Bridging Worlds in Reinforcement Learning with Model-Advantage

Anonymous Authors[1]

Abstract

Despite the breakthroughs achieved by Reinforcement Learning (RL) in recent years, RL agents often fail to perform well in unseen environments. This inability to *generalize* to new environments prevents their deployment in the real world. To help measure this gap in performance, we introduce *model-advantage* - a quantity similar to the well-known (policy) advantage function. First, we show relationships between the proposed model-advantage and generalization in RL — using which we provide guarantees on the gap in performance of an agent in new environments. Further, we conduct toy experiments to show that even a sub-optimal policy (learnt with minimal interactions with the target environment) can help *predict* if a training environment (say, a simulator) helps learn policies that generalize. We then show connections with Model Based RL.

1. Introduction

Reinforcement Learning (RL) has emerged as a promising learning paradigm owing to its successes in applications like strategy games (Mnih et al., 2015; Silver et al., 2016; 2017) and robotics (Levine et al., 2016; Gu et al., 2016). Due to the high cost of interacting with the real world – *e.g.* accidents in the case of physical robots or loss of revenue in recommendation systems – it is common to train RL agents in a more accessible environment. For example, if we want to train a self-driving agent to drive in Maine, a good starting point is to train using a simulator or in a different environment which is more accessible such as Nevada. However, an agent with good performance in this training environment may or may not achieve similar performance in the test environment (the *real world*). This *lack of generalization* of RL agents prevents its reliable deployment in the real world.

Intuitively, better performance can be expected when the training environment informs the agent of the trajectories it is likely to take in the target environment. Ideally, we would like that in both the training and target MDP, taking actions at states lead to the same transitions as well as rewards. To formalize this notion, we introduce *model-advantage* – a quantity similar to the well-known advantage function in RL. The standard advantage function – which we refer to as policy-advantage – evaluates the *advantage* of playing a particular action as opposed to the action of a reference policy. Similarly, we define model-advantage as the advantage of transitioning to a state as opposed to transitioning according to an MDP M, while acting according to some policy. Specifically, we are interested in the expected advantage of transitioning according to one MDP with respect to another one as reference, which allows us to evaluate the effectiveness of using one model in lieu of the other – much like how policy-advantage helps compare two different policies.

2. Model-Advantage

In this section, we formally introduce *model-advantage* and definitions associated with it. Recall that $\mathbb{A}^\pi(s,a)$ *i.e.* policy-advantage defined over states $s \in \mathcal{S}$ and actions $a \in \mathcal{A}$ measures the *utility* of taking action a as opposed to acting according to policy π. However, unlike policy-advantage that measures the difference in utility of taking an action, we are interested in knowing the utility difference of transitioning to a particular state, while following a single policy. Specifically, *model-advantage* denoted by $\boldsymbol{A}^\pi_M(s,s')$ compares the utility of moving to state s' and following the trajectory governed by model M as opposed to doing it from state s; both under policy π.

We define the model-dependent value function as follows, where we make explicit the MDP M (and hence, the transition function $\mathcal{P}_M$) used to generate trajectories.

$$V^\pi_M(s) = \mathbb{E}_{\rho^\pi_M}\left[\sum_{t=0}^{\infty} \mathcal{R}(s_t,a_t) \mid \pi, M, s_0 = s\right]$$

Here, ρ_π is the distribution of trajectories $(s_0, a_0, s_1, \dots)$, $s_0 \sim \mathcal{P}_0$, when acting according to policy π. Now, the intuition of model-advantage [1] is given by:

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

[1]A $Q(s,s')$ function can also be defined; however, it requires additional formalism not necessary for our exposition. See concurrent work (Edwards et al., 2020) for a detailed discussion.

$$\boldsymbol{A}^{\pi}_{M}(s,s') = \gamma\left[V^{\pi}_{M}(s') - \mathbb{E}_{s''\sim P_M(s,\pi)} V^{\pi}_{M}(s'')\right] \quad (1)$$

Analogous to policy-advantage that compares different policies in the same environment, model-advantage helps compare the same policy acting in two different environments. For such a comparison, we need to look at the quantity $\mathbb{E}_{s'\sim M'}\left[\boldsymbol{A}^{\pi}_{M}(s,s')\right]$ – the expected model-advantage (evaluated at π) when the next state s' is obtained from the MDP M'. We formalize this by the following *(model) performance difference* lemma. The proof resembles the proof by (Kakade & Langford, 2002) and is provided in Appendix A.

Lemma 2.1. *(Model Performance Difference Lemma) Let M and M' be two different MDPs. Further, define $\mathcal{R}^{\pi}(s) = \mathbb{E}_{a\sim\pi(\cdot|s)}[\mathcal{R}(s,a)]$ and $\mathcal{R}^{\pi}_{\epsilon}(s) = \mathcal{R}^{\pi}_{M}(s) - \mathcal{R}^{\pi}_{M'}(s)$. For any policy $\pi \in \Pi$ we have:*

$$J_M(\pi) - J_{M'}(\pi) = \mathbb{E}_{s\sim d_{M,\pi}}\left[\mathcal{R}_{\epsilon}(s)\right] + \frac{1}{1-\gamma}\mathbb{E}_{s\sim d_{M,\pi}}\mathbb{E}_{s'\sim\mathcal{P}_M(s'|s,\pi)}\left[\boldsymbol{A}^{\pi}_{M'}(s,s')\right] \quad (2)$$

Here, we use a model-dependent stationary state distribution $d_{M,\pi}(s)$ where the dynamics $\mathcal{P}_M$ are used, assuming a start state distribution $\mathcal{P}_0$. Compared to its policy counterpart the (model) performance difference lemma involves an additional reward error term $\mathbb{E}_{d_{M,\pi}}\mathcal{R}_{\epsilon}$ that vanishes when the two MDPs differ only in the transition probabilities.
We can also use the Bellman evaluation operator for policy π to obtain an equivalent formalization (see Appendix A):

$$J_M(\pi) = J_{M'}(\pi) + \frac{1}{1-\gamma}\mathbb{E}_{d_{M,\pi}}\underbrace{\left[\mathcal{T}^{\pi}_{M}V^{\pi}_{M} - \mathcal{T}^{\pi}_{M'}V^{\pi}_{M}\right]}_{\text{deviation error}} \quad (3)$$

The term $\mathcal{T}^{\pi}_{M}V^{\pi}_{M} - \mathcal{T}^{\pi}_{M'}V^{\pi}_{M}$, which we denote by $\delta^{\pi}_{M,M'}$ [2] represents the *deviation* in the value function when acted upon by Bellman operators corresponding to two different MDPs. This term is exactly equal to model-advantage when the reward functions of the two MDPs are the same. We can form an upper bound on the extrinsic error as:

$$\begin{aligned} J_M(\pi) - J_{M'}(\pi) &\leq \frac{1}{1-\gamma}\left\|\mathcal{T}^{\pi}_{M}V^{\pi} - \mathcal{T}^{\pi}_{M'}V^{\pi}\right\|_{\infty} \\ &\leq \frac{1}{1-\gamma}\left[\epsilon_R + \gamma\epsilon_P\left\|V^{\pi}\right\|_{\infty}\right] \end{aligned} \quad (4)$$

where the rewards and the dynamics themselves are individually bounded *i.e.* $\max_s \max_a |R_M(s,a) - R'_{M'}(s,a)| \leq \epsilon_R$ and $\max_s \max_a \|P_M(s,a) - P_{M'}(s,a)\|_1 \leq \epsilon_P$. Of course, note that $\|V^{\pi}\|_{\infty}$ is trivially bounded by $\frac{1}{1-\gamma}R_{\max}$, assuming rewards are bounded by $R_{\max}$.

[2] Optionally, when using the optimality operators corresponding to MDPs M and M', we drop the π and denote the deviation error as $\delta_{M,M'}(V(s))$.

3. Generalization in RL

An RL problem is characterized by an MDP M and is considered solved when a policy $\pi \in \Pi$ that maximizes the expected (discounted) return is found. However, in practice, the RL agent may then be deployed in a slightly different environment characterized by an MDP M'. Therefore, we are interested in how well an RL agent performs in an unseen environment – in other words, its ability to *generalize*.

Related Work. Studying and benchmarking generalization properties of RL agents via large-scale experiments has been the focus of many works in the recent years (Zhang et al., 2018a;b; Cobbe et al., 2018). Additionally, (Slaoui et al., 2019) derives a lemma comparable to lemma 2.1 for policies that are Lipschitz continuous over a set of state-representations. Importantly, (Wang et al., 2019a) formally define generalization gap which we adopt in this work (see eq. (5)) They give formal bounds on this gap in the setting of *reparametrizable RL* while making additional assumptions like Lipschitz continuity on value functions. While we do not require any such assumptions, it is important to note that it is not possible to guarantee tighter bounds without them.

3.1. Generalization Gap.

Let $J(\pi) := \mathbb{E}[\sum_t \mathcal{R}(s_t,a_t)]$ denote the cost function, where the stochasticity is due to the policy and transition dynamics; let $\hat{J}_n(\pi)$ denote its empirical estimate with n samples. Given an RL agent trained in MDP M with finite data, we are interested in its performance in a different MDP M'. We can formally write this *generalization gap* as:

$$\begin{aligned} \Phi &= \left|\hat{J}_n(\pi) - J'(\pi)\right| \\ &\leq \underbrace{\left|\hat{J}_n(\pi) - J(\pi)\right|}_{\text{intrinsic error}} + \underbrace{\left|J(\pi) - J'(\pi)\right|}_{\text{extrinsic error}} \end{aligned} \quad (5)$$

The generalization gap can be bounded with two different sources of error as indicated in eq. (5). Following (Wang et al., 2019a), we call them *intrinsic* error and *extrinsic* error to denote the error due to learning from finite samples and the error due to mismatch in training and deployment environments. The intrinsic error decreases, typically as $\mathcal{O}(1/\sqrt{n})$, with more samples; this is well-studied in RL literature (Kakade et al., 2003; Azar et al., 2012; Jin et al., 2018). The extrinsic error on the other hand is an artifact of training and deploying the RL agent in different environments and therefore, cannot be avoided.

When does π generalize? Observe that the extrinsic error is nothing but the difference in performance due to model mismatch. From lemma 2.1, we know that this is equal to the model-advantage, allowing us to both estimate and bound this error term. In other words, if the

model-advantage is bounded by ϵ (see eq. (4)) *i.e.*

$$|J_M(\pi_M) - J_{M'}(\pi_M)| \leq \epsilon$$

we can say that π_M, the policy learnt with experiences from MDP M achieves similar performance in the target MDP M'. As model of the "test" environment is not known, a reasonable estimate of the model-advantage can be obtained with enough samples – allowing one to *predict* the extent to which the policy performs in the novel environment.

How good is π really? However, note that the above generalization gap still does not provide the complete picture. Ideally, we would like the policy π_M to have performance comparable to $\pi^*_{M'}$, the optimal policy in the target MDP M' *i.e.*

$$\begin{aligned} &|J_{M'}(\pi_M) - J_{M'}(\pi^*_{M'})| \\ &\leq \underbrace{|J_{M'}(\pi_M) - J_M(\pi_M)|}_{\text{term-I}} + \underbrace{|J_M(\pi_M) - J_{M'}(\pi^*_{M'})|}_{\text{term-II}} \end{aligned} \quad (6)$$

It is easy to see that term-I is nothing but the extrinsic error in eq. (5) and is related to the model-advantage (evaluated under policy π_M) through lemma 2.1. Intuitively, this this term corresponds to the cost of transfering π_M learnt in the seen MDP M to the novel MDP M'.

In the rest of this section, we will bound term-II for specific instantiations of obtaining policy π_M – specifically, Value Iteration (VI) and Fitted Q-Iteration (FQI), with the former being a model-based and the latter, a model-free approach to solve MDPs.

3.2. Generalization with Value Iteration and Fitted Q iteration

Value Iteration. When the dynamics and the reward functions are known, Value Iteration (VI) and its variants are often employed to arrive at the optimal policy. VI is an iterative algorithm that applies the Bellman optimality operator $\mathcal{T}_M$ [3] at each step *i.e.* $V^{(n)} = \mathcal{T}_M V^{(n-1)}$. The obtained iterates converge to V^*_M , the value function of $\pi^\star_M$, asymptotically as $\mathcal{T}_M$ is a contraction in the infinity-norm. We can bound the difference in value from training on another MDP with the following theorem:

Theorem 3.1. *Let M, M' be two MDPs s.t. $\max_s \max_a |\mathcal{R}_M(s,a) - \mathcal{R}_{M'}(s,a)| \leq \epsilon_R$ and $\max_s \max_a \|p_M(s,a) - p_{M'}(s,a)\|_1 \leq \epsilon_P$. Let π_{n+1} be the policy obtained after n VI iterations on MDP M. Then we have,*

$$\left\|V^{\pi_{n+1}}_{M'} - V^\star_{M'}\right\|_\infty \leq \frac{1}{1-\gamma}\left[\gamma\epsilon^{(n)} + 2\epsilon_R + \frac{2\epsilon_P R_{max}}{1-\gamma}\right]$$

[3] Assume optimality operator by default if policy is not explicitly defined

We provide a similar bound when following a Fitted Q iteration (FQI) method, which is more effective when dealing with a large (or infinite) state space or unknown dynamics/reward functions. The statement and proof of the bound can be found in Appendix B.2.

3.3. Sim2Real: Is My Simulator Good?

The fundamental bottleneck preventing the usage of RL to train agents in the real-world is *exploration*. As the model of the environment is not available, finding the optimal policy not only requires exploring a large search space but is also *costly*. A common strategy to avoid this issue is to learn a *coarse* policy using a simulator and then *fine-tune* it upon deployment. But how does one build the simulator in the first place? It either requires considerable domain expertise or a large number of samples from the real-world, and we must know a priori that the simulator can *express* all variations feasible in the real world. We are left with the question: *Given a set of simulators, which one is likely to "generalize" best to the real-world?*

Predicting Generalization with Model-Advantage. Recall that (model) performance difference lemma lemma 2.1 allows us to compare two models given a policy. Given M, the simulator MDP and M', the real-world MDP and π_{exp}, an expert policy for M', we can write:

$$\left|J_M(\pi_{\text{exp}}) - J_{M'}(\pi_{\text{exp}})\right| = \left|\mathbb{E}_{(s,s')\sim M}\left[\boldsymbol{A}^{\pi_{\text{exp}}}_{M'}(s,s')\right]\right|$$

To compute the advantage function $\boldsymbol{A}^{\pi_{\text{exp}}}_{M'}$, the expert policy has to be executed in the real-world. Alternately, such an "expert" policy and its corresponding value function in MDP M' can be learnt by collecting a finite set of data from the real-world – for instance, by running FQI on the collected dataset [4]. After paying this one-time cost of interacting with the real-world, the model-advantage can be estimated in an inexpensive manner as it only requires sampling from the simulator.

Grid World Experiments. We consider the toy environment of FrozenLake available as part of OpenAI Gym [5] to illustrate the effectiveness of the proposed model-advantage term in evaluating simulators. We treat the original setting as M, the real-world MDP and then, corrupt the transition dynamics with various levels of random noise to obtain a set of "simulators" $\{M'_i\}_{i=1}^K$. We then run DQN (Mnih et al., 2013) that uses a single hidden-layer MLP to learn Q-values in M and obtain a sub-optimal "expert" Q-function by not running the training to completion. As can be seen from section 3.3, we see that even for Q-values far from optimal, the

[4] Note that convergence to optimal value-function is guaranteed only if the distribution used to sample states is exploratory. While the "tax" of exploration cannot be waived, the hope here is that a small amount if data is sufficient to learn a sub-optimal expert.

[5] https://gym.openai.com/

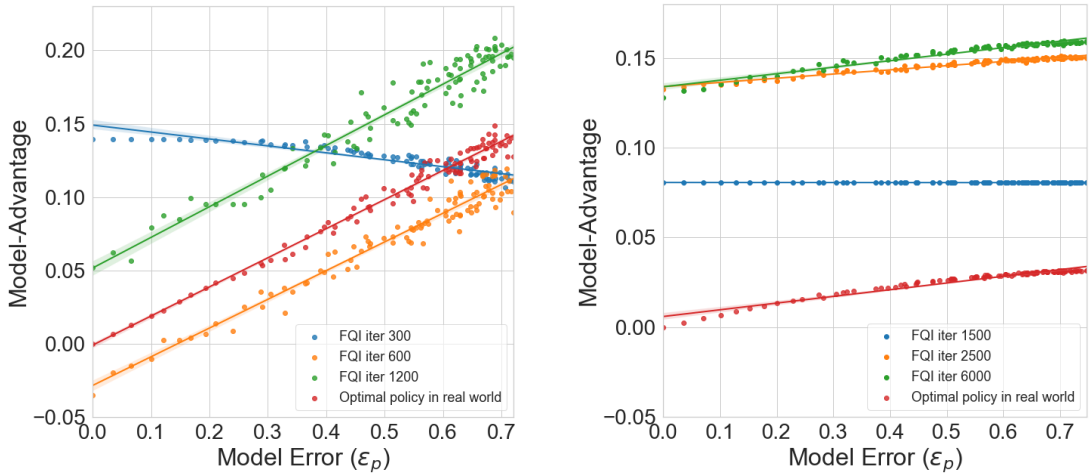

Figure 1. Generalization gap on sub-optimal real world policies on FrozenLake environments (FrozenLake 4x4 on the left and 8x8 on the right). Even a sub-optimal policy obtained with minimal interactions with the real-world is sufficient to use model-advantage and compare different *training* environments or simulators.

model-advantage increases with increasing modeling error ϵ_P – the same trend exhibited by the optimal Q-function in the real-world.

3.4. Connections to Model-Based RL

Model Based Reinforcement Learning (MBRL) is a dominant RL framework that first learns an *approximate* model of the real-world, and then runs standard planning algorithms like VI to find a suitable policy. Variants of the model performance difference lemma (see lemma 2.1) involving the approximated model have been derived in this context – for *e.g.* see (Kearns & Singh, 2002; Kakade et al., 2003) and concurrent works, (Rajeswaran et al., 2020; Kidambi et al., 2020; Xu et al., 2018). In this section, we introduce a novel perspective of MBRL and MBRL techniques.

The objective of MBRL is to construct a model that mimics the dynamics and reward functions of the real-world as accurately as possible. As seen from eq. (4), the model-advantage is upper bounded as a function of ϵ_R and ϵ_P, the error in reward and transition dynamics functions. Therefore, many previous works minimize this upper bound (Azar et al., 2012; 2013; 2017). The drawback is that when the upper bound is fairly loose, it leads to a "dynamics bottleneck"(Wang et al., 2019b; Lambert et al., 2020).

Directly Minimizing Model-Advantage. From lemma 2.1 and eq. (4), a tighter bound on the model-advantage can be minimized (Wang et al., 2019b; Lambert et al., 2020) –

$$\left|J_M(\pi) - J_{M'}(\pi)\right| = \left|\mathbb{E}_{(s,s')\sim M}\left[A^{\pi}_{M'}(s,s')\right]\right| \leq \frac{\epsilon_R}{1-\gamma} + \frac{\gamma \epsilon_P R_{\max}}{1-\gamma}$$

This is explored by the Value-Aware Model Learning framework (Farahmand, 2018a;b); specifically, they minimize the $\mathbb{E}\left|\delta^{\pi}_{(M,M')}\right|^2$ for a worst-case policy that seeks to maximize the deviation. The crux of MBRL algorithms is to assume access to the real-world – either through trajectories collected by running a policy or with a generative model of the MDP. These samples correspond to *expert* dynamics (or rewards) and can be learnt by using the Imitation Learning (IL) framework. This naturally leads to an algorithm that alternately updates a model to minimize the dynamics and reward error with imitation learning, and updates a policy to maximize reward and further collect samples from the true MDP. We defer the reader to Appendix C for guarantees of such an algorithm with MBRL and IL.

4. Conclusion

In this work, we proposed *model-advantage* that helps compare two models, similar to policy advantage that can be used to compare two policies. This term provides a theoretical framework for understanding generalization in RL – specifically, we provide formal guarantees on the generalization ability of policies learnt via Value Iteration (VI) and Fitted Q-Iteration (FQI). Further, we conduct toy experiments to show that even a sub-optimal policy, learnt from minimal interactions with the target environment, can help identify the training environment that facilitates maximum generalization. Finally, we discuss connections between model-advantage and Model Based Reinforcement Learning (MBRL), and formally establish connections between MBRL and Imitation Learning using the proposed model-advantage terms.

References

Azar, M. G., Munos, R., and Kappen, B. On the sample complexity of reinforcement learning with a generative model. *arXiv preprint arXiv:1206.6461*, 2012.

Azar, M. G., Munos, R., and Kappen, H. J. Minimax pac bounds on the sample complexity of reinforcement learning with a generative model. *Machine learning*, 91(3):325–349, 2013.

Azar, M. G., Osband, I., and Munos, R. Minimax regret bounds for reinforcement learning. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 263–272. JMLR. org, 2017.

Cobbe, K., Klimov, O., Hesse, C., Kim, T., and Schulman, J. Quantifying generalization in reinforcement learning. *arXiv preprint arXiv:1812.02341*, 2018.

Edwards, A. D., Sahni, H., Liu, R., Hung, J., Jain, A., Wang, R., Ecoffet, A., Miconi, T., Isbell, C., and Yosinski, J. Estimating q (s, s') with deep deterministic dynamics gradients. *arXiv preprint arXiv:2002.09505*, 2020.

Farahmand, A.-m. Iterative value-aware model learning. In *Advances in Neural Information Processing Systems*, pp. 9072–9083, 2018a.

Farahmand, A.-m. Iterative value-aware model learning. In *Advances in Neural Information Processing Systems*, pp. 9072–9083, 2018b.

Gu, S., Lillicrap, T., Sutskever, I., and Levine, S. Continuous deep q-learning with model-based acceleration. In *International Conference on Machine Learning*, pp. 2829–2838, 2016.

Jin, C., Allen-Zhu, Z., Bubeck, S., and Jordan, M. I. Is q-learning provably efficient? In *Advances in Neural Information Processing Systems*, pp. 4863–4873, 2018.

Kakade, S. and Langford, J. Approximately optimal approximate reinforcement learning. In *ICML*, volume 2, pp. 267–274, 2002.

Kakade, S., Kearns, M. J., and Langford, J. Exploration in metric state spaces. In *Proceedings of the 20th International Conference on Machine Learning (ICML-03)*, pp. 306–312, 2003.

Kearns, M. and Singh, S. Near-optimal reinforcement learning in polynomial time. *Machine learning*, 49(2-3):209–232, 2002.

Kidambi, R., Rajeswaran, A., Netrapalli, P., and Joachims, T. Morel: Model-based offline reinforcement learning. *arXiv preprint arXiv:2005.05951*, 2020.

Lambert, N., Amos, B., Yadan, O., and Calandra, R. Objective mismatch in model-based reinforcement learning. *arXiv preprint arXiv:2002.04523*, 2020.

Levine, S., Finn, C., Darrell, T., and Abbeel, P. End-to-end training of deep visuomotor policies. *The Journal of Machine Learning Research*, 17(1):1334–1373, 2016.

Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., and Riedmiller, M. Playing atari with deep reinforcement learning. December 2013. URL http://arxiv.org/abs/1312.5602.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015.

Munos, R. and Szepesvári, C. Finite-time bounds for fitted value iteration. *Journal of Machine Learning Research*, 9(May):815–857, 2008.

Rajeswaran, A., Mordatch, I., and Kumar, V. A game theoretic framework for model based reinforcement learning. *arXiv preprint arXiv:2004.07804*, 2020.

Ross, S. and Bagnell, J. A. Agnostic system identification for model-based reinforcement learning. *arXiv preprint arXiv:1203.1007*, 2012.

Ross, S., Gordon, G., and Bagnell, D. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pp. 627–635, 2011.

Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., Van Den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., et al. Mastering the game of go with deep neural networks and tree search. *nature*, 529(7587): 484–489, 2016.

Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A., Hubert, T., Baker, L., Lai, M., Bolton, A., et al. Mastering the game of go without human knowledge. *Nature*, 550(7676):354–359, 2017.

Slaoui, R. B., Clements, W. R., Foerster, J. N., and Toth, S. Robust domain randomization for reinforcement learning. *arXiv preprint arXiv:1910.10537*, 2019.

Wang, H., Zheng, S., Xiong, C., and Socher, R. On the generalization gap in reparameterizable reinforcement learning. In *International Conference on Machine Learning*, pp. 6648–6658, 2019a.

Wang, T., Bao, X., Clavera, I., Hoang, J., Wen, Y., Langlois, E., Zhang, S., Zhang, G., Abbeel, P., and Ba, J. Benchmarking model-based reinforcement learning. *arXiv preprint arXiv:1907.02057*, 2019b.

Xu, H., Li, Y., Tian, Y., Darrell, T., and Ma, T. Algorithmic framework for model-based reinforcement learning with theoretical guarantees. *arXiv preprint arXiv:1807.03858*, 2018.

Zhang, A., Ballas, N., and Pineau, J. A dissection of overfitting and generalization in continuous reinforcement learning. *arXiv preprint arXiv:1806.07937*, 2018a.

Zhang, A., Wu, Y., and Pineau, J. Natural environment benchmarks for reinforcement learning. *arXiv preprint arXiv:1811.06032*, 2018b.

Appendix

A. Performance Difference

A.1. Proof of Lemma 2.1

Restating Lemma 2.1:

Lemma A.1. *(Model Performance Difference Lemma) Let M and M' be two different MDPs. Further, define $\mathcal{R}^\pi(s) = \mathbb{E}_{a\sim\pi(\cdot|s)}[\mathcal{R}(s,a)]$ and $\mathcal{R}^\pi_\epsilon(s) = \mathcal{R}^\pi_M(s) - \mathcal{R}^\pi_{M'}(s)$. For any policy $\pi \in \Pi$ we have:*

$$J_M(\pi) = J_{M'}(\pi) + \mathbb{E}_{s\sim d_{M,\pi}}[\mathcal{R}_\epsilon(s)] + \frac{1}{1-\gamma}\mathbb{E}_{s\sim d_{M,\pi}}\mathbb{E}_{s'\sim\mathcal{P}_M(s'|s,\pi)}\left[\boldsymbol{A}^\pi_{M'}(s,s')\right]$$

Proof. Let $\mathcal{P}_0$ be the start state distribution for both MDPs, $P^\pi_{M,t}$ be the state distribution at time t, starting from $s_0 \sim \mathcal{P}_0$ in M and $d_{M,\pi}$ denote the stationary state distribution under MDP M, policy π and start state $s_0 \sim \mathcal{P}_0$. Then, we have:

$$\begin{aligned}
&J_M(\pi) - J_{M'}(\pi)\\
&= \mathbb{E}_{s_0\sim\mathcal{P}_0}\left[V^\pi_M(s_0) - V^\pi_{M'}(s_0)\right]\\
&= (1-\gamma)\sum_{t=0}^{\infty}\gamma^t\mathbb{E}_{s_t\sim P^\pi_{M,t}}\mathbb{E}_{a_t\sim\pi(\cdot|s_t)}\left[\mathcal{R}_M(s_t,a_t)\right]\\
&\qquad - \mathbb{E}_{s_0\sim\mathcal{P}_0}\left[V^\pi_{M'}(s_0)\right]\\
&= (1-\gamma)\sum_{t=0}^{\infty}\gamma^t\mathbb{E}_{s_t\sim P^\pi_{M,t}}\left[\mathcal{R}^\pi_M(s_t)\right] - \mathbb{E}_{s_0\sim\mathcal{P}_0}\left[V^\pi_{M'}(s_0)\right]\\
&= \sum_{t=0}^{\infty}\gamma^t\mathbb{E}_{s_t\sim P^\pi_{M,t}}\left[(1-\gamma)\mathcal{R}^\pi_M(s_t) + V^\pi_{M'}(s_t) - V^\pi_{M'}(s_t)\right]\\
&\qquad - \mathbb{E}_{s_0\sim\mathcal{P}_0}\left[V^\pi_{M'}(s_0)\right]\\
&= \sum_{t=0}^{\infty}\gamma^t\mathop{\mathbb{E}}_{\substack{s_t\sim P^\pi_{M,t}\\ s_{t+1}\sim P^\pi_{M,t+1}}}\left[(1-\gamma)\mathcal{R}^\pi_M(s_t) + \gamma V^\pi_{M'}(s_{t+1}) - V^\pi_{M'}(s_t)\right]
\end{aligned}$$

Expanding $V^\pi_{M'}(s_t)$ with a one-step bellman evaluation operator:

$$\begin{aligned}
&= \sum_{t=0}^{\infty}\gamma^t\mathop{\mathbb{E}}_{\substack{s_t\sim P^\pi_{M,t}\\ s_{t+1}\sim P^\pi_{M,t+1}}}\Big[(1-\gamma)\mathcal{R}^\pi_M(s_t) + \gamma V^\pi_{M'}(s_{t+1})\\
&\qquad - \Big((1-\gamma)\mathcal{R}^\pi_{M'}(s_t) + \gamma\mathbb{E}_{s''\sim\mathcal{P}^\pi_{M'}(s_t,\pi)}\left[V^\pi_{M'}(s'')\right]\Big)\Big]\\
&= \sum_{t=0}^{\infty}\gamma^t\mathop{\mathbb{E}}_{\substack{s_t\sim P^\pi_{M,t}\\ s_{t+1}\sim P^\pi_{M,t+1}}}\Big[(1-\gamma)(\mathcal{R}^\pi_M(s_t) - \mathcal{R}^\pi_{M'}(s_t))\\
&\qquad + \gamma V^\pi_{M'}(s_{t+1}) - \gamma\mathbb{E}_{s''\sim\mathcal{P}^\pi_{M'}(s_t,\pi)}\left[V^\pi_{M'}(s'')\right]\Big]\\
&= \sum_{t=0}^{\infty}\gamma^t\mathop{\mathbb{E}}_{\substack{s_t\sim P^\pi_{M,t}\\ s_{t+1}\sim P^\pi_{M,t+1}}}\Big[(1-\gamma)R^\pi_\epsilon(s_t)\\
&\qquad + \gamma V^\pi_{M'}(s_{t+1}) - \gamma\mathbb{E}_{s''\sim\mathcal{P}^\pi_{M'}(s_t,\pi)}\left[V^\pi_{M'}(s'')\right]\Big]
\end{aligned}$$

Using definition of $\boldsymbol{A}^\pi_{M'}$:

$$\begin{aligned}
&= \sum_{t=0}^{\infty}\gamma^t\mathop{\mathbb{E}}_{\substack{s_t\sim P^\pi_{M,t}\\ s_{t+1}\sim P^\pi_{M,t+1}}}\left[(1-\gamma)R^\pi_\epsilon(s_t) + \boldsymbol{A}^\pi_{M'}(s_t,s_{t+1})\right]\\
&= \frac{1}{1-\gamma}\mathbb{E}_{s\sim d_{M,\pi}}\mathbb{E}_{s'\sim\mathcal{P}_M(s,\pi)}\left[\boldsymbol{A}^\pi_{M'}(s,s')\right] + \mathbb{E}_{s\sim d_{M,\pi}}\left[\mathcal{R}^\pi_\epsilon(s)\right]
\end{aligned}$$

□

A.2. Corollary: Deviation Error

The following is a useful corollary for subsequent proofs.

Corollary A.2. *Let M and M' be two different MDPs. For any policy $\pi \in \Pi$ we have:*

$$J_M(\pi) = J_{M'}(\pi) + \frac{1}{1-\gamma}\mathbb{E}_{s\sim d_{M,\pi}}\underbrace{\left[\mathcal{T}^\pi_M V^\pi_M(s) - \mathcal{T}^\pi_{M'}V^\pi_M(s)\right]}_{\textit{deviation error}} \tag{7}$$

Proof.

$$\begin{aligned}
&J_M(\pi) - J_{M'}(\pi)\\
&= \mathbb{E}_{s_0\sim\mathcal{P}_0}\left[V^\pi_M(s_0) - V^\pi_{M'}(s_0)\right]\\
&= \cdots
\end{aligned}$$

Proceeding similar to the previous proof upto the following line:

$$\begin{aligned}
&= \sum_{t=0}^{\infty}\gamma^t\mathop{\mathbb{E}}_{\substack{s_t\sim P^\pi_{M,t}\\ s_{t+1}\sim P^\pi_{M,t+1}}}\Big[(1-\gamma)\mathcal{R}^\pi_M(s_t) + \gamma V^\pi_{M'}(s_{t+1})\\
&\qquad - \Big((1-\gamma)\mathcal{R}^\pi_{M'}(s_t) + \gamma\mathbb{E}_{s''\sim\mathcal{P}^\pi_{M'}(s_t,\pi)}\left[V^\pi_{M'}(s'')\right]\Big)\Big]\\
&= \sum_{t=0}^{\infty}\gamma^t\mathbb{E}_{s_t\sim P^\pi_{M,t}}\Big[\mathcal{T}^\pi_M V^\pi_{M'}(s_t)\\
&\qquad - \Big((1-\gamma)\mathcal{R}^\pi_{M'}(s_t) + \gamma\mathbb{E}_{s''\sim\mathcal{P}^\pi_{M'}(s_t,\pi)}\left[V^\pi_{M'}(s'')\right]\Big)\Big]\\
&= \sum_{t=0}^{\infty}\gamma^t\mathbb{E}_{s_t\sim P^\pi_{M,t}}\left[\mathcal{T}^\pi_M V^\pi_{M'}(s_t) - \mathcal{T}^\pi_{M'}V^\pi_{M'}(s_t)\right]\\
&= \frac{1}{1-\gamma}\mathbb{E}_{s\sim d_{\pi,M}}\left[\mathcal{T}^\pi_M V^\pi_{M'}(s) - \mathcal{T}^\pi_{M'}V^\pi_{M'}(s)\right]
\end{aligned}$$

□

Note that we can further upper bound the difference in values across MDPs for a policy as follows, which will be useful in subsequent proofs. We compute this bound at an arbitrary start state s_0, and it will then hold for any start state. Let $d_{M,s_0,\pi}$ be the stationary state distribution of following policy π in MDP M, starting at state s_0.

$$\begin{aligned}
&V^\pi_M(s_0) - V^\pi_{M'}(s_0)\\
&= \cdots
\end{aligned} \tag{8}$$

Proceeding similar to the proof in Appendix A.1, we get the following:

$$= \frac{1}{1-\gamma}\mathbb{E}_{d_{M,s_0,\pi}}\left[\mathcal{T}_M^\pi V_{M'}^\pi - \mathcal{T}_{M'}^\pi V_{M'}^\pi\right]$$

$$\leq \frac{1}{1-\gamma}\left\|\mathcal{T}_M^\pi V_{M'}^\pi - \mathcal{T}_{M'}^\pi V_{M'}^\pi\right\|_\infty \tag{9}$$

$$\leq \frac{1}{1-\gamma}\Bigg[\left\|\mathcal{R}_{M'}^\pi - \mathcal{R}_M^\pi\right\|_\infty \tag{10}$$

$$+\gamma \max_s \sum_{s'\in\mathcal{S}} V_{M'}^\pi(s')\mathbb{E}_{a\sim\pi(s)}\left[p_{M'}(s'|s,a) - p_M(s',a)\right]\Bigg]$$

$$\leq \frac{1}{1-\gamma}\Bigg[\epsilon_R \tag{11}$$

$$+\gamma\left\|V_{M'}^\pi\right\|_\infty \underbrace{\max_s \max_a \left\|p_{M'}(s'|s,a) - p_M(s',a)\right\|_1}_{\epsilon_P}\Bigg]$$

$$\leq \frac{1}{1-\gamma}\left[\epsilon_R + \gamma\left\|V_{M'}^\pi\right\|_\infty \epsilon_P\right]$$

$$\leq \frac{1}{1-\gamma}\left[\epsilon_R + \frac{\gamma\epsilon_P R_{\max}}{1-\gamma}\right] \tag{12}$$

B. Generalization with VI & FQI

B.1. Value Iteration: Proof of Theorem 3.1

Restating Theorem 3.1:

Theorem B.1. *Let M, M' be two MDPs s.t. $\max_s \max_a |\mathcal{R}_M(s,a) - \mathcal{R}_{M'}(s,a)| \leq \epsilon_R$ and $\max_s \max_a \|p_M(s,a) - p_{M'}(s,a)\|_1 \leq \epsilon_P$. Let π_{n+1} be the policy obtained after n VI iterations on MDP M. Then we have,*

$$\left\|V_{M'}^{\pi_{n+1}} - V_{M'}^\star\right\|_\infty \leq \frac{1}{1-\gamma}\left[\gamma\epsilon^{(n)} + 2\epsilon_R + \frac{2\epsilon_P R_{max}}{1-\gamma}\right]$$

Proof. In the main paper, we derived the following results:

$$\left\|V_{M'}^{\pi_{n+1}} - V_{M'}^{\pi'^\star}\right\|_\infty \leq \left\|V_{M'}^{\pi_{n+1}} - V_M^{\pi_{n+1}}\right\|_\infty + \left\|V_M^{\pi_{n+1}} - V_{M'}^{\pi'^\star}\right\|_\infty \tag{13}$$

$$\left\|V_M^{\pi_{n+1}} - V_{M'}^{\pi'^\star}\right\|_\infty \leq \frac{\gamma\epsilon_n}{1-\gamma} + \frac{\delta_{n+1}\left(V_M^{\pi_{n+1}}\right)}{1-\gamma} \tag{14}$$

Now, the first term in the RHS of eq. (13), we use the upper bound derived in Equation (12):

$$V_{M'}^\pi - V_M^\pi \leq \frac{1}{1-\gamma}\left[\epsilon_R + \frac{\gamma\epsilon_P R_{\max}}{1-\gamma}\right] \tag{15}$$

Notice that $\delta_{M,M'}$ occurs in eq. (9), and using the same proof as that of eq. (12), we get the following bount on $\delta_{M,M'}$ for any V.

$$\delta_{M,M'}(V) \leq \epsilon_R + \gamma\epsilon_P\|V\|_\infty$$

$$\leq \epsilon_R + \frac{\gamma\epsilon_P R_{\max}}{1-\gamma} \tag{16}$$

Putting together eqs. (14) to (16), we get the desired bound. □

B.2. Fitted-Q Iteration

Recall that $D = \{(s,a,r,s')_i\}_{i=1}^n$ is a dataset of experiences where $(s,a) \sim \mu \times U$ for some state-distribution μ and policy U, $s' \sim \mathcal{P}_M(s' \mid s,a)$ and $r \sim \mathcal{R}_M(s,a)$. FQI is an iterative algorithm that learns a function f in the model-class $\mathcal{F}$ that approximates the Q-function. At each iteration $f_k = \hat{\mathcal{T}}_M f_{k-1}$ where $\hat{\mathcal{T}}_M$ is the *empirical* Bellman operator defined as follows:

$$\hat{\mathcal{T}}_M f := \operatorname*{argmin}_{f\in\mathcal{F}} \mathcal{L}_D(f,f')$$

$$\mathcal{L}_D(f,f') := \frac{1}{|D|}\sum_{(s,a,r,s')\in D}\left(f(s,a) - r - \gamma V_{f'}(s')\right)^2$$

We define norms for functions over $\mathcal{S}\times\mathcal{A}$, similar to (Munos & Szepesvári, 2008), as follows: $\|g\|_{\nu\times\pi} := \left(\mathbb{E}_{s\sim\nu,a\sim\pi}\left[|g|^2\right]\right)^{1/2}$ for $\nu\times\pi \in \Delta(\mathcal{S}\times\mathcal{A})$ and $g : \mathcal{S}\times\mathcal{A} \to \mathbb{R}$

Theorem B.2. *Let M, M' be two MDPs s.t. $\max_s \max_a |\mathcal{R}_M(s,a) - \mathcal{R}_{M'}(s,a)| \leq \epsilon_R$ and $\max_s \max_a \|p_M(s,a) - p_{M'}(s,a)\|_1 \leq \epsilon_P$. Let $D = \{(s,a,r,s')_i\}_{i=1}^n$ be generated as $(s,a) \sim \mu\times U$, where μ is exploratory and U is uniform over actions, $r \sim R_M(s,a)$, $s' \sim P_M(s,a)$. Let π_{f_k} be the greedy policy w.r.t. f_k, obtained after k iterations of FQI on D. Then we have,*

$$\left\|Q_{M'}^{\pi_{f_k}} - Q_{M'}^\star\right\|_{\nu\times\pi}$$

$$\leq \frac{1}{1-\gamma}\left[\gamma^k R_{max} + O\left(\sqrt{|\mathcal{A}|\,\epsilon^{(n)}}\right) + 2\epsilon_R + \frac{2\epsilon_P R_{max}}{1-\gamma}\right]$$

Proof. Let $\hat{\pi} := \pi_{f_k}$, let $v'^\star, \pi'^\star$ denote the optimal value function and policy in the second MDP M', and $Q_{M'}^\star := Q_{M'}^{\pi'^\star}$. Let

$$v_{M'}^{\pi'^\star} - v_{M'}^{\hat{\pi}} \leq \sum_{t=1}^\infty \gamma^{t-1}\mathbb{E}_{s\sim P_{M',t}^{\hat{\pi}}}\left[Q_{M'}^\star(s,\pi'^\star) - Q_{M'}^\star(s,\hat{\pi})\right]$$

$$\leq \sum_{h=1}^\infty \gamma^{h-1}\Bigg[\left\|Q_{M'}^\star - Q_{M'}^{\hat{\pi}}\right\|_{P_{M',t}^{\hat{\pi}}\times\pi'^\star}$$

$$+\left\|Q_{M'}^\star - Q_{M'}^{\hat{\pi}}\right\|_{P_{M',t}^{\hat{\pi}}\times\hat{\pi}}\Bigg]$$

$$\leq \left(\frac{2}{1-\gamma}\right)\max_{\nu\times\pi\in\Delta(\mathcal{S}\times\mathcal{A})}\left\|Q_{M'}^\star - Q_{M'}^{\hat{\pi}}\right\|_{\nu\times\pi}$$

Now, it remains to bound $\left\|Q_{M'}^{\hat{\pi}} - Q_{M'}^\star\right\|_{\nu\times\pi}$ for any $\nu\times\pi \in \Delta(\mathcal{S}\times\mathcal{A})$. First we look at the term $\left\|Q_M^{\hat{\pi}} - Q_{M'}^\star\right\|_{\nu\times\pi} = \|f_k - Q_{M'}^\star\|_{\nu\times\pi}$ below. Recall that $f_k = \hat{\mathcal{T}}_M f_{k-1}$.

$$\|f_k - Q_{M'}^\star\|_{\nu\times\pi}$$

$$= \left\|f_k - \mathcal{T}_M f_{k-1} + \mathcal{T}_M f_{k-1} - Q_{M'}^\star\right\|_{\nu\times\pi}$$

$$\leq \|f_k - \mathcal{T}_M f_{k-1}\|_{\nu\times\pi}$$

$$+\left\|\mathcal{T}_M f_{k-1} - \mathcal{T}_{M'} f_{k-1} + \mathcal{T}_{M'} f_{k-1} - Q_{M'}^\star\right\|_{\nu\times\pi}$$

$$\leq \sqrt{|\mathcal{A}|C}\,\|f_k - \mathcal{T}_M f_{k-1}\|_{\mu\times U}$$

$$+\underbrace{\left\|\mathcal{T}_M f_{k-1} - \mathcal{T}_{M'} f_{k-1}\right\|_{\nu\times\pi}}_{\text{Deviation error}} + \left\|\mathcal{T}_{M'} f_{k-1} - Q_{M'}^\star\right\|_{\nu\times\pi}$$

$$\leq \sqrt{2|\mathcal{A}|C\epsilon_n} + \delta_{M,M'}^{(k-1)} + \gamma\left\|V_M^{\pi_{f_{k-1}}} - V_{M'}^\star\right\|_{P'(\nu\times\pi)}$$

Defining $\pi_{f,f_k}(s) := \operatorname{argmax}_{a\in\mathcal{A}} \max\{f(s,a), f_k(s,a)\}$, it is easy to verify that $\|V_f - V_{f_k}\|_\nu \leq \|f - f_k\|_{\nu\times\pi_{f,f_k}}$

$$\leq \sqrt{2|\mathcal{A}|C\epsilon_n} + \delta^{(k-1)}_{M,M'} + \gamma \left\| f_{k-1} - Q^\star_{M'} \right\|_{P'(\nu\times\pi)\times\pi_{f_{k-1},Q'^*}}$$

$$\Rightarrow \left\| f_k - Q^\star_{M'} \right\|_{\nu\times\pi} \leq \frac{1-\gamma^k}{1-\gamma}\sqrt{2|\mathcal{A}|C\epsilon_n} + \sum_{i=1}^{k} \gamma^i \delta^{(i-1)}_{M,M'} + \gamma^k \left\| f_0 - Q^\star_{M'} \right\|_{P'(\nu\times\pi)\times\pi_{f_0,Q'^*}}$$

Following (Munos & Szepesvári, 2008), we denote ϵ_n as a bound on the error of using an empirical bellman operator (with a finite dataset) as opposed to the true bellman operator.

$$\leq \frac{1-\gamma^k}{1-\gamma}\sqrt{2|\mathcal{A}|C\epsilon_n} + \sum_{i=1}^{k} \gamma^i \delta^{(i-1)}_{M,M'} + \frac{\gamma^k R_{\max}}{1-\gamma}$$

Now, we can use this to bound $\left\| Q^{\hat{\pi}}_{M'} - Q^\star_{M'} \right\|_{\nu\times\pi}$ as follows, where $f_k := Q^{\hat{\pi}}_M$:

$$\begin{aligned}
\left\| Q^{\hat{\pi}}_{M'} - Q^\star_{M'} \right\|_{\nu\times\pi} &= \left\| Q^{\hat{\pi}}_{M'} - Q^{\hat{\pi}}_M + Q^{\hat{\pi}}_M - Q^\star_{M'} \right\|_{\nu\times\pi} \\
&\leq \left\| Q^{\hat{\pi}}_{M'} - Q^{\hat{\pi}}_M \right\|_{\nu\times\pi} + \left\| f_k - Q^\star_{M'} \right\|_{\nu\times\pi} \\
&\leq \underbrace{\left\| Q^{\hat{\pi}}_{M'} - Q^{\hat{\pi}}_M \right\|_{\nu\times\pi}}_{\text{Transfer Error}} + \underbrace{\frac{1-\gamma^k}{1-\gamma}\sqrt{2|\mathcal{A}|C\epsilon_n}}_{\text{FQI finite data error}} \\
&\quad + \underbrace{\sum_{i=1}^{k} \gamma^i \delta^{(i-1)}_{M,M'}}_{\text{Deviation Error}} + \underbrace{\frac{\gamma^k R_{\max}}{1-\gamma}}_{\text{FQI finite iterations error}}
\end{aligned}$$

We can write the deviation error:

$$\begin{aligned}
&\delta^{(k-1)}_{M,M'} \\
&= \left\| \mathcal{T}_M f_{k-1} - \mathcal{T}_{M'} f_{k-1} \right\|_{\nu\times\pi} \\
&= \mathbb{E}_{(s,a)\sim\nu\times\pi}\left[(\mathcal{T}_M f_{k-1})(s,a) - (\mathcal{T}_{M'} f_{k-1})(s,a) \right] \\
&= \mathbb{E}_{(s,a)\sim\nu\times\pi}\Big[\Big(R_M(s,a) - R_{M'}(s,a) + \\
&\quad \gamma \left(\mathbb{E}_{s'\sim P_M(s,a)}\left[V_{f_{k-1}}(s') \right] - \mathbb{E}_{s'\sim P_{M'}(s,a)}\left[V_{f_{k-1}}(s') \right] \right) \Big) \Big] \\
&\leq \epsilon_R + \Big\| \gamma \Big(\mathbb{E}_{s'\sim P_M(s,a)}\left[V_{f_{k-1}}(s') \right] \\
&\qquad - \mathbb{E}_{s'\sim P_{M'}(s,a)}\left[V_{f_{k-1}}(s') \right] \Big) \Big\|_{\nu\times\pi}
\end{aligned}$$

Considering the term on the right (squared):

$$\begin{aligned}
&\left\| \gamma \left(\mathbb{E}_{s'\sim P_M(s,a)}\left[V_{f_{k-1}}(s') \right] - \mathbb{E}_{s'\sim P_{M'}(s,a)}\left[V_{f_{k-1}}(s') \right] \right) \right\|^2_{\nu\times\pi} \\
&= \mathbb{E}_{(s,a)\sim\nu\times\pi}\Bigg[\\
&\gamma^2 \left(\int_{s'\in\mathcal{S}} V_{f_{k-1}}(s') [p_M(s'|s,a) - p_{M'}(s'|s,a)] ds' \right)^2 \Bigg]
\end{aligned}$$

Using Holder's with $1/q_1 + 1/q_2 = 1$:

$$\begin{aligned}
&\leq \mathbb{E}_{(s,a)\sim\nu\times\pi}\Bigg[\gamma^2 \left(\int_{s'\in\mathcal{S}} \left| V_{f_{k-1}}(s') \right|^{q_2} ds' \right)^{2/q_2} \\
&\left(\int_{s'\in\mathcal{S}} \left| p_M(s'|s,a) - p_{M'}(s'|s,a) \right|^{q_1} ds' \right)^{2/q_1} \Bigg]
\end{aligned}$$

Setting $q_1 := 1$, $q_2 := \infty$, and using $\max_s \max_a \left\| p_M(s,a) - p_{M'}(s,a) \right\|_1 \leq \epsilon_P$

Algorithm 1 Imitation Learning for MBRL

Input: Generative model of $\mathcal{P}_{M'}$, reward function $\mathcal{R}_{M'}$, exploratory policy μ, initial model $\hat{\mathcal{P}}_{M,1}$, parameters N,m, online learner OLALGORITHM, policy learning algorithm POLICYUPDATE
Output: Sequence of policies $\pi_{1:N}$

Initialize $\mathcal{D} \leftarrow \varnothing$
Initialize $\pi_1 \leftarrow$ POLICYUPDATE$(\hat{\mathcal{P}}_{M,1}, \mathcal{R}_{M'})$
for $i = 2$ **to** N **do**
  Collect m samples of (s,a) using a mixture of $d_{\pi,\mathcal{P}_{M'}}$ and $d_{\mu,\mathcal{P}_{M'}}$ (with equal probabilities)
  Create dataset $\mathcal{D}_i = \{(s,a,\mathcal{P}_{M'}(s,a))$
  Aggregate datasets $\mathcal{D} \leftarrow \mathcal{D} \bigcup \mathcal{D}_i$
  Train $\hat{\mathcal{P}}_{M,i}$ over $\mathcal{D}$ using a first order oracle of loss $\ell_i(\hat{P}_{M,i})$ and OLALGORITHM
  $\pi_i \leftarrow$ POLICYUPDATE$(\mathcal{R}_{M'}, \hat{P}_{M,i})$
end for

$$\begin{aligned}
&= \mathbb{E}_{(s,a)\sim\nu\times\pi}\left[\gamma^2 \epsilon_P^2 \left\| V_{f_{k-1}} \right\|^2_\infty \right] \\
&= \gamma^2 \epsilon_P^2 \left\| V_{f_{k-1}} \right\|^2_\infty
\end{aligned}$$

Now, we get the following relation for $\delta^{(k-1)}_{M,M'}$:

$$\begin{aligned}
\Rightarrow \delta^{(k-1)}_{M,M'} &\leq \epsilon_R + \gamma\epsilon_P \left\| V_{f_{k-1}} \right\|_\infty \\
&\leq \epsilon_R + \frac{\gamma\epsilon_P R_{\max}}{1-\gamma}
\end{aligned}$$

We get the following bound on transfer error:

$$\left\| Q^{\hat{\pi}}_{M'} - Q^{\hat{\pi}}_M \right\|_{\nu\times\pi} \leq \epsilon_R + \frac{\gamma\epsilon_P R_{\max}}{1-\gamma}$$

Substituting back the deviation error, we get the desired bound.

$$\left\| Q^{\hat{\pi}}_{M'} - Q^\star_{M'} \right\|_{\nu\times\pi} \tag{17}$$

$$\begin{aligned}
&\leq \frac{1}{1-\gamma}\Bigg[\gamma^k R_{\max} + \left(1-\gamma^k\right)\sqrt{2|\mathcal{A}|C\epsilon_n} \\
&\qquad + 2\left(\frac{\gamma\epsilon_P R_{\max}}{1-\gamma} + \epsilon_R \right)\left(1-\gamma^k\right) \Bigg] \qquad (18) \\
&\leq \frac{1}{1-\gamma}\left[\gamma^k R_{\max} + O\left(\sqrt{|\mathcal{A}|\,\epsilon^{(n)}} \right) + 2\epsilon_R + \frac{2\epsilon_P R_{\max}}{1-\gamma} \right]
\end{aligned}$$

□

C. Connections to Imitation Learning

Consider the following algorithm. Note that this algorithm has been presented by (Ross & Bagnell, 2012) where they show analysis with DAGGER as their online learner.

Theorem C.1. *Let $\ell_i(\mathcal{P}_M) := \mathbb{E}_{d_{M,\pi_i}}\mathbb{E}_{\mathcal{P}_M}[\tilde{c}]$ where $\tilde{c} : \mathcal{S}\times\mathcal{S} \to \mathbb{R}$ is a surrogate loss satisfying: $\forall s \in \mathcal{S}, \pi \in \Pi$, $\exists$ constant $C_{M'} > 0$ s.t. $\mathbb{E}_{\mathcal{P}_{M'}}[|\boldsymbol{A}^\pi_M|] \leq C_{M'}\mathbb{E}_{\mathcal{P}_{M'}}[\tilde{c}]$. Assume access*

to a sampling distribution $\nu \in \Delta(\mathcal{S} \times \mathcal{A})$ which is exploratory and $c_\nu^\pi := \sup_{s,a} \frac{d_{M',\pi}(s,a)}{\nu(s,a)}$. Given a no-regret online learner that plays $\{\mathcal{P}_{M,i}\}_{i=1}^N$ and a corresponding sequence of policies $\{\pi_i\}_{i=1}^N$ s.t. $\pi_i = \text{planner}(\mathcal{P}_{M,i})$, then we have

$$\frac{1}{N}\left[\sum_{i=1}^N J_{M'}(\pi_n)\right] - J_{M'}(\pi^\star_{M'})$$
$$\leq \bar{\epsilon}^{\pi^\star_{M'}}_{\text{planner}} + \frac{2c_\nu^{\pi^\star_{M'}} C_{M'}}{1-\gamma}\left(\epsilon_{\text{model}} + \epsilon_{\text{regret}}\right)$$

where $\bar{\epsilon}^{\pi^\star_{M'}}_{\text{planner}} = \frac{1}{N}\sum_{i=1}^N \left[J_{M,i}(\pi_i) - J_{M,i}(\pi^\star_{M'})\right]$, $\epsilon_{\text{model}} = \inf_{\hat{\mathcal{P}}} \mathbb{E}_{(s,a)\sim d_{M',\bar{\rho}}}\left[\ell_i(\hat{\mathcal{P}})\right]$ *and* $\epsilon_{\text{regret}} = \mathcal{O}(1/\sqrt{N})$.

Proof. Let $\rho_i := \frac{1}{2}\nu + \frac{1}{2}D_{\mu,\pi_i}$ and $\bar{\rho} := \sum_{i=1}^N \rho_i$

$$\min_{\pi \in \pi_{1:N}} J_{M'}(\pi) - J_{M'}\left(\pi'\right)$$
$$\leq \frac{1}{N}\sum_{i=1}^N \left[J_{M'}\left(\pi_i\right) - J_{M'}\left(\pi'\right)\right]$$
$$\leq \frac{1}{N}\sum_{i=1}^N \Big[J_{M'}\left(\pi_i\right) - J_M\left(\pi_i\right) + J_M\left(\pi_i\right) - J_M\left(\pi'\right) + J_M\left(\pi'\right) - J_{M'}\left(\pi'\right)\Big]$$

Using the model performance difference lemma 2.1

$$\leq \underbrace{\frac{1}{N}\sum_{i=1}^N \left[J_M\left(\pi_i\right) - J_M\left(\pi'\right)\right]}_{\bar{\epsilon}^{\pi'}_{\text{planner}}}$$
$$+ \frac{1}{(1-\gamma)N}\sum_{i=1}^N \Bigg[\mathbb{E}_{s\sim d_{M',\pi_i}} \underbrace{\mathbb{E}_{s'\sim \mathcal{P}_{M'}(s'|s,\pi_i)}\left[\left|\boldsymbol{A}_M^{\pi_i}(s,s')\right|\right]}_{\leq \text{surrogate loss}} + \mathbb{E}_{s\sim d_{M',\pi'}}\mathbb{E}_{s'\sim \mathcal{P}_{M'}(s'|s,\pi')}\left[\left|\boldsymbol{A}_M^{\pi'}(s,s')\right|\right]\Bigg]$$
$$\leq \bar{\epsilon}^{\pi'}_{\text{planner}} + \frac{C_{M'}}{1-\gamma}\frac{1}{N}\sum_{i=1}^N \Big[\mathbb{E}_{(s,a)\sim d_{M',\pi_i}\pi_i}\left[\ell_i(\hat{\mathcal{P}})\right] + \mathbb{E}_{(s,a)\sim d_{M',\pi'}\pi'}\left[\ell_i(\hat{\mathcal{P}})\right]\Big]$$

Using the definition of $c_\nu^{\pi'}$

$$\leq \bar{\epsilon}^{\pi'}_{\text{planner}} + \frac{C_{M'}}{1-\gamma}\frac{1}{N}\sum_{i=1}^N \Big[\mathbb{E}_{(s,a)\sim d_{M',\pi_i}\pi_i}\left[\ell_i(\hat{\mathcal{P}})\right] + c_\nu^{\pi'}\mathbb{E}_{(s,a)\sim\nu}\left[\ell_i(\hat{\mathcal{P}})\right]\Big]$$
$$\leq \bar{\epsilon}^{\pi'}_{\text{planner}} + c_\nu^{\pi'}\frac{C_{M'}}{1-\gamma}\frac{1}{N}\sum_{i=1}^N \Big[\mathbb{E}_{(s,a)\sim d_{M',\pi_i}\pi_i}\left[\ell_i(\hat{\mathcal{P}})\right] + \mathbb{E}_{(s,a)\sim\nu}\left[\ell_i(\hat{\mathcal{P}})\right]\Big]$$
$$= \bar{\epsilon}^{\pi'}_{\text{planner}} + 2c_\nu^{\pi'}\frac{C_{M'}}{1-\gamma}\frac{1}{N}\sum_{i=1}^N \left[\mathbb{E}_{(s,a)\sim\rho_i}\left[\ell_i(\hat{\mathcal{P}})\right]\right]$$

Adding and substracting the best loss in hindsight

$$= \bar{\epsilon}^{\pi'}_{\text{planner}} + 2c_\nu^{\pi'}\frac{C_{M'}}{1-\gamma}\left[\epsilon_{\text{regret}} + \inf_{\hat{\mathcal{P}}}\mathbb{E}_{(s,a)\sim\bar{\rho},}\left[\ell_i(\hat{\mathcal{P}})\right]\right]$$
$$= \bar{\epsilon}^{\pi'}_{\text{planner}} + 2c_\nu^{\pi'}\frac{C_{M'}}{1-\gamma}\left[\epsilon_{\text{regret}} + \epsilon_{\text{model}}\right]$$

Now, the proof for $\epsilon_{\text{regret}} = \mathcal{O}(1/\sqrt{N})$ is the same as that of any no-regret online learning algorithm (see (Ross et al., 2011)). Hence, we get the desired bound. □