathbf{a}) - d^{\pi_\beta}(\mathbf{s})\pi_\beta(\mathbf{a}|\mathbf{s})}{d^{\pi_\beta}(\mathbf{s})\pi_\beta(\mathbf{a}|\mathbf{s})} \right) \tag{18}$$

$$= \sum_{\mathbf{s},\mathbf{a}} d_{\overline{\mathcal{M}}}(\mathbf{s},\mathbf{a}) \left(\frac{\rho(\mathbf{s},\mathbf{a}) - d^{\pi_\beta}(\mathbf{s})\pi(\mathbf{a}|\mathbf{s}) + d^{\pi_\beta}(\mathbf{s})\pi(\mathbf{a}|\mathbf{s}) - d^{\pi_\beta}(\mathbf{s})\pi_\beta(\mathbf{a}|\mathbf{s})}{d^{\pi_\beta}(\mathbf{s})\pi_\beta(\mathbf{a}|\mathbf{s})} \right) \tag{19}$$

$$= \underbrace{\sum_{\mathbf{s},\mathbf{a}} d_{\overline{\mathcal{M}}}(\mathbf{s},\mathbf{a}) \frac{\pi(\mathbf{a}|\mathbf{s}) - \pi_\beta(\mathbf{a}|\mathbf{s})}{\pi_\beta(\mathbf{a}|\mathbf{s})}}_{(1)} + \sum_{\mathbf{s},\mathbf{a}} d_{\overline{\mathcal{M}}}(\mathbf{s},\mathbf{a}) \cdot \frac{\rho(\mathbf{s}) - d^{\pi_\beta}(\mathbf{s})}{d^{\pi_\beta}(\mathbf{s})} \cdot \frac{\pi(\mathbf{a}|\mathbf{s})}{\pi_\beta(\mathbf{a}|\mathbf{s})} \tag{20}$$

The term marked (1) is identical to the CQL term that appears on the right in Equation 17. Thus the inequality in Equation 17 is satisfied when the second term above is negative. To show this, first note that $d^{\pi_\beta}(\mathbf{s}) = d_{\overline{\mathcal{M}}}(\mathbf{s})$ which results in a cancellation. Finally, re-arranging the second term into expectations gives us the desired result. An analogous condition can be derived when $f \neq 1$, but we omit that derivation as it will be hard to interpret terms appear in the final inequality.

A.4 Proof of Proposition 4.3

To prove the policy improvement result in Proposition 4.3, we first observe that using Equation 4 for Bellman backups amounts to finding a policy that maximizes the return of the policy in the a modified "f-interpolant" MDP which admits the Bellman backup $\widehat{\mathcal{B}}^\pi$, and is induced by a linear interpolation of backups in the empirical MDP $\overline{\mathcal{M}}$ and the MDP induced by a dynamics model $\widehat{\mathcal{M}}$ and the return of a policy π in this effective f-interpolant MDP is denoted by $J(\overline{\mathcal{M}}, \widehat{\mathcal{M}}, f, \pi)$. Alongside this, the return is penalized by the conservative penalty where ρ^π denotes the marginal state-action distribution of policy π in the learned model $\widehat{\mathcal{M}}$.

$$\hat{J}(f,\pi) = J(\overline{\mathcal{M}}, \widehat{\mathcal{M}}, f, \pi) - \beta \frac{\nu(\rho^\pi, f)}{1-\gamma}. \tag{21}$$

We will require bounds on the return of a policy π in this f-interpolant MDP, $J(\overline{\mathcal{M}}, \widehat{\mathcal{M}}, f, \pi)$, which we first prove separately as Lemma A.2 below and then move to the proof of Proposition 4.3.

Lemma A.2 (Bound on return in f-interpolant MDP)**.** *For any two MDPs, $\mathcal{M}_1$ and $\mathcal{M}_2$, with the same state-space, action-space and discount factor, and for a given fraction $f \in [0,1]$, define the f-interpolant MDP $\mathcal{M}_f$ as the MDP on the same state-space, action-space and with the same discount as the MDP with dynamics: $P_{\mathcal{M}_f} := fP_{\mathcal{M}_1} + (1-f)P_{\mathcal{M}_2}$ and reward function: $r_{\mathcal{M}_f} := fr_{\mathcal{M}_1} + (1-f)r_{\mathcal{M}_2}$. Then, given any auxiliary MDP, $\mathcal{M}$, the return of any policy π in $\mathcal{M}_f$, $J(\pi, \mathcal{M}_f)$, also denoted by $J(\mathcal{M}_1, \mathcal{M}_2, f, \pi)$, lies in the interval:*

$$\big[J(\pi,\mathcal{M}) - \alpha,\ \ J(\pi,\mathcal{M}) + \alpha\big], \qquad \textit{where } \alpha \textit{ is given by:}$$

$$\alpha = \frac{2\gamma(1-f)}{(1-\gamma)^2} R_{\max} D\left(P_{\mathcal{M}_2}, P_{\mathcal{M}}\right) + \frac{\gamma f}{1-\gamma} \left| \mathbb{E}_{d^\pi_{\mathcal{M}}\pi} \left[\left(P^\pi_{\mathcal{M}} - P^\pi_{\mathcal{M}_1} \right) Q^\pi_{\mathcal{M}} \right] \right|$$
$$+ \frac{f}{1-\gamma} \mathbb{E}_{\mathbf{s},\mathbf{a}\sim d^\pi_{\mathcal{M}}\pi} [|r_{\mathcal{M}_1}(\mathbf{s},\mathbf{a}) - r_{\mathcal{M}}(\mathbf{s},\mathbf{a})|] + \frac{1-f}{1-\gamma} \mathbb{E}_{\mathbf{s},\mathbf{a}\sim d^\pi_{\mathcal{M}}\pi} [|r_{\mathcal{M}_2}(\mathbf{s},\mathbf{a}) - r_{\mathcal{M}}(\mathbf{s},\mathbf{a})|]. \tag{22}$$

Proof. To prove this lemma, we note two general inequalities. First, note that for a fixed transition dynamics, say P, the return decomposes linearly in the components of the reward as the expected return is linear in the reward function:

$$J(P, r_{\mathcal{M}_f}) = J(P, fr_{\mathcal{M}_1} + (1-f)r_{\mathcal{M}_2}) = fJ(P, r_{\mathcal{M}_1}) + (1-f)J(P, r_{\mathcal{M}_2}).$$

As a result, we can bound $J(P, r_{\mathcal{M}_f})$ using $J(P, r)$ for a new reward function r of the auxiliary MDP, $\mathcal{M}$, as follows

$$\begin{aligned}
J(P, r_{\mathcal{M}_f}) &= J(P, f r_{\mathcal{M}_1} + (1-f) r_{\mathcal{M}_2}) = J(P, r + f(r_{\mathcal{M}_1} - r) + (1-f)(r_{\mathcal{M}_2} - r) \\
&= J(P, r) + f J(P, r_{\mathcal{M}_1} - r) + (1-f) J(P, r_{\mathcal{M}_2} - r) \\
&= J(P, r) + \frac{f}{1-\gamma} \mathbb{E}_{\mathbf{s},\mathbf{a} \sim d^{\pi}_{\mathcal{M}}(\mathbf{s})\pi(\mathbf{a}|\mathbf{s})} \left[r_{\mathcal{M}_1}(\mathbf{s}, \mathbf{a}) - r(\mathbf{s}, \mathbf{a}) \right] \\
&+ \frac{1-f}{1-\gamma} \mathbb{E}_{\mathbf{s},\mathbf{a} \sim d^{\pi}_{\mathcal{M}}(\mathbf{s})\pi(\mathbf{a}|\mathbf{s})} \left[r_{\mathcal{M}_2}(\mathbf{s}, \mathbf{a}) - r(\mathbf{s}, \mathbf{a}) \right].
\end{aligned}$$

Second, note that for a given reward function, r, but a linear combination of dynamics, the following bound holds:

$$\begin{aligned}
J(P_{\mathcal{M}_f}, r) &= J(f P_{\mathcal{M}_1} + (1-f) P_{\mathcal{M}_2}, r) \\
&= J(P_{\mathcal{M}} + f(P_{\mathcal{M}_1} - P_{\mathcal{M}}) + (1-f)(P_{\mathcal{M}_2} - P_{\mathcal{M}}), r) \\
&= J(P_{\mathcal{M}}, r) - \frac{\gamma(1-f)}{1-\gamma} \mathbb{E}_{\mathbf{s},\mathbf{a} \sim d^{\pi}_{\mathcal{M}}(\mathbf{s})\pi(\mathbf{a}|\mathbf{s})} \left[\left(P^{\pi}_{\mathcal{M}_2} - P^{\pi}_{\mathcal{M}} \right) Q^{\pi}_{\mathcal{M}} \right] \\
&- \frac{\gamma f}{1-\gamma} \mathbb{E}_{\mathbf{s},\mathbf{a} \sim d^{\pi}_{\mathcal{M}}(\mathbf{s})\pi(\mathbf{a}|\mathbf{s})} \left[\left(P^{\pi}_{\mathcal{M}} - P^{\pi}_{\mathcal{M}_1} \right) Q^{\pi}_{\mathcal{M}} \right] \\
&\in \Bigg[J(P_{\mathcal{M}}, r) \pm \Bigg(\frac{\gamma f}{(1-\gamma)} \left| \mathbb{E}_{\mathbf{s},\mathbf{a} \sim d^{\pi}_{\mathcal{M}}(\mathbf{s})\pi(\mathbf{a}|\mathbf{s})} \left[\left(P^{\pi}_{\mathcal{M}} - P^{\pi}_{\mathcal{M}_1} \right) Q^{\pi}_{\mathcal{M}} \right] \right| \\
&+ \frac{2\gamma(1-f) R_{\max}}{(1-\gamma)^2} D(P_{\mathcal{M}_2}, P_{\mathcal{M}}) \Bigg) \Bigg].
\end{aligned}$$

To observe the third equality, we utilize the result on the difference between returns of a policy π on two different MDPs, $P_{\mathcal{M}_1}$ and $P_{\mathcal{M}_f}$ from Agarwal et al. [1] (Chapter 2, Lemma 2.2, Simulation Lemma), and additionally incorporate the auxiliary MDP $\mathcal{M}$ in the expression via addition and subtraction in the previous (second) step. In the fourth step, we finally bound one term that corresponds to the learned model via the total-variation divergence $D(P_{\mathcal{M}_2}, P_{\mathcal{M}})$ and the other term corresponding to the empirical MDP $\overline{\mathcal{M}}$ is left in its expectation form to be bounded later.

Using the above bounds on return for reward-mixtures and dynamics-mixtures, proving this lemma is straightforward:

$$\begin{aligned}
&J(\mathcal{M}_1, \mathcal{M}_2, f, \pi) := J(P_{\mathcal{M}_f}, f r_{\mathcal{M}_1} + (1-f) r_{\mathcal{M}_2}) = J(f P_{\mathcal{M}_1} + (1-f) P_{\mathcal{M}_2}, r_{\mathcal{M}_f}) \\
&\in \Big[J(P_{\mathcal{M}_f}, r_{\mathcal{M}}) \pm \\
&\underbrace{\left(\frac{f}{1-\gamma} \mathbb{E}_{\mathbf{s},\mathbf{a} \sim d^{\pi}_{\mathcal{M}}\pi} [|r_{\mathcal{M}_1}(\mathbf{s}, \mathbf{a}) - r_{\mathcal{M}}(\mathbf{s}, \mathbf{a})|] + \frac{1-f}{1-\gamma} \mathbb{E}_{\mathbf{s},\mathbf{a} \sim d^{\pi}_{\mathcal{M}}\pi} [|r_{\mathcal{M}_2}(\mathbf{s}, \mathbf{a}) - r_{\mathcal{M}}(\mathbf{s}, \mathbf{a})|] \right)}_{:= \Delta_R} \Big],
\end{aligned}$$

where the second step holds via linear decomposition of the return of π in $\mathcal{M}_f$ with respect to the reward interpolation, and bounding the terms that appear in the reward difference. For convenience, we refer to these offset terms due to the reward as Δ_R. For the final part of this proof, we bound $J(P_{\mathcal{M}_f}, r_{\mathcal{M}})$ in terms of the return on the actual MDP, $J(P_{\mathcal{M}}, r_{\mathcal{M}})$, using the inequality proved above that provides intervals for mixture dynamics but a fixed reward function. Thus, the overall bound is given by $J(\pi, \mathcal{M}_f) \in [J(\pi, \mathcal{M}) - \alpha, J(\pi, \mathcal{M}) + \alpha]$, where α is given by:

$$\alpha = \frac{2\gamma(1-f)}{(1-\gamma)^2} R_{\max} D\left(P_{\mathcal{M}_2}, P_{\mathcal{M}}\right) + \frac{\gamma f}{1-\gamma} \left| \mathbb{E}_{d^{\pi}_{\mathcal{M}}\pi} \left[\left(P^{\pi}_{\mathcal{M}} - P^{\pi}_{\mathcal{M}_1} \right) Q^{\pi}_{\mathcal{M}} \right] \right| + \Delta_R. \tag{23}$$

This concludes the proof of this lemma. □

Finally, we prove Theorem 4.3 that shows how policy optimization with respect to $\hat{J}(f, \pi)$ affects the performance in the actual MD by using Equation 21 and building on the analysis of pure model-free algorithms from Kumar et al. [29]. We restate a more complete statement of the theorem below and present the constants at the end of the proof.

Theorem 2 (Formal version of Proposition 4.3)**.** *Let $\hat{\pi}_{out}(\mathbf{a}|\mathbf{s})$ be the policy obtained by COMBO. Assume $\nu(\rho^{\pi_{out}}, f) - \nu(\rho^{\beta}, f) \geq C$ for some constant $C > 0$. Then, the policy $\pi_{out}(\mathbf{a}|\mathbf{s})$ is a ζ-safe policy improvement over π_β in the actual MDP $\mathcal{M}$, i.e., $J(\pi_{out}, \mathcal{M}) \geq J(\pi_\beta, \mathcal{M}) - \zeta$, with probability at least $1 - \delta$, where ζ is given by (where $\rho^{\beta}(\mathbf{s}, \mathbf{a}) := d^{\pi_\beta}_{\widehat{\mathcal{M}}}(\mathbf{s}, \mathbf{a})$):*

$$\mathcal{O}\left(\frac{\gamma f}{(1-\gamma)^2}\right)\left[\mathbb{E}_{\mathbf{s}\sim d^{\pi_{out}}_{\mathcal{M}}}\left[\sqrt{\frac{|\mathcal{A}|}{|\mathcal{D}(\mathbf{s})|}(\mathrm{D}_{CQL}(\pi_{out}, \pi_\beta) + 1)}\right]\right] + \mathcal{O}\left(\frac{\gamma(1-f)}{(1-\gamma)^2}\right)\mathrm{D}_{\mathrm{TV}}(P_{\mathcal{M}}, P_{\widehat{\mathcal{M}}}) - \beta\frac{C}{(1-\gamma)}.$$

Proof. We first note that since policy improvement is not being performed in the same MDP, $\mathcal{M}$ as the f-interpolant MDP, $\mathcal{M}_f$, we need to upper and lower bound the amount of improvement occurring in the actual MDP due to the f-interpolant MDP. As a result our first is to relate $J(\pi, \mathcal{M})$ and $J(\pi, \mathcal{M}_f) := J(\overline{\mathcal{M}}, \widehat{\mathcal{M}}, f, \pi)$ for any given policy π.

Step 1: Bounding the return in the actual MDP due to optimization in the f-interpolant MDP. By directly applying Lemma A.2 stated and proved previously, we obtain the following upper and lower-bounds on the return of a policy π:

$$J(\overline{\mathcal{M}}, \widehat{\mathcal{M}}, f, \pi) \in \left[J(\pi, \mathcal{M}) - \alpha,\ \ J(\pi, \mathcal{M}) + \alpha\right],$$

where α is shown in Equation 22. As a result, we just need to bound the terms appearing the expression of α to obtain a bound on the return differences. We first note that the terms in the expression for α are of two types: **(1)** terms that depend only on the reward function differences (captured in Δ_R in Equation 23), and **(2)** terms that depend on the dynamics (the other two terms in Equation 23).

To bound Δ_R, we simply appeal to concentration inequalities on reward (Assumption A1), and bound Δ_R as:

$$\begin{aligned}\Delta_R &:= \frac{f}{1-\gamma}\mathbb{E}_{\mathbf{s},\mathbf{a}\sim d^{\pi}_{\mathcal{M}}\pi}[|r_{\mathcal{M}_1}(\mathbf{s}, \mathbf{a}) - r_{\mathcal{M}}(\mathbf{s}, \mathbf{a})|] + \frac{1-f}{1-\gamma}\mathbb{E}_{\mathbf{s},\mathbf{a}\sim d^{\pi}_{\mathcal{M}}\pi}[|r_{\mathcal{M}_2}(\mathbf{s}, \mathbf{a}) - r_{\mathcal{M}}(\mathbf{s}, \mathbf{a})|] \\ &\leq \frac{C_{r,\delta}}{1-\gamma}\mathbb{E}_{\mathbf{s},\mathbf{a}\sim d^{\pi}_{\mathcal{M}}\pi}\left[\frac{1}{\sqrt{D(\mathbf{s}, \mathbf{a})}}\right] + \frac{1}{1-\gamma}||R_{\mathcal{M}} - R_{\widehat{\mathcal{M}}}|| := \Delta^u_R.\end{aligned}$$

Note that both of these terms are of the order of $\mathcal{O}(1/(1-\gamma))$ and hence they don't figure in the informal bound in Theorem 4.3 in the main text, as these are dominated by terms that grow quadratically with the horizon. To bound the remaining terms in the expression for α, we utilize a result directly from Kumar et al. [29] for the empirical MDP, $\overline{\mathcal{M}}$, which holds for any policy $\pi(\mathbf{a}|\mathbf{s})$, as shown below.

$$\begin{aligned}&\frac{\gamma}{(1-\gamma)}\left|\mathbb{E}_{\mathbf{s},\mathbf{a}\sim d^{\pi}_{\mathcal{M}}(\mathbf{s})\pi(\mathbf{a}|\mathbf{s})}\left[\left(P^{\pi}_{\mathcal{M}} - P^{\pi}_{\mathcal{M}_1}\right)Q^{\pi}_{\mathcal{M}}\right]\right| \\ &\leq \frac{2\gamma R_{\max} C_{P,\delta}}{(1-\gamma)^2}\mathbb{E}_{\mathbf{s}\sim d^{\pi}_{\mathcal{M}}(\mathbf{s})}\left[\frac{\sqrt{|\mathcal{A}|}}{\sqrt{|\mathcal{D}(\mathbf{s})|}}\sqrt{D_{\mathrm{CQL}}(\pi, \pi_\beta)(\mathbf{s}) + 1}\right].\end{aligned}$$

Step 2: Incorporate policy improvement in the f-inrerpolant MDP. Now we incorporate the improvement of policy π_{out} over the policy π_β on a weighted mixture of $\widehat{\mathcal{M}}$ and $\overline{\mathcal{M}}$. In what follows, we derive a lower-bound on this improvement by using the fact that policy π_{out} is obtained by maximizing $\hat{J}(f, \pi)$ from Equation 21. As a direct consequence of Equation 21, we note that

$$\hat{J}(f, \pi_{\text{out}}) = J(\overline{\mathcal{M}}, \widehat{\mathcal{M}}, f, \pi_{\text{out}}) - \beta\frac{\nu(\rho^{\pi}, f)}{1-\gamma} \geq \hat{J}(f, \pi_\beta) = J(\overline{\mathcal{M}}, \widehat{\mathcal{M}}, f, \pi_\beta) - \beta\frac{\nu(\rho^{\beta}, f)}{1-\gamma} \quad (24)$$

Following **Step 1**, we will use the upper bound on $J(\overline{\mathcal{M}}, \widehat{\mathcal{M}}, f, \pi)$ for policy $\pi = \pi_{\text{out}}$ and a lower-bound on $J(\overline{\mathcal{M}}, \widehat{\mathcal{M}}, f, \pi)$ for policy $\pi = \pi_\beta$ and obtain the following inequality:

$$J(\pi_{\text{out}}, \mathcal{M}) - \beta \frac{\nu(\rho^\pi, f)}{1-\gamma} \geq \Big\{ J(\pi_\beta, \mathcal{M}) - \beta \frac{\nu(\rho^\beta, f)}{1-\gamma} - \frac{4\gamma(1-f)R_{\max}}{(1-\gamma)^2} D(P_{\mathcal{M}}, P_{\widehat{\mathcal{M}}}) \\ \underbrace{- \frac{2\gamma f}{(1-\gamma)} \left| \mathbb{E}_{d^{\pi_{\text{out}}}_{\mathcal{M}}} \left[\left(P^{\pi_{\text{out}}}_{\mathcal{M}} - P^{\pi_{\text{out}}}_{\overline{\mathcal{M}}} \right) Q^{\pi_{\text{out}}}_{\mathcal{M}} \right] \right|}_{:=(*)} \\ \underbrace{- \frac{4\gamma R_{\max} C_{P,\delta} f}{(1-\gamma)^2} \mathbb{E}_{\mathbf{s} \sim d^{\pi_\beta}_{\mathcal{M}}} \left[\sqrt{\frac{|\mathcal{A}|}{|\mathcal{D}(\mathbf{s})|}} \right]}_{:=(\wedge)} - \Delta^u_R \Big\}.$$

The term marked by $(*)$ in the above expression can be upper bounded by the concentration properties of the dynamics as done in Step 1 in this proof:

$$(*) \leq \frac{4\gamma f C_{P,\delta} R_{\max}}{(1-\gamma)^2} \mathbb{E}_{\mathbf{s} \sim d^{\pi_{\text{out}}}_{\mathcal{M}}(\mathbf{s})} \left[\frac{\sqrt{|\mathcal{A}|}}{\sqrt{|\mathcal{D}(\mathbf{s})|}} \sqrt{D_{\text{CQL}}(\pi_{\text{out}}, \pi_\beta)(\mathbf{s}) + 1} \right]. \tag{25}$$

Finally, using Equation 25, we can lower-bound the policy return difference as:

$$J(\pi_{\text{out}}, \mathcal{M}) - J(\pi_\beta, \mathcal{M}) \geq \beta \frac{\nu(\rho^\pi, f)}{1-\gamma} - \beta \frac{\nu(\rho^\beta, f)}{1-\gamma} - \frac{4\gamma(1-f)R_{\max}}{(1-\gamma)^2} D(P_{\mathcal{M}}, P_{\widehat{\mathcal{M}}}) - (*) - \Delta^u_R \\ \geq \beta \frac{C}{1-\gamma} - \frac{4\gamma(1-f)R_{\max}}{(1-\gamma)^2} D(P_{\mathcal{M}}, P_{\widehat{\mathcal{M}}}) - (*) - \Delta^u_R.$$

Plugging the bounds for terms (a), (b) and (c) in the expression for ζ where $J(\pi_{\text{out}}, \mathcal{M}) - J(\pi_\beta, \mathcal{M}) \geq \zeta$, we obtain:

$$\zeta = \left(\frac{4f\gamma R_{\max} C_{P,\delta}}{(1-\gamma)^2} \right) \mathbb{E}_{\mathbf{s} \sim d^{\pi_{\text{out}}}_{\mathcal{M}}(\mathbf{s})} \left[\frac{\sqrt{|\mathcal{A}|}}{\sqrt{|\mathcal{D}(\mathbf{s})|}} \sqrt{D_{\text{CQL}}(\pi_{\text{out}}, \pi_\beta)(\mathbf{s}) + 1} \right] + (\wedge) - \Delta^u_R \\ + \frac{4(1-f)\gamma R_{\max}}{(1-\gamma)^2} D(P_{\mathcal{M}}, P_{\widehat{\mathcal{M}}}) - \beta \frac{C}{1-\gamma}. \tag{26}$$

□

Remark 3 (Interpretation of Proposition 4.3). *Now we will interpret the theoretical expression for ζ in Equation 26, and discuss the scenarios when it is* negative. *When the expression for ζ is negative, the policy π_{out} is an improvement over π_β in the original MDP, $\mathcal{M}$.*

- *We first discuss if the assumption of $\nu(\rho^{\pi_{out}}, f) - \nu(\rho^\beta, f) \geq C > 0$ is reasonable in practice. Note that we have never used the fact that the learned model $P_{\widehat{\mathcal{M}}}$ is close to the actual MDP, $P_{\mathcal{M}}$ on the states visited by the behavior policy π_β in our analysis. We will use this fact now: in practical scenarios, $\nu(\rho^\beta, f)$ is expected to be smaller than $\nu(\rho^\pi, f)$, since $\nu(\rho^\beta, f)$ is directly controlled by the difference and density ratio of $\rho^\beta(\mathbf{s}, \mathbf{a})$ and $d(\mathbf{s}, \mathbf{a})$: $\nu(\rho^\beta, f) \leq \nu(\rho^\beta, f = 1) = \sum_{\mathbf{s},\mathbf{a}} d^{\pi_\beta}_{\widehat{\mathcal{M}}}(\mathbf{s}, \mathbf{a}) \left(d^{\pi_\beta}_{\widehat{\mathcal{M}}}(\mathbf{s}, \mathbf{a}) / d^{\pi_\beta}_{\overline{\mathcal{M}}}(\mathbf{s}, \mathbf{a}) - 1 \right)^2$ by Lemma A.1 which is expected to be small for the behavior policy π_β in cases when the behavior policy marginal in the empirical MDP, $d^{\pi_\beta}_{\overline{\mathcal{M}}}(\mathbf{s}, \mathbf{a})$, is broad. This is a direct consequence of the fact that the learned dynamics integrated with the policy under the learned model: $P^{\pi_\beta}_{\widehat{\mathcal{M}}}$ is closer to its counterpart in the empirical MDP: $P^{\pi_\beta}_{\overline{\mathcal{M}}}$ for π_β. Note that this is not true for any other policy besides the behavior policy that performs several counterfactual actions in a rollout and deviates from the data. For such a learned policy π, we incur an extra error which depends on the importance ratio of policy densities, compounded over the horizon and manifests as the D_{CQL} term (similar to Equation 25, or Lemma D.4.1 in Kumar et al. [29]). Thus, in practice, we argue that we are interested in situations where the assumption $\nu(\rho^{\pi_{out}}, f) - \nu(\rho^\beta, f) \geq C > 0$ holds, in which case by increasing β, we can make the expression for ζ in Equation 26 negative, allowing for policy improvement.*

- *In addition, note that when f is close to 1, the bound reverts to a standard model-free policy improvement bound and when f is close to 0, the bound reverts to a typical model-based policy improvement bound. In scenarios with high sampling error (i.e. smaller $|\mathcal{D}(\mathbf{s})|$), if we can learn a good model, i.e., $D(P_{\mathcal{M}}, P_{\widehat{\mathcal{M}}})$ is small, we can attain policy improvement better than model-free methods by relying on the learned model by setting f closer to 0. A similar argument can be made in reverse for handling cases when learning an accurate dynamics model is hard.*

B Experimental details

In this section, we include all details of our empirical evaluations of COMBO.

B.1 Practical algorithm implementation details

Model training. In the setting where the observation space is low-dimensional, as mentioned in Section 3, we represent the model as a probabilistic neural network that outputs a Gaussian distribution over the next state and reward given the current state and action:

$$\widehat{T}_\theta(\mathbf{s}_{t+1}, r|\mathbf{s}, \mathbf{a}) = \mathcal{N}(\mu_\theta(\mathbf{s}_t, \mathbf{a}_t), \Sigma_\theta(\mathbf{s}_t, \mathbf{a}_t)).$$

We train an ensemble of 7 such dynamics models following [20] and pick the best 5 models based on the validation prediction error on a held-out set that contains 1000 transitions in the offline dataset $\mathcal{D}$. During model rollouts, we randomly pick one dynamics model from the best 5 models. Each model in the ensemble is represented as a 4-layer feedforward neural network with 200 hidden units. For the generalization experiments in Section 5.1, we additionally use a two-head architecture to output the mean and variance after the last hidden layer following [67].

In the image-based setting, we follow Rafailov et al. [48] and use a variational model with the following components:

$$\begin{aligned}
&\text{Image encoder:} && \mathbf{h}_t = E_\theta(\mathbf{o}_t) \\
&\text{Inference model:} && \mathbf{s}_t \sim q_\theta(\mathbf{s}_t|\mathbf{h}_t, \mathbf{s}_{t-1}, \mathbf{a}_{t-1}) \\
&\text{Latent transition model:} && \mathbf{s}_t \sim \widehat{T}_\theta(\mathbf{s}_t|\mathbf{s}_{t-1}, \mathbf{a}_{t-1}) \\
&\text{Reward predictor:} && r_t \sim p_\theta(r_t|\mathbf{s}_t) \\
&\text{Image decoder:} && \mathbf{o}_t \sim D_\theta(\mathbf{o}_t|\mathbf{s}_t).
\end{aligned} \tag{27}$$

We train the model using the evidence lower bound:

$$\max_\theta \sum_{\tau=0}^{T-1} \left[\mathbb{E}_{q_\theta}[\log D_\theta(\mathbf{o}_{\tau+1}|\mathbf{s}_{\tau+1})]\right] - \mathbb{E}_{q_\theta}\left[D_{KL}[q_\theta(\mathbf{o}_{\tau+1}, \mathbf{s}_{\tau+1}|\mathbf{s}_\tau, \mathbf{a}_\tau)\|\widehat{T}_{\theta_\tau}(\mathbf{s}_{\tau+1}, a_{\tau+1})]\right]$$

At each step τ we sample a latent forward model $\widehat{T}_{\theta_\tau}$ from a fixed set of K models $[\widehat{T}_{\theta_1}, \ldots, \widehat{T}_{\theta_K}]$. For the encoder E_θ we use a convolutional neural network with kernel size 4 and stride 2. For the Walker environment we use 4 layers, while the Door Opening task has 5 layers. The D_θ is a transposed convolutional network with stride 2 and kernel sizes $[5, 5, 6, 6]$ and $[5, 5, 5, 6, 6]$ respectively. The inference network has a two-level structure similar to Hafner et al. [18] with a deterministic path using a GRU cell with 256 units and a stochastic path implemented as a conditional diagonal Gaussian with 128 units. We only train an ensemble of stochastic forward models, which are also implemented as conditional diagonal Gaussians.

Policy Optimization. We sample a batch size of 256 transitions for the critic and policy learning. We set $f = 0.5$, which means we sample 50% of the batch of transitions from $\mathcal{D}$ and another 50% from $\mathcal{D}_{\text{model}}$. The equal split between the offline data and the model rollouts strikes the balance between conservatism and generalization in our experiments as shown in our experimental results in Section 5. We represent the Q-networks and policy as 3-layer feedforward neural networks with 256 hidden units.

For the choice of $\rho(\mathbf{s}, \mathbf{a})$ in Equation 2, we can obtain the Q-values that lower-bound the true value of the learned policy π by setting $\rho(\mathbf{s}, \mathbf{a}) = d^{\pi}_{\widehat{\mathcal{M}}}(\mathbf{s})\pi(\mathbf{a}|\mathbf{s})$. However, as discussed in [29], computing π by alternating the full off-policy evaluation for the policy $\hat{\pi}^k$ at each iteration k and one step of policy improvement is computationally expensive. Instead, following [29], we pick a particular distribution $\psi(\mathbf{a}|\mathbf{s})$ that approximates the the policy that maximizes the Q-function at the current iteration and set $\rho(\mathbf{s}, \mathbf{a}) = d^{\pi}_{\widehat{\mathcal{M}}}(\mathbf{s})\psi(\mathbf{a}|\mathbf{s})$. We formulate the new objective as follows:

$$\hat{Q}^{k+1} \leftarrow \arg\min_{Q} \beta \left(\mathbb{E}_{\mathbf{s} \sim d^{\pi}_{\widehat{\mathcal{M}}}(\mathbf{s}), \mathbf{a} \sim \psi(\mathbf{a}|\mathbf{s})}[Q(\mathbf{s}, \mathbf{a})] - \mathbb{E}_{\mathbf{s}, \mathbf{a} \sim \mathcal{D}}[Q(\mathbf{s}, \mathbf{a})] \right) + \frac{1}{2} \mathbb{E}_{\mathbf{s}, \mathbf{a}, \mathbf{s}' \sim d_f} \left[\left(Q(\mathbf{s}, \mathbf{a}) - \widehat{\mathcal{B}}^{\pi} \hat{Q}^k(\mathbf{s}, \mathbf{a}) \right)^2 \right] + \mathcal{R}(\psi), \tag{28}$$

where $\mathcal{R}(\psi)$ is a regularizer on ψ. In practice, we pick $\mathcal{R}(\psi)$ to be the $-D_{\text{KL}}(\psi(\mathbf{a}|\mathbf{s})\|\text{Unif}(\mathbf{a}))$ and under such a regularization, the first term in Equation 28 corresponds to computing softmax of the Q-values at any state $\mathbf{s}$ as follows:

$$\hat{Q}^{k+1} \leftarrow \arg\min_{Q} \max_{\psi} \beta \left(\mathbb{E}_{\mathbf{s} \sim d^{\pi}_{\widehat{\mathcal{M}}}(\mathbf{s})} \left[\log \sum_{\mathbf{a}} Q(\mathbf{s}, \mathbf{a}) \right] - \mathbb{E}_{\mathbf{s}, \mathbf{a} \sim \mathcal{D}}[Q(\mathbf{s}, \mathbf{a})] \right) + \frac{1}{2} \mathbb{E}_{\mathbf{s}, \mathbf{a}, \mathbf{s}' \sim d_f} \left[\left(Q(\mathbf{s}, \mathbf{a}) - \widehat{\mathcal{B}}^{\pi} \hat{Q}^k(\mathbf{s}, \mathbf{a}) \right)^2 \right]. \tag{29}$$

We estimate the `log-sum-exp` term in Equation 29 by sampling 10 actions at every state $\mathbf{s}$ in the batch from a uniform policy $\text{Unif}(\mathbf{a})$ and the current learned policy $\pi(\mathbf{a}|\mathbf{s})$ with importance sampling following [29].

B.2 Hyperparameter Selection

In this section, we discuss the hyperparameters that we use for COMBO. In the D4RL and generalization experiments, our method are built upon the implementation of MOPO provided at: https://github.com/tianheyu927/mopo. The hyperparameters used in COMBO that relates to the backbone RL algorithm SAC such as twin Q-functions and number of gradient steps follow from those used in MOPO with the exception of smaller critic and policy learning rates, which we will discuss below. In the image-based domains, COMBO is built upon LOMPO without any changes to the parameters used there. For the evaluation of COMBO, we follow the evaluation protocol in D4RL [12] and a variety of prior offline RL works [29, 67, 26] and report the normalized score of the smooth undiscounted averaged return over 3 random seeds for all environments except `sawyer-door-close` and `sawyer-door` where we report the average success rate over 3 random seeds. As mentioned in Section 3, we use the regularization objective in Eq. 2 to select the hyperparameter from a range of pre-specified candidates in a fully offline manner, unlike prior model-based offline RL schemes such as [67] and [26] that similar hyperparameters as COMBO and tune them manually based on policy performance obtained via online rollouts.

We now list the additional hyperparameters as follows.

- **Rollout length h.** We perform a short-horizon model rollouts in COMBO similar to Yu et al. [67] and Rafailov et al. [48]. For the D4RL experiments and generalization experiments, we followed the defaults used in MOPO and used $h = 1$ for walker2d and `sawyer-door-close`, $h = 5$ for hopper, halfcheetah and `halfcheetah-jump`, and $h = 25$ for `ant-angle`. In the image-based domain we used rollout length of $h = 5$ for both the the `walker-walk` and `sawyer-door-open` environments following the same hyperparameters used in Rafailov et al. [48].
- **Q-function and policy learning rates.** On state-based domains, we apply our automatic selection rule to the set $\{1e-4, 3e-4\}$ for the Q-function learning rate and the set $\{1e-5, 3e-5, 1e-4\}$ for the policy learning rate. We found that $3e-4$ for the Q-function learning rate (also used previously in Kumar et al. [29]) and $1e-4$ for the policy learning rate (also recommended previously in Kumar et al. [29] for gym domains) work well for almost all domains except that on walker2d where a smaller Q-function learning rate of $1e-4$ and a correspondingly smaller policy learning rate of $1e-5$ works the best according to our automatic hyperparameter selection scheme. In the image-based domains, we followed the defaults from prior work [48] and used $3e-4$ for both the policy and Q-function.

- **Conservative coefficient β.** We use our hyperparameter selection rule to select the right β from the set $\{0.5, 1.0, 5.0\}$ for β, which correspond to low conservatism, medium conservatism and high conservatism. A larger β would be desirable in more narrow dataset distributions with lower-coverage of the state-action space that propagates error in a backup whereas a smaller β is desirable with diverse dataset distributions. On the D4RL experiments, we found that $\beta = 0.5$ works well for halfcheetah agnostic of dataset quality, while on hopper and walker2d, we found that the more "narrow" dataset distributions: medium and medium-expert datasets work best with larger $\beta = 5.0$ whereas more "diverse" dataset distributions: random and medium-replay datasets work best with smaller $\beta = 0.5$ which is consistent with the intuition. On generalization experiments, $\beta = 1.0$ works best for all environments. In the image-domains we use $\beta = 0.5$ for the medium-replay `walker-walk` task and and $\beta = 1.0$ for all other domains, which again is in accordance with the impact of β on performance.
- **Choice of $\rho(\mathbf{s}, \mathbf{a})$.** We first decouple $\rho(\mathbf{s}, \mathbf{a}) = \rho(\mathbf{s})\rho(\mathbf{a}|\mathbf{s})$ for convenience. As discussed in Appendix B.1, we use $\rho(\mathbf{a}|\mathbf{s})$ as the soft-maximum of the Q-values and estimated with `log-sum-exp`. For $\rho(\mathbf{s})$, we apply the automatic hyperparameter selection rule to the set $\{d^{\pi}_{\widehat{\mathcal{M}}}, \rho(\mathbf{s}) = d_f\}$. We found that $d^{\pi}_{\widehat{\mathcal{M}}}$ works better the hopper task in D4RL while d_f is better for the rest of the environments. For the remaining domains, we found $\rho(\mathbf{s}) = d_f$ works well.
- **Choice of $\mu(\mathbf{a}|\mathbf{s})$.** For the rollout policy μ, we use our automatic selection rule on the set $\{\text{Unif}(\mathbf{a}), \pi(\mathbf{a}|\mathbf{s})\}$, i.e. the set that contains a random policy and a current learned policy. We found that $\mu(\mathbf{a}|\mathbf{s}) = \text{Unif}(\mathbf{a})$ works well on the hopper task in D4RL and also in the `ant-angle` generalization experiment. For the remaining state-based environments, we discovered that $\mu(\mathbf{a}|\mathbf{s}) = \pi(\mathbf{a}|\mathbf{s})$ excels. In the image-based domain, we found that $\mu(\mathbf{a}|\mathbf{s}) = \text{Unif}(\mathbf{a})$ works well in the `walker-walk` domain and $\mu(\mathbf{a}|\mathbf{s}) = \pi(\mathbf{a}|\mathbf{s})$ is better for the `sawyer-door` environment. We observed that $\mu(\mathbf{a}|\mathbf{s}) = \text{Unif}(\mathbf{a})$ behaves less conservatively and is suitable to tasks where dynamics models can be learned fairly precisely.
- **Choice of f.** For the ratio between model rollouts and offline data f, we input the set $\{0.5, 0.8\}$ to our automatic hyperparameter selection rule to figure out the best f on each domain. We found that $f = 0.8$ works well on the medium and medium-expert in the walker2d task in D4RL. For the remaining environments, we find $f = 0.5$ works well.

We also provide additional experimental results on how our automatic hyperparameter selection rule selects hyperparameters. As shown in Table 4, 5, 6 and 7, our automatic hyperparameter selection rule is able to pick the hyperparameters β, $\mu(\mathbf{a}|\mathbf{s})$, $\rho(\mathbf{s})$ and f and that correspond to the best policy performance based on the regularization value.

Task	$\beta = 0.5$ performance	$\beta = 0.5$ regularizer value	$\beta = 5.0$ performance	$\beta = 5.0$ regularizer value
halfcheetah-medium	**54.2**	**-778.6**	40.8	-236.8
halfcheetah-medium-replay	**55.1**	**28.9**	9.3	283.9
halfcheetah-medium-expert	89.4	189.8	**90.0**	**6.5**
hopper-medium	75.0	-740.7	**97.2**	**-2035.9**
hopper-medium-replay	**89.5**	**37.7**	28.3	107.2
hopper-medium-expert	**111.1**	**-705.6**	75.3	-64.1
walker2d-medium	1.9	51.5	**81.9**	**-1991.2**
walker2d-medium-replay	**56.0**	**-157.9**	27.0	53.6
walker2d-medium-expert	10.3	-788.3	**103.3**	**-3891.4**

Table 4: We include our automatic hyperparameter selection rule of β on a set of representative D4RL environments. We show the policy performance (bold with the higher number) and the regularizer value (bold with the lower number). Lower regularizer value consistently corresponds to the higher policy return, suggesting the effectiveness of our automatic selection rule.

B.3 Details of generalization environments

For `halfcheetah-jump` and `ant-angle`, we follow the same environment used in MOPO. For `sawyer-door-close`, we train the `sawyer-door` environment in https://github.com/rlworkgroup/metaworld with dense rewards for opening the door until convergence. We collect 50000 transitions with half of the data collected by the final expert policy and a policy that reaches the performance of about half the expert level performance. We relabel the reward such that

Task	$\mu(\mathbf{a}\|\mathbf{s}) = \text{Unif}(\mathbf{a})$ performance	$\mu(\mathbf{a}\|\mathbf{s}) = \text{Unif}(\mathbf{a})$ regularizer value	$\mu(\mathbf{a}\|\mathbf{s}) = \pi(\mathbf{a}\|\mathbf{s})$ performance	$\mu(\mathbf{a}\|\mathbf{s}) = \pi(\mathbf{a}\|\mathbf{s})$ regularizer value
hopper-medium	**97.2**	**-2035.9**	52.6	-14.9
walker2d-medium	7.9	-106.8	**81.9**	**-1991.2**

Table 5: We include our automatic hyperparameter selection rule of $\mu(\mathbf{a}|\mathbf{s})$ on the medium datasets in the hopper and walker2d environments from D4RL. We follow the same convention defined in Table 4 and find that our automatic selection rule can effectively select μ offline.

Task	$\rho(\mathbf{s}) = d^{\pi}_{\hat{\mathcal{M}}}$ performance	$\rho(\mathbf{s}) = d^{\pi}_{\hat{\mathcal{M}}}$ regularizer value	$\rho(\mathbf{s}) = d_f$ performance	$\rho(\mathbf{s}) = d_f$ regularizer value
hopper-medium	**97.2**	**-2035.9**	56.0	-6.0
walker2d-medium	1.8	14617.4	**81.9**	**-1991.2**

Table 6: We include our automatic hyperparameter selection rule of $\rho(\mathbf{s})$ on the medium datasets in the hopper and walker2d environments from D4RL. We follow the same convention defined in Table 4 and find that our automatic selection rule can effectively select ρ offline.

the reward is 1 when the door is fully closed and 0 otherwise. Hence, the offline RL agent is required to learn the behavior that is different from the behavior policy in a sparse reward setting. We provide the datasets in the following anonymous link[1].

B.4 Details of image-based environments

Figure 3: Our image-based environments: The observations are 64×64 and 128×128 raw RGB images for the `walker-walk` and `sawyer-door` tasks respectively. The `sawyer-door-close` environment used in in Section 5.1 also uses the `sawyer-door` environment.

We visualize our image-based environments in Figure 3. We use the standard `walker-walk` environment from Tassa et al. [61] with 64×64 pixel observations and an action repeat of 2. Datasets were constructed the same way as Fu et al. [12] with 200 trajectories each. For the `sawyer-door` we use 128×128 pixel observations. The medium-expert dataset contains 1000 rollouts (with a rollout length of 50 steps) covering the state distribution from grasping the door handle to opening the door. The expert dataset contains 1000 trajectories samples from a fully trained (stochastic) policy. The data was obtained from the training process of a stochastic SAC policy using dense reward function as defined in Yu et al. [66]. However, we relabel the rewards, so an agent receives a reward of 1 when the door is fully open and 0 otherwise. This aims to evaluate offline-RL performance in a sparse-reward setting. All the datasets are from [48].

B.5 Computation Complexity

For the D4RL and generalization experiments, COMBO is trained on a single NVIDIA GeForce RTX 2080 Ti for one day. For the image-based experiments, we utilized a single NVIDIA GeForce RTX 2070. We trained the `walker-walk` tasks for a day and the `sawyer-door-open` tasks for about two days.

B.6 License of datasets

We acknowledge that all datasets used in this paper use the MIT license.

[1]The datasets of the generalization environments are available at the anonymous link: https://drive.google.com/file/d/1pn6dS5OgPQVp_ivGws-tmWdZoU7m_LvC/view?usp=sharing.

Task	$f = 0.5$ performance	$f = 0.5$ regularizer value	$f = 0.8$ performance	$f = 0.8$ regularizer value
hopper-medium	**97.2**	**-2035.9**	93.8	-21.3
walker2d-medium	70.9	-1707.0	**81.9**	**-1991.2**

Table 7: We include our automatic hyperparameter selection rule of f on the medium datasets in the hopper and walker2d environments from D4RL. We follow the same convention defined in Table 4 and find that our automatic selection rule can effectively select f offline.

Environment	**Batch Mean**	**Batch Max**	**COMBO (Ours)**	**CQL+MBPO**
halfcheetah-jump	-1022.6	1808.6	**5392.7**$\pm$575.5	4053.4$\pm$176.9
ant-angle	866.7	2311.9	**2764.8**$\pm$43.6	809.2$\pm$135.4
sawyer-door-close	5%	100%	**100**%$\pm$0.0%	62.7%$\pm$24.8%

Table 8: Comparison between COMBO and CQL+MBPO on tasks that require out-of-distribution generalization. Results are in average returns of `halfcheetah-jump` and `ant-angle` and average success rate of `sawyer-door-close`. All results are averaged over 6 random seeds, $\pm$ the 95%-confidence interval.

C Comparison to the Naive Combination of CQL and MBPO

In this section, we stress the distinction between COMBO and a direct combination of two previous methods CQL and MBPO (denoted as CQL + MBPO). CQL+MBPO performs Q-value regularization using CQL while expanding the offline data with MBPO-style model rollouts. While COMBO utilizes Q-value regularization similar to CQL, the effect is very different. CQL only penalizes the Q-value on unseen actions on the states observed in the dataset whereas COMBO penalizes Q-values on states generated by the learned model while maximizing Q values on state-action tuples in the dataset. Additionally, COMBO also utilizes MBPO-style model rollouts for also augmenting samples for training Q-functions.

To empirically demonstrate the consequences of this distinction, CQL + MBPO performs quite a bit worse than COMBO on generalization experiments (Section 5.1) as shown in Table 8. The results are averaged across 6 random seeds ($\pm$ denotes 95%-confidence interval of the various runs). This suggests that carefully considering the state distribution, as done in COMBO, is crucial.