On the Generalization of Representations in Reinforcement Learning

Charline Le Lan
University of Oxford

Stephen Tu
Google Brain

Adam Oberman
McGill University

Rishabh Agarwal
Google Brain

Marc Bellemare
Google Brain

Abstract

In reinforcement learning, state representations are used to tractably deal with large problem spaces. State representations serve both to approximate the value function with few parameters, but also to generalize to newly encountered states. Their features may be learned implicitly (as part of a neural network) or explicitly (for example, the successor representation of Dayan (1993)). While the approximation properties of representations are reasonably well-understood, a precise characterization of how and when these representations generalize is lacking. In this work, we address this gap and provide an informative bound on the generalization error arising from a specific state representation. This bound is based on the notion of effective dimension which measures the degree to which knowing the value at one state informs the value at other states. Our bound applies to any state representation and quantifies the natural tension between representations that generalize well and those that approximate well. We complement our theoretical results with an empirical survey of classic representation learning methods from the literature and results on the Arcade Learning Environment, and find that the generalization behaviour of learned representations is well-explained by their effective dimension.

Proceedings of the 25th International Conference on Artificial Intelligence and Statistics (AISTATS) 2022, Valencia, Spain. PMLR: Volume 151.

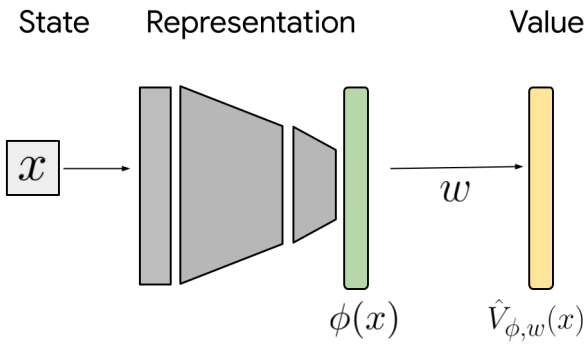

Figure 1: A deep RL architecture seen as a deep representation ϕ and a value prediction $\hat{V}_{\phi,w}$.

1 INTRODUCTION

At the heart of reinforcement learning (RL) is the problem of predicting the expected return that can be obtained from different states. In most practical situations, these predictions are made on the basis of parametric function approximation, needed in order to make accurate predictions on the basis of limited samples – technically speaking, to estimate the *value function* (Sutton and Barto, 2018). Linear function approximation, for example, estimates the value function using a fixed state representation ϕ which maps states to vectors in $\mathbb{R}^k$; general-purpose algorithms for constructing state representations include tile coding (Sutton, 1996), the Fourier basis (Konidaris et al., 2011), local basis functions (Ratitch and Precup, 2004), and methods based on properties of the transition function (Mahadevan and Maggioni, 2007; Ghosh and Bellemare, 2020). Common deep RL network architectures such as DQN (Mnih et al., 2015) use multiple layers of nonlinear transformations to map perceptual inputs to a final layer which is linearly transformed into a value function prediction (Figure 1); accordingly, we may also view this final layer as a (time-varying) state representation ϕ (Levine et al., 2017; Chung et al., 2018).

It is generally believed that auxiliary tasks, known to improve performance in deep reinforcement learning (Jaderberg et al., 2017; Bellemare et al., 2017), play an important role in shaping the learned state representation (Bellemare et al., 2019; Dabney et al., 2020; Lyle et al., 2021). This motivates the need to understand how representation learning impacts policy evaluation. In this paper, we give a theoretical characterization of the generalization properties of a given or learned representation. While there are a number of results characterizing the approximation error due to a representation (Petrik, 2007; Parr et al., 2008), its effect on statistical error is relatively unknown.

Our first contribution is a bound on the generalization error (approximation + estimation) that arises when performing Monte Carlo value function estimation with a given k-dimensional representation ϕ (Section 3). Critically, this bound depends on the (in)coherence of the feature matrix Φ (Candès and Recht, 2009), which in turns defines the *effective dimension* of the representation. This effective dimension determines how many samples are needed to obtain a good generalization of the value function with the chosen representation; it may be as low as k, indicating that generalization is as good as possible, or as high as $|S|$, the number of states, indicating no generalization at all. The bound applies more broadly to the generalization error incurred in least-squares regression problems where a subset of a larger set of points is observed.

In Section 4, we demonstrate the usefulness of our bound by specializing it to study the generalization properties of the successor representation (SR) (Dayan, 1993). Specifically, we consider the state representation constructed from the top k singular vectors of the SR (Stachenfeld et al., 2014; Machado et al., 2017; Behzadian and Petrik, 2018). Empirically, we find that the effective dimension of this representation – and consequently its generalization characteristics – can vary substantially according to the transition structure of the environment. We also show empirically that the effective dimension is important to determine the generalization capacity of different theoretically-motivated representations in the four room domain (Sutton et al., 1999).

In an empirical study on the Arcade Learning Environment (Bellemare et al., 2013), we find that the notions of incoherence and effective dimension correlate with the observed empirical performance of existing value-based deep RL agents (Subsection 5.2). Furthermore, we find that a simple auxiliary loss motivated by our bound shows promising gains in the offline deep RL setting.

2 BACKGROUND

We consider a Markov Decision Process (MDP) $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, \mathcal{R}, \mathcal{P}, \gamma \rangle$ (Puterman, 1994) with finite state space $\mathcal{S}$, discrete set of actions $\mathcal{A}$, transition kernel $\mathcal{P} : \mathcal{S} \times \mathcal{A} \to \mathscr{P}(\mathcal{S})$, deterministic reward function $\mathcal{R} : \mathcal{S} \times \mathcal{A} \to [-R_{\max}, R_{\max}]$, and discount factor $\gamma \in [0, 1)$. For simplicity, we make the correspondence $\mathcal{S} = \{1, ..., S\}$. We write $\mathcal{P}_s^a$ to denote the next-state distribution over $\mathcal{S}$ resulting from selecting action a in s and write $\mathcal{R}_s^a$ for the corresponding reward.

A stationary policy $\pi : \mathcal{S} \to \mathscr{P}(\mathcal{A})$ is a mapping from states to distributions over actions, describing a particular way of interacting with the environment. We denote the set of all policies by Π. For any policy $\pi \in \Pi$, the value function $V^\pi(s)$ measures the expected discounted sum of rewards received when starting from state $s \in \mathcal{S}$ and acting according to π:

$$V^\pi(s) := \mathop{\mathbb{E}}_{\pi, \mathcal{P}} \left[\sum_{t=0}^{\infty} \gamma^t \mathcal{R}_{s_t}^{a_t} \mid s_0 = s, a_t \sim \pi(\cdot \mid s_t) \right].$$

The upper-bound value is $V_{\max} := \frac{R_{\max}}{1-\gamma}$. In vector notation (Puterman, 1994), let $r_\pi \in \mathbb{R}^S$ denote the vector of expected rewards, and let $P_\pi \in \mathbb{R}^{S \times S}$ be the transition matrix whose entries are

$$P_\pi(s, s') = \sum_{s' \in \mathcal{S}} \mathcal{P}_s^a(s') \pi(a \mid s).$$

We then have

$$V^\pi = \sum_{t=0}^{\infty} (\gamma P_\pi)^t r_\pi = (I - \gamma P_\pi)^{-1} r_\pi.$$

In this paper we consider approximating the value function V^π using a linear combination of features. We call the map $\phi : \mathcal{S} \to \mathbb{R}^k$ a *k-dimensional state representation*; $\phi(s)$ is the feature vector for a state $s \in \mathcal{S}$. In general, we will be interested in the setting where $k \ll S$. The value function approximation at s is

$$V_{\phi,w}(s) = \phi(s)^\top w,$$

where $w \in \mathbb{R}^k$ is a weight vector. We collect the per-state feature vectors into a feature matrix $\Phi \in \mathbb{R}^{S \times k}$. For simplicity, we assume Φ has full column rank. In vector form, the value function approximation (a S-dimensional vector) is more directly expressed as

$$V_{\phi,w} = \Phi w.$$

2.1 Statistical Learning Theory

We consider the *batch Monte Carlo policy evaluation* setting, in which we are given a sample of training

examples $D = \{(s_1, y_1), \ldots, (s_n, y_n)\} \in (\mathcal{S} \times \mathbb{R})^n$ and wish to determine a good linear approximation to V^π on the basis of this sample. Here, s_i is a state and y_i is a realisation of the random return $G^\pi(s_i)$ (Bellemare et al., 2017; Sutton and Barto, 2018), defined by the random-variable equation

$$G^\pi(s) = \sum_{t=0}^{\infty} \gamma^t \mathcal{R}_{s_t}^{a_t}, \quad s_0 = s, a_t \sim \pi(\cdot \mid s_t).$$

We assume that s_i is drawn uniformly at random from $\mathcal{S}$.[1] The batch Monte Carlo setting obviates some of the technical challenges in analyzing iterative methods such as least-squares TD (LSTD) but still allows us to provide practically-relevant theoretical guarantees.

We measure the quality of a linear approximation $V_{\phi,w}$ in terms of the expected squared error

$$R(V_{\phi,w}) = \frac{1}{S} \sum_{s \in \mathcal{S}} \mathop{\mathbb{E}}_{y \sim G^\pi(s)} \left(V_{\phi,w}(s) - y\right)^2. \qquad (1)$$

For a value function V, we express this error and related quantities in terms of the uniformly-weighted L^2 norm

$$\|V\|_{S,2} = \sqrt{\frac{1}{S} \sum_{s \in \mathcal{S}} \left(V(s)\right)^2}.$$

Following terminology from statistical learning theory (Vapnik, 1995), we call $R(V_{\phi,w})$ the *population risk* of $V_{\phi,w}$. One can verify that $R(V_{\phi,w})$ is minimized when $V_{\phi,w} = V^\pi$.

Given the dataset D and a fixed state representation ϕ, least-squares regression determines the weight vector $\hat{w}$ minimizing the *empirical risk function*

$$\hat{R}(V_{\phi,w}) = \frac{1}{n} \sum_{i=1}^{n} (V_{\phi,w}(s_i) - y_i)^2.$$

Notice that $\hat{R}$ is a random function as it depends on the training sample D.

We are interested in the performance of the least-squares approximation $V_{\phi,\hat{w}}$ compared to the true value function V^π. Let us denote by V_{ϕ,w^*} the linear approximation minimizing the population risk, such that

$$w^* = \arg\min_{w \in \mathbb{R}^k} R(V_{\phi,w}).$$

For clarity of exposition, we will assume this approximation is unique. The *excess risk* $\mathcal{E}(V_{\phi,\hat{w}}) = R(V_{\phi,\hat{w}}) - R(V^\pi)$ measures the additional error suffered by the approximation $V_{\phi,\hat{w}}$ compared to the true value function. We decompose it into an estimation error term, measuring the performance gap with the best-in-class, and an approximation error term arising from considering a restricted set of k-dimensional value function approximations:

$$\mathcal{E}(V_{\phi,\hat{w}}) = \underbrace{R(V_{\phi,\hat{w}}) - R(V_{\phi,w^*})}_{\text{estimation error}} + \underbrace{R(V_{\phi,w^*}) - R(V^\pi)}_{\text{approximation error}}.$$

2.2 The Successor Representation

The successor representation (Dayan, 1993) describes a state in terms of the frequency at which it visits future states; it is also related to the fundamental matrix in the study of Markov chains see Kemeny and Snell (1961); Brémaud (2013); Grinstead and Snell (2012).

Definition 1. *The successor representation (SR) with respect to a policy π for a state $s \in \mathcal{S}$ is the expected discounted sum of future occupancies for each state $s' \in \mathcal{S}$. Specifically, $\psi^\pi(s) = (\psi^\pi(s, s'))_{s' \in \mathcal{S}}$, where*

$$\psi^\pi(s, s') = \mathop{\mathbb{E}}_{\pi, \mathcal{P}} \left[\sum_{t=0}^{\infty} \gamma^t \mathbb{I}\left[s_t = s'\right] \mid s_0 = s \right].$$

Expressed as a matrix $\Psi^\pi \in \mathbb{R}^{S \times S}$, the successor representation can be written as:

$$\Psi^\pi = \left(I - \gamma P_\pi\right)^{-1}.$$

As a consequence of the Bellman equation, we can express the value function in terms of the successor representation as follows:

$$V^\pi = \Psi^\pi r_\pi.$$

This makes it a particularly appealing candidate to use as a state representation. In particular, it is well-established that the top eigenvectors (Mahadevan and Maggioni, 2007) or singular vectors (Behzadian and Petrik, 2018) of the successor representation form a useful representation (Stachenfeld et al., 2014). Petrik (2007) derived an analytical bound on the approximation error for linear value function approximation for a representation made of the top eigenvectors of Ψ^π in the particular setting where P_π is symmetric. By contrast, in this paper, we consider the more general setting of an arbitrary transition matrix P_π and consider a generalization bound that accounts for the statistical nature of the learning process.

3 CHARACTERIZING EXCESS RISK

Our first result characterizes how the choice of representation affects the generalization of value functions. Theorem 1 applies beyond the setting of reinforcement

[1] Results for a larger class of distributions are given in Appendix A

learning, and more generally characterizes the excess risk of a broad class of least-squares regression problems.

To begin, we assume that the labels $y_1, \ldots, y_n$ satisfy

$$y_i = V(s_i) + \eta_i,$$

where $V : \mathcal{S} \to \mathbb{R}$ and η_i is i.i.d. zero mean σ-sub-Gaussian noise (Vershynin, 2010). This includes the batch Monte Carlo setting, in which case $V = V^\pi$ and $\eta_i \stackrel{D}{=} G^\pi(s_i) - V^\pi(s_i)$, where $G^\pi(s_i)$ is the random return from s_i.

For a feature matrix Φ, we write P_Φ for the orthogonal projector onto its column space, and $P_\Phi^\perp$ for the orthogonal projector onto the corresponding nullspace. We have

$$P_\Phi = \Phi(\Phi^\mathsf{T}\Phi)^{-1}\Phi^\mathsf{T} \quad P_\Phi^\perp = I_S - P_\Phi.$$

In particular, the approximation error for a given state representation ϕ is

$$R(V_{\phi,w^*}) - R(V) = \|P_\Phi^\perp V\|_{S,2}^2.$$

A key quantity in our analysis is the notion of the *effective dimension* of a state representation, which dictates the number of samples required to achieve a low estimation error.

Definition 2 (Effective dimension)**.** *Let $\Phi \in \mathbb{R}^{S\times k}$ be a feature matrix. The effective dimension of Φ (vis-a-vis the standard basis (e_i)) is defined as the quantity*

$$d_{\text{eff}}(\Phi) := S \max_{i=1,\ldots,S} \|P_\Phi e_i\|_2^2,$$

where P_Φ is the orthogonal projector onto the column space of Φ.

It is simple to check that the effective dimension is only a function of the column space of Φ and that $d_{\text{eff}}(\Phi)$ satisfies

$$\text{rank}(\Phi) \leqslant d_{\text{eff}}(\Phi) \leqslant S.$$

Our notion of effective dimension is derived from the *coherence* of Φ, defined as

$$\mu(\Phi) = \frac{d_{\text{eff}}}{\text{rank}(\Phi)}.$$

The notion of coherence is from Candès and Recht (2009), who demonstrate that coherence can be used to characterize the feasibility of low-rank matrix recovery. Informally, $\mu(\Phi)$ (and $d_{\text{eff}}(\Phi)$) measure the (lack of) sparsity of the column space of Φ. At one extreme, if $\Phi \in \mathbb{R}^{S\times 1}$ is the all-ones vector, then $d_{\text{eff}}(\Phi) = \text{rank}(\Phi)$, saturating the lower bound. On the other hand, if $\Phi = e_i$ for some $i \in \{1, \ldots, S\}$ then $d_{\text{eff}}(\Phi) = S$, saturating the upper bound. As we now show, the effective dimension of Φ can be used to bound the excess risk of least-squares regression applied to the state representation ϕ.

Theorem 1 (Excess risk)**.** *Fix any $\delta \in (0, 1)$. Suppose that $n \geqslant 8 d_{\text{eff}}(\Phi)\log(6k/\delta)$. With probability at least $1-\delta$, the empirical risk minimizer $V_{\phi,\hat{w}}$ satisfies:*

$$\mathcal{E}(V_{\phi,\hat{w}}) \leqslant \|P_\Phi^\perp V\|_{S,2}^2 + 384c\frac{d_{\text{eff}}(\Phi)}{n}\|P_\Phi^\perp V\|_{S,2}^2 + 48\sigma^2\frac{2k+3c}{n} + \frac{64}{3}\frac{d_{\text{eff}}(\Phi)}{n^2}\|P_\Phi^\perp V\|_\infty^2 c^2,$$

where $c = \log(3/\delta)$ and $\|\cdot\|_\infty$ denotes the usual supremum norm.

Proof. The proof is given in Appendix A, and follows arguments for the analysis of random design linear least-squares problems (Hsu et al., 2012b) and matrix concentration inequalities (Tropp, 2015). The result can also be obtained by instantiating Theorem 1 of Hsu et al. (2012b) to our setting, at the cost of added complexity. □

In Theorem 1, the term $\|P_\Phi^\perp V\|_{S,2}^2$ is the approximation error and reflects the error due to using a k-dimensional linear approximation. The remainder of the bound corresponds to the estimation error. The theorem demonstrates that the ability of a representation to generalize is quantified not only by the approximation error but also the effective dimension $d_{\text{eff}}(\Phi)$. Not only does $d_{\text{eff}}(\Phi)$ appear in the bound, but it also dictates a minimum number of samples needed to obtain a high probability bound: when $d_{\text{eff}}(\Phi)$ is small, the bound holds for fewer samples.

In the specific context of batch Monte Carlo policy evaluation, Theorem 1 holds as-is with $V = V^\pi$. Additionally, the noise variance σ^2 can be bounded as

$$\sigma^2 \leqslant \frac{V_{\max}^2}{4}.$$

The term $\|P_\Phi e_i\|_2^2$ that drives the effective dimension of Φ differs (for non orthogonal representations Φ) from the quantity $\max_i \|\phi(s_i)\|_2^2$ that appears in Rademacher complexity bounds for regression in the case of a family of linear predictors (Mohri et al., 2018) (see also Maillard and Munos (2009)). Compared to such bounds, Theorem 1 is also sharper for all representations as it offers a $O(1/n)$ dependency rather than $O(1/\sqrt{n})$. In subsequent sections, we will provide empirical evidence illustrating how the effective dimension plays a critical role in determining the generalization capability of ϕ.

3.1 Illustrative Examples

To understand how the bound is instantiated in particular settings, consider first the scenario in which $\Phi = I_S$ is the tabular representation. This corresponds to using the feature vector $e_i \in \mathbb{R}^S$ for the i-th state.

In this case, the approximation error is 0 and the estimation error reduces to the classic $\sigma^2 S/n$ rate for least-squares regression:

$$R(V_{\phi,\hat{w}}) - R(V) \lesssim \frac{\sigma^2(S + \log(1/\delta))}{n}.$$

With this choice of features, good generalization requires a number of samples n linear in S.

At the other extreme, it is possible to improve the sample complexity to avoid the dependency on S. In the ideal case, $d_{\text{eff}}(\Phi) = k$. In the next section we will demonstrate that, in environments with a particular transition structure, representations derived from the successor representation achieve this bound.

To make this argument more concrete, suppose that we have a family $(\phi_k)_{k=1}^S$ of representations (resp. matrices (Φ_k)) whose effective dimension satisfies $d_{\text{eff}}(\Phi_k) \approx k$. Furthermore, assume that the approximation error $\|P_{\Phi_k}^{\perp} V\|_{S,2}^2$ scales as $\psi(k)$, where $\psi(k)$ is a monotonically decreasing function of k. Fix $\varepsilon > 0$ and define $\bar{k} = \bar{k}(\varepsilon) := \min\{k : \psi(k) \leqslant \varepsilon\}$, and let $\bar{w}$ be the weight vector found by least-squares regression applied with $\phi_{\bar{k}}$. Observe that as long as n satisfies:

$$n \gtrsim \max\left\{\max\left\{\frac{\sigma^2}{\varepsilon}, 1\right\}\bar{k}(\varepsilon)\log\frac{\bar{k}(\varepsilon)}{\delta}, \sqrt{\bar{k}(\varepsilon)S}\log\frac{1}{\delta}\right\},$$

then we have $\mathcal{E}(V_{\phi_{\bar{k}},\bar{w}}) \leqslant 4\varepsilon$. As a particular example, let $\psi(k) = \rho^k$ for some $\rho \in (0,1)$. Then $\bar{k}(\varepsilon) \leqslant \lceil \frac{1}{1-\rho}\log\left(\frac{1}{\varepsilon}\right)\rceil$, in which case the sample complexity only depends sublinearly on S.

4 GENERALIZATION FOR THE SUCCESSOR REPRESENTATION

An effective approach for constructing a family of representations is to take the k singular vectors of the successor representation (SR) whose singular values are the greatest. For a given policy π, let Ψ^π be the successor representation for π. We write

$$\Psi^\pi = F\Sigma B^\top,$$

where $F, B \in \mathbb{R}^{S\times S}$ are matrices whose columns are orthogonal and have unit norm. Additionally, $\Sigma = \text{diag}(\sigma_1, ..., \sigma_S)$ where σ_i are the singular values of Ψ sorted in decreasing order.

For a fixed integer k satisfying $1 \leqslant k \leqslant S$, let us partition F into two matrices, $F_k \in \mathbb{R}^{S\times k}$ and $F_k^{\perp}$, which respectively contains the top k and bottom $S-k$ columns of F. Correspondingly, we partition Σ into $\Sigma_k \in \mathbb{R}^{k\times k}$ and $\Sigma_k^{\perp}$ and B into B_k and $B_k^{\perp}$. With this notation, we obtain the family of state representations (expressed as feature matrices) $\Phi_k = F_k$.

4.1 Approximation Error: $\|P_{\Phi}^{\perp} V^\pi\|_{S,2}^2$

Given a reward vector $r_\pi \in \mathbb{R}^S$, the value function $V^\pi \in \mathbb{R}^S$ is given by $V^\pi = \Psi^\pi r_\pi$. As demonstrated by Theorem 1, the first key quantity that appears in the generalization bound is the approximation error $\|P_{F_k}^{\perp} V^\pi\|_{S,2}^2$. With the successor representation, we can write:

$$\|P_{F_k}^{\perp} V^\pi\|_{S,2}^2 = \|P_{F_k}^{\perp}\Psi^\pi r_\pi\|_{S,2}^2 = \|F_k^{\perp}\Sigma_k^{\perp}(B_k^{\perp})^\top r_\pi\|_{S,2}^2.$$

Following the argument from Petrik (2007) for the specific case of proto-value functions (Mahadevan and Maggioni, 2007), the worst-case unit-norm reward vector r_π in this case approximately corresponds to the $(k+1)$-th vector b_{k+1}. This is because

$$F_k^{\perp}\Sigma_k^{\perp}(B_k^{\perp})^\top b_{k+1} = f_{k+1}\sigma_{k+1},$$

and the fact that $\sigma_{k+1} \geqslant \sigma_{k+i}$, for all $i \geqslant 1$. To make the bound comparable for different k and MDPs, let us fix $R_{\max}$ and write

$$r_\pi = \frac{b_{k+1}R_{\max}}{\|b_{k+1}\|_\infty}. \tag{2}$$

In this case, since $\|f_{k+1}\|_2^2 = 1$, we have that

$$\|P_{F_k}^{\perp} V^\pi\|_{S,2}^2 \leqslant \frac{\sigma_{k+1}^2 R_{\max}^2}{S\|b_{k+1}\|_\infty^2} \leqslant \sigma_{k+1}^2 R_{\max}^2.$$

The dependence on $\|b_{k+1}\|_\infty$ relates to the operator norm of Ψ from L^2 to L^∞, and illustrates how b_{k+1} is only approximately the worst-case reward vector.

A frequent scenario in reinforcement learning occurs when the reward is nonzero in a single state. Suppose that the reward vector r_π is $r_\pi = R_{\max}e_i$ for some $i \in \{i, \ldots, S\}$. Then we have that:

$$\begin{aligned}\|P_{F_k}^{\perp} V^\pi\|_{S,2}^2 &= \frac{R_{\max}^2 \text{tr}((\Sigma_k^{\perp})^2)\|(B_k^{\perp})^\top e_i\|_2^2}{S} \\ &\leqslant \frac{\sigma_{k+1}^2 R_{\max}^2 d_{\text{eff}}(B_k^{\perp})}{S}.\end{aligned}$$

When the effective dimension of $B_k^{\perp}$ is $O(S-k)$, the approximation error may be a factor $\frac{S-k}{S}$ smaller than the error for the worst-case reward vector (Equation (2)).

These arguments show that the generalization quality of a given family of representations can be partially quantified in terms of its spectrum $(\sigma_i)_{i=1}^S$. When the transition matrix is symmetric, we can bound the spectrum $(\sigma_i)_{i=1}^S$ in terms of the effective horizon implied by the discount factor. This is given by the following lemma.

Lemma 1. *Let $P \in \mathbb{R}^{|\mathcal{S}|\times|\mathcal{S}|}$ be a symmetric row stochastic matrix, and let $\gamma \in (0,1)$. Let $\sigma(\cdot)$ denote the set of singular values of a matrix. We have that:*

$$\sigma((I-\gamma P)^{-1}) \subseteq \left[\tfrac{1}{1+\gamma}, \tfrac{1}{1-\gamma}\right].$$

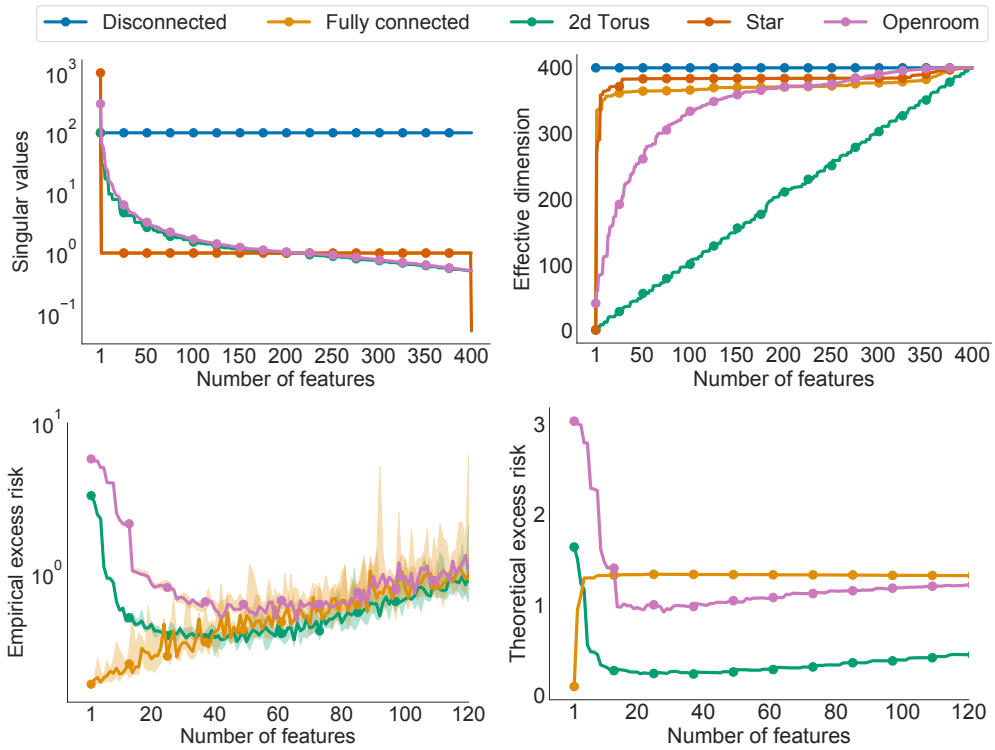

Figure 2: **Top left**: Singular values of the successor representation Ψ^π, in decreasing order and for different graphical structures (the fully connected and star graphs' spectra overlap). **Top right**: Effective dimension of the representation $\Phi_k = F_k$. **Bottom left and right**: Median empirical excess risk over 10 runs, with 95% CIs as shaded regions, and theoretical excess risk, respectively, for the open room, torus, and fully connected graphs.

Because the value function is generally of magnitude $V_{\max} = \frac{R_{\max}}{1-\gamma}$, an approximation error of order $\frac{1}{1+\gamma}$ is quite small, suggesting that the corresponding basis functions may be safely omitted from the representation.

Intuitively (and as supported by the analysis above), choosing a representation with a larger number of features k reduces the approximation error. However, as will see in the next section, a larger k necessarily increases the effective dimension, often in a manner that is superlinear in k.

4.2 Effect of Transition Structure

We next study characteristics of families of representations induced by the SVD of the successor representation for different environment transition structures. To this end, we consider different types of graphs over which we define a uniform random walk; the resulting representations are specifically proto-value functions (PVF, Mahadevan and Maggioni, 2007). We consider the two key quantities identified above: the spectrum of the representation, which informs us on the profile of the approximation error $\|P_{F_k}^\top V^\pi\|_{S,2}^2$ for different F_k, and the effective dimension of F_k as a function of k.

We consider five graphical structures, each with $S = 400$ states (illustrations of these structures as well as results for additional structures are given in the appendix): a fully-connected graph, Baird's star graph (Baird, 1995), a disconnected graph (on which each node self-transitions), a 20×20 grid, and a 20×20 torus. The torus has the same "shape" as the grid but allows transitions from one edge to its opposite, while the fully-connected graph is similar to the star graph in that both mix quickly. These graph were chosen to illustrate the diversity in generalization profiles arising from different transition structures. In all cases, $\gamma = 0.99$.

Figure 2, top left illustrates three types of spectra. The fully-connected and star structures have a flat spectrum, both with an important first component but with a last component that is much smaller in the case of the star structure (see Appendix B for a closed-form description of the spectrum of the star graph). By contrast, the grid and torus exhibit a decaying spectrum, suggesting that attaining a low approximation error may require many features. As expected, the disconnected graph produces a flat spectrum with values $\sigma_i = (1-\gamma)^{-1}$.

Figure 2, top right shows the effective dimension as a function of the number of features k, and paints a relatively different picture. Here, both star and fully-

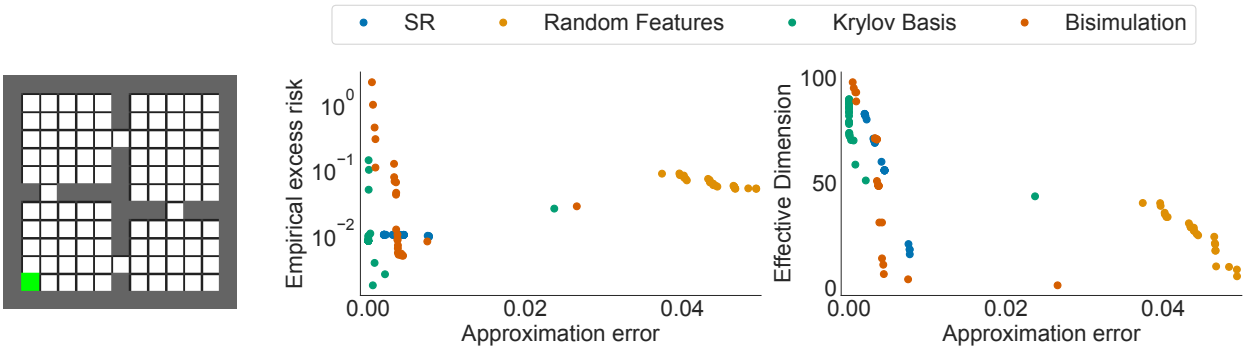

Figure 3: The four-room domain (**Left**). Median empirical excess risk (**Middle**) and effective dimension (**Right**) as a function of approximation error for the top k left singular vectors of the SR, random features, the Krylov basis and the bisimulation metric matrix in the four-room domain.

connected graphs exhibit a high effective dimension, despite having relatively simple structure. This is because effective dimension reflects in some sense the degree to which a single sample might give misleading information about the value at other states. Because the first singular vectors capture most of the symmetry in these graphs, additional features must in some sense be misleading. On the other hand, the open room and torus, despite an almost-identical spectrum, exhibit notedly different profiles: while the torus achieves the lower bound $d_{\text{eff}}(F_k) \approx k$, the grid results in generally poor features for k large.

To understand the consequences of these characteristic differences, we performed least-squares regression to estimate value functions in three of these structures (fully-connected, grid, and torus). In all cases, we sampled a reward function by assigning rewards to each state-action pair from a normal distribution (see Appendix C). We then sampled $n = 300$ states with replacement and performed a Monte Carlo rollout to obtain the sample return $(y_i)_{i=1}^n$. We measured the excess risk of the linear approximation found by the least-squares procedure. For each graph structure, we repeated the experiment 10 times.

Figure 2, bottom depicts the outcome of this experiment. Experimentally, the PVF of the torus generalizes significantly better than the PVF of the grid (left panel). This is reflected in a heuristic calculation of the theoretical bound (right panel), given more explicitly by the formula

$$\|P_{F_k}^{\perp} V^{\pi}\|_{S,2}^2 + \frac{d_{\text{eff}}(F_k)}{n} + \frac{d_{\text{eff}}(F)}{n^2} \|P_{F_k}^{\perp} V^{\pi}\|_{\infty}^2.$$

The number of features k minimizing the empirical and theoretical excess risk differ, but follow the same qualitative pattern: for small k, the open room PVF generalizes poorly, while the minimum is achieved in the fully-connected graph by $k = 1$, highlighting again its high degree of symmetry.

4.3 Analysis of the One-dimensional Torus

As evidenced by the experiments of the previous section, the proto-value functions of the two-dimensional torus have particularly appealing generalization characteristics. Analytically, similarly good generalization can be demonstrated on the one-dimensional torus, as we now show.

The one-dimensional torus consists in S states arranged on a chain, such that s_i connects to $s_{i-1}, s_{i+1} \mod S$. As such, the random walk on this torus induces a transition function P_π described by a circulant matrix. Since P_π is symmetric, we may write[2]

$$(I - \gamma P_\pi)^{-1} = U_S \Sigma U_S^*.$$

Following Gray (2006), the k-th singular value of $(I - \gamma P_\pi)^{-1}$ is given by

$$\sigma_k = \frac{1}{1 - \gamma \cos(\frac{2\pi}{S} \lceil \frac{k-1}{2} \rceil)}$$

for $k = 1, ..., S$.[3] Additionally, we have that $U_S = \frac{1}{\sqrt{S}} F_S^*$, with $(F_S)_{j,k} = \exp(-2\pi ijk/S)$ the discrete Fourier transform matrix in dimension S. From this we deduce that each entry of U_S has modulus $1/\sqrt{S}$, and therefore any orthogonal matrix formed from any k distinct columns of U_S will have coherence 1 and effective dimension k. This shows that the proto-value functions of the one-dimensional torus give in some sense an ideal state representation.

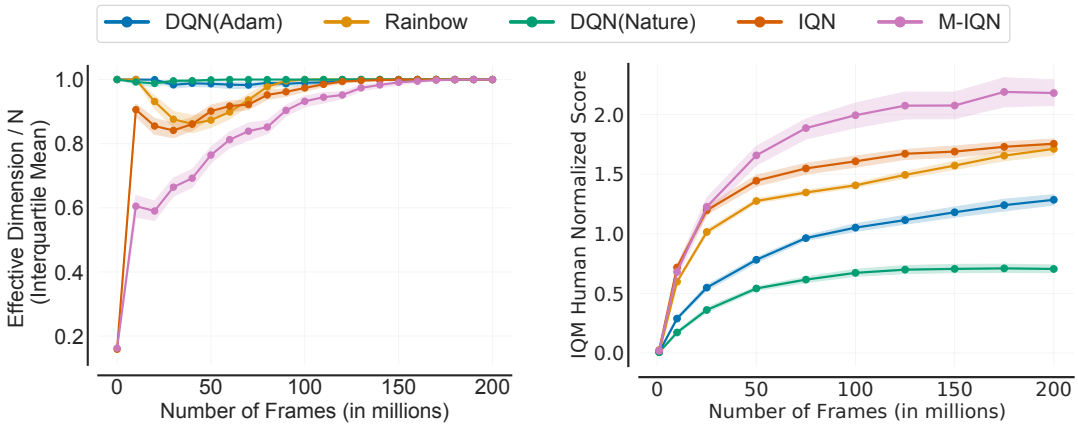

Figure 4: **Left:** Interquartile mean (IQM) (Agarwal et al., 2021b) for the effective dimension, normalized by the batch size used $N = 2^{15}$. **Right:** for human-normalized scores over the course of training across 60 Atari games. IQM measures the mean on the middle 50% of the data points combined across all runs and games. These statistics are over 5 independent runs and shading gives 95% stratified bootstrap confidence intervals based on Rliable (Agarwal et al., 2021b).

5 EXPERIMENTS

5.1 Comparing State Representations

We now compare the Successor Representation to other theoretically-motivated representations: the bisimulation metric matrix (Ferns et al., 2004), the Krylov basis (Petrik, 2007) and some random features, in terms of effective dimension and excess risk, in the setting of Section 4.2. Figure 3 shows some of these results on the four room domain (Sutton et al., 1999; Solway et al., 2014). These give further weight to the idea that effective dimension plays an important role in determining the usefulness of a representation, as for a given approximation error better effective dimension corresponds to better excess risk.

The SR of the four-room domain is fairly well-studied and have been shown to give rise to effective representations (Machado et al., 2017; Bellemare et al., 2019).It generalizes well but has worse approximation error compared to the Krylov basis or the Bisimulation metric which take into account the reward. For small approximation errors, the krylov basis has smaller effective dimension and is performing best. Finally, random features which are agnostic to the structure of the MDP have very high approximation error making them unappealing.

[2]We ignore the issue of real diagonalizable versus complex diagonalizable.

[3]The spectrum of the torus is briefly mentioned in Blier et al. (2021).

5.2 Deep Reinforcement Learning

We conclude with an empirical evaluation demonstrating the usefulness of our results in characterizing generalization in a larger setting. Specifically, we measure the effective dimensions of a representation ϕ implied by a deep neural network. We consider the hidden layer of 512 rectified linear units learnt by five deep RL agents, namely DQN (Mnih et al., 2015), DQN with Adam optimizer, Rainbow (Hessel et al., 2018), IQN (Dabney et al., 2018), and Munchausen-IQN (M-IQN) (Vieillard et al., 2020). We are interested in how the notion of effective dimension explains the relative performance of these deep RL agents aggregated across 60 Atari 2600 games (Bellemare et al., 2013) and at different points in training until 200M environment frames (Castro et al., 2018).

We compare estimates of the effective dimension of these representations throughout training and reported results in Figure 4 (Left) (see per game comparison in Appendix C.2). When computing such estimates, we use a large batch size ($=2^{15}$), sampled uniformly from the offline Atari-replay datasets (Agarwal et al., 2020), as a proxy for the ambient dimension S used in the definition of the effective dimension.

We observe that higher performance on a game typically correlates with lower effective dimension. The relative ordering of effective dimension (Figure 4, left) matches the performance ranking of different agents(Figure 4, right). Furthermore, we can notice a rise in the effective dimension from iteration 50 which suggests an overfitting of the representation to the current value function, in line with the evidence of

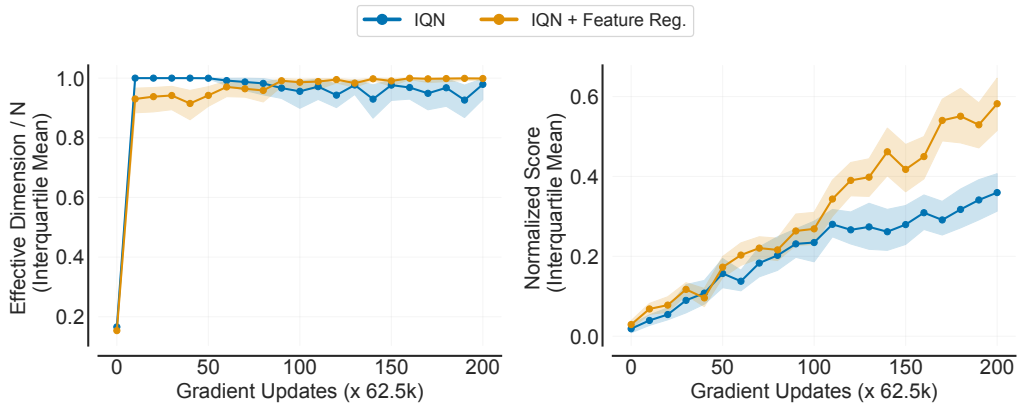

Figure 5: Effective dimension, normalized by the batch size $N = 2^{15}$ and performance of IQN and IQN with feature regularization L_ϕ on 17 Atari games in the offline RL setting.

late-training overfitting found by Dabney et al. (2020).

To further corroborate that low effective dimension corresponds to better generalization, we investigate whether optimizing an auxiliary loss $\mathcal{L}_\phi$, motivated by the idea of reducing the effective dimension of the learned representation, improves performance. To do so, we use $\mathcal{L}_\phi = \log \sum_i \exp(\|\phi(s_i)\|_2^2)$ for states s_i in a randomly sampled mini-batch of size 32. To avoid confounding effects from exploration, we study the offline RL setting (Levine et al., 2020). Specifically, we use the 5% Atari-replay dataset (Agarwal et al., 2020) on 17 games and evaluate IQN, one of the top performing agents on the offline Atari dataset (Gulcehre et al., 2020). As shown in Figure 5, right, combining IQN with the loss $\mathcal{L}_\phi$ results in significantly higher average returns compared to IQN on all 17 games. We also compare estimates of the effective dimension of the representations induced by these two agents in Figure 5, left, and find the auxiliary loss $\mathcal{L}_\phi$ results in lower effective dimension during the first 80 iterations. Surprisingly, we also notice that IQN with feature regularization prevents the substantial loss in rank of the feature matrix observed previously by Kumar et al. (2021, 2022) (see Figure 13 and Figure 12), making it hard to disentangle between approximation and estimation error effects. Further study of this phenomenon would be an interesting direction for future work.

6 CONCLUSION

In this paper we provided a theoretical characterisation of how a given representation affects generalization in reinforcement learning. While we focused here on the batch Monte Carlo setting for simplicity, a similar but more involved analysis can in theory also be performed to analyze algorithms such as LSTD.

Providing fresh evidence regarding the benefits of successor representations in shaping an agent's representation, both our analysis and experiments on synthetic environments demonstrate that indeed, the left-singular vectors of SRs generally provide good generalization. While natural given the successor representation's close relationship with the value function, one surprising result is that the effective dimension of such a representation is relatively sensitive to the particular transition structure, as illustrated by the differences between the torus and open room representations. In addition, the effective dimension of this representation does not immediately correlate with mixing time, as one might have expected. These findings suggests that it should be possible to devise algorithms inspired by the same principles, but that work well across a variety of transition structures, for example by leveraging contrastive graph representations (Madjiheurem and Toni, 2019).

Our analysis of Atari 2600-playing agents gives further evidence of the important role played by the representation in deep reinforcement learning. While not a surprise in itself, we find a strong correlation between effective dimension and performance, this suggests that generalization is key to explaining many performance improvements. In particular, it is by now well-understood that auxiliary tasks (Jaderberg et al., 2017; Bellemare et al., 2017) shape the learned representation of the agent, and under ideal conditions cause it to match the SVD of an auxiliary task matrix (Bellemare et al., 2019; Lyle et al., 2021). Controlling the bound of Theorem 1 by means of such tasks or deep learning mechanisms such as hindsight experience replay (Andrychowicz et al., 2017) may provide further performance improvements. Our results also suggest that it may be possible to derive theoretical guarantees regarding transfer between policies or MDPs (Taylor and Stone, 2009), in particular with a learned representation (Agarwal et al., 2021a).

Acknowledgements

The authors would like to thank Matthieu Geist, Mark Rowland, Pablo Samuel Castro, Ahmed Touati, Marlos Machado, Dale Schuurmans, Robert Dadashi, Tomas Vaskevicius, Olivier Pietquin, Martha White, Hanie Sedghi, Damien Vincent, Dominic Richards, Nino Vieillard, Leonard Hussenot, Amartya Sanyal, Sephora Madjiheurem, Laura Toni and the anonymous reviewers for useful discussions and feedback on this paper.

We would also like to thank the Python community (Van Rossum and Drake Jr, 1995; Oliphant, 2007) for developing tools that enabled this work, including NumPy (Oliphant, 2006; Walt et al., 2011; Harris et al., 2020), SciPy (Jones et al., 2001), Matplotlib (Hunter, 2007) and JAX (Bradbury et al., 2018).

References

Rishabh Agarwal, Dale Schuurmans, and Mohammad Norouzi. An optimistic perspective on offline reinforcement learning. In *International Conference on Machine Learning*, pages 104–114. PMLR, 2020.

Rishabh Agarwal, Marlos C. Machado, Pablo Samuel Castro, and Marc G Bellemare. Contrastive behavioral similarity embeddings for generalization in reinforcement learning. In *International Conference on Learning Representations*, 2021a.

Rishabh Agarwal, Max Schwarzer, Pablo Samuel Castro, Aaron Courville, and Marc G Bellemare. Deep reinforcement learning at the edge of the statistical precipice. *Advances in Neural Information Processing Systems*, 2021b.

Marcin Andrychowicz, Filip Wolski, Alex Ray, Jonas Schneider, Rachel Fong, Peter Welinder, Bob McGrew, Josh Tobin, Pieter Abbeel, and Wojciech Zaremba. Hindsight experience replay. *arXiv preprint arXiv:1707.01495*, 2017.

Leemon Baird. Residual algorithms: Reinforcement learning with function approximation. In *Machine Learning Proceedings 1995*, pages 30–37. Elsevier, 1995.

Bahram Behzadian and Marek Petrik. Low-rank feature selection for reinforcement learning. In *ISAIM*, 2018.

Marc Bellemare, Will Dabney, Robert Dadashi, Adrien Ali Taiga, Pablo Samuel Castro, Nicolas Le Roux, Dale Schuurmans, Tor Lattimore, and Clare Lyle. A geometric perspective on optimal representations for reinforcement learning. *Advances in neural information processing systems*, 32:4358–4369, 2019.

Marc G Bellemare, Yavar Naddaf, Joel Veness, and Michael Bowling. The arcade learning environment: An evaluation platform for general agents. *Journal of Artificial Intelligence Research*, 47:253–279, 2013.

Marc G. Bellemare, Will Dabney, and Rémi Munos. A distributional perspective on reinforcement learning. In *Proceedings of the International Conference on Machine Learning*, 2017.

Léonard Blier, Corentin Tallec, and Yann Ollivier. Learning successor states and goal-dependent values: A mathematical viewpoint. *arXiv preprint arXiv:2101.07123*, 2021.

James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, and Skye Wanderman-Milne. Jax: composable transformations of python+ numpy programs. *URL http://github. com/google/jax*, 2018.

Pierre Brémaud. *Markov chains: Gibbs fields, Monte Carlo simulation, and queues*, volume 31. Springer Science & Business Media, 2013.

Emmanuel J Candès and Benjamin Recht. Exact matrix completion via convex optimization. *Foundations of Computational mathematics*, 9(6):717–772, 2009.

Pablo S. Castro, Subhodeep Moitra, Carles Gelada, Saurabh Kumar, and Marc G. Bellemare. Dopamine: A research framework for deep reinforcement learning. *arXiv*, 2018.

Wesley Chung, Somjit Nath, Ajin Joseph, and Martha White. Two-timescale networks for nonlinear value function approximation. In *International conference on learning representations*, 2018.

Will Dabney, Georg Ostrovski, David Silver, and Rémi Munos. Implicit quantile networks for distributional reinforcement learning. In *International conference on machine learning*, pages 1096–1105. PMLR, 2018.

Will Dabney, André Barreto, Mark Rowland, Robert Dadashi, John Quan, Marc G Bellemare, and David Silver. The value-improvement path: Towards better representations for reinforcement learning. *arXiv preprint arXiv:2006.02243*, 2020.

Peter Dayan. Improving generalization for temporal difference learning: The successor representation. *Neural Computation*, 5(4):613–624, 1993.

Dibya Ghosh and Marc G Bellemare. Representations for stable off-policy reinforcement learning. In *International Conference on Machine Learning*, pages 3556–3565. PMLR, 2020.

Robert M. Gray. Toeplitz and circulant matrices: A review. 2006.

Charles Miller Grinstead and James Laurie Snell. *Introduction to probability*. American Mathematical Soc., 2012.

Caglar Gulcehre, Ziyu Wang, Alexander Novikov, Thomas Paine, Sergio Gómez, Konrad Zolna, Rishabh Agarwal, Josh S Merel, Daniel J Mankowitz,

Cosmin Paduraru, et al. Rl unplugged: A collection of benchmarks for offline reinforcement learning. *Advances in Neural Information Processing Systems*, 33, 2020.

Charles R Harris, K Jarrod Millman, Stéfan J van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J Smith, et al. Array programming with numpy. *Nature*, 585(7825):357–362, 2020.

Matteo Hessel, Joseph Modayil, Hado van Hasselt, Tom Schaul, Georg Ostrovski, Will Dabney, Dan Horgan, Bilal Piot, Mohammad Azar, and David Silver. Rainbow: Combining improvements in deep reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2018.

Daniel Hsu, Sham Kakade, and Tong Zhang. A tail inequality for quadratic forms of subgaussian random vectors. *Electronic Communications in Probability*, 17:1–6, 2012a.

Daniel Hsu, Sham M Kakade, and Tong Zhang. Random design analysis of ridge regression. In *Conference on learning theory*, pages 9–1. JMLR Workshop and Conference Proceedings, 2012b.

John D Hunter. Matplotlib: A 2d graphics environment. *Computing in science & engineering*, 9(3): 90–95, 2007.

Max Jaderberg, Volodymyr Mnih, Wojciech M. Czarnecki, Tom Schaul, Joel Z Leibo, David Silver, and Koray Kavukcuoglu. Reinforcement learning with unsupervised auxiliary tasks. In *Proceedings of the International Conference on Learning Representations*, 2017.

Eric Jones, Travis Oliphant, Pearu Peterson, et al. Scipy: Open source scientific tools for python. 2001.

John G Kemeny and J Laurie Snell. Finite continuous time markov chains. *Theory of Probability & Its Applications*, 6(1):101–105, 1961.

George D. Konidaris, Sarah Osentoski, and Philip S. Thomas. Value function approximation in reinforcement learning using the fourier basis. In *Proceedings of the 25th Conference on Artificial Intelligence*, 2011.

Aviral Kumar, Rishabh Agarwal, Dibya Ghosh, and Sergey Levine. Implicit under-parameterization inhibits data-efficient deep reinforcement learning. In *International Conference on Learning Representations*, 2021.

Aviral Kumar, Rishabh Agarwal, Tengyu Ma, Aaron Courville, George Tucker, and Sergey Levine. Dr3: Value-based deep reinforcement learning requires explicit regularization. 2022.

Nir Levine, Tom Zahavy, Daniel Mankowitz, Aviv Tamar, and Shie Mannor. Shallow updates for deep reinforcement learning. In *Advances in Neural Information Processing Systems*, 2017.

Sergey Levine, Aviral Kumar, George Tucker, and Justin Fu. Offline reinforcement learning: Tutorial, review, and perspectives on open problems. *arXiv preprint arXiv:2005.01643*, 2020.

Clare Lyle, Mark Rowland, Georg Ostrovski, and Will Dabney. On the effect of auxiliary tasks on representation dynamics. In *International Conference on Artificial Intelligence and Statistics*, pages 1–9. PMLR, 2021.

M.C. Machado, M.G. Bellemare, and M. Bowling. A Laplacian framework for option discovery in reinforcement learning. In *Proceedings of the International Conference on Machine Learning*, 2017.

Sephora Madjiheurem and Laura Toni. Representation learning on graphs: A reinforcement learning application. In *Proceedings of the International Conference on Machine Learning*, 2019.

Sridhar Mahadevan and Mauro Maggioni. Proto-value functions: A laplacian framework for learning representation and control in markov decision processes. *Journal of Machine Learning Research*, 8(10), 2007.

Odalric-Ambrym Maillard and Rémi Munos. Compressed least-squares regression. In *NIPS 2009*, 2009.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, Stig Petersen, Charles Beattie, Amir Sadik, Ioannis Antonoglou, Helen King, Dharshan Kumaran, Daan Wierstra, Shane Legg, and Demis Hassabis. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015.

Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of machine learning*. MIT press, 2018.

Travis E Oliphant. *A guide to NumPy*, volume 1. Trelgol Publishing USA, 2006.

Travis E Oliphant. Python for scientific computing. *Computing in Science & Engineering*, 9(3):10–20, 2007.

Ronald Parr, Lihong Li, Gavin Taylor, Christopher Painter-Wakefield, and Michael L Littman. An analysis of linear models, linear value-function approximation, and feature selection for reinforcement learning. In *Proceedings of the 25th international conference on Machine learning*, pages 752–759, 2008.

Marek Petrik. An analysis of laplacian methods for value function approximation in mdps. In *IJCAI*, pages 2574–2579, 2007.

Martin L Puterman. Markov decision processes: Discrete stochastic dynamic programming. 1994.

Bohdana Ratitch and Doina Precup. Sparse distributed memories for on-line value-based reinforcement learning. In *Proceedings of the 15th European Conference on Machine Learning*, 2004.

Alec Solway, Carlos Diuk, Natalia Córdova, Debbie Yee, Andrew G Barto, Yael Niv, and Matthew M Botvinick. Optimal behavioral hierarchy. *PLoS Computational Biology*, 10(8):e1003779, aug 2014.

Kimberly L. Stachenfeld, Matthew Botvinick, and Samuel J. Gershman. Design principles of the hippocampal cognitive map. In *Advances in Neural Information Processing Systems*, 2014.

Richard S Sutton. Generalization in reinforcement learning: Successful examples using sparse coarse coding. *Advances in neural information processing systems*, pages 1038–1044, 1996.

Richard S. Sutton and Andrew G. Barto. *Reinforcement learning: An introduction*. MIT Press, 2nd edition, 2018.

R.S. Sutton, D. Precup, and S. Singh. Between MDPs and semi-MDPs: A framework for temporal abstraction in reinforcement learning. *Artificial Intelligence*, 112:181–211, 1999.

Matthew E. Taylor and Peter Stone. Transfer learning for reinforcement learning domains: A survey. *Journal of Machine Learning Research*, 10(1):1633–1685, 2009.

Joel A. Tropp. An introduction to matrix concentration inequalities. *Foundations and Trends in Machine Learning*, 8, 2015.

Guido Van Rossum and Fred L Drake Jr. *Python reference manual*. Centrum voor Wiskunde en Informatica Amsterdam, 1995.

Vladimir N Vapnik. The nature of statistical learning. *Theory*, 1995.

Roman Vershynin. Introduction to the non-asymptotic analysis of random matrices. *arXiv preprint arXiv:1011.3027*, 2010.

Nino Vieillard, Olivier Pietquin, and Matthieu Geist. Munchausen reinforcement learning. *arXiv preprint arXiv:2007.14430*, 2020.

Stéfan van der Walt, S Chris Colbert, and Gael Varoquaux. The numpy array: a structure for efficient numerical computation. *Computing in science & engineering*, 13(2):22–30, 2011.

On the Generalization of Representations in Reinforcement Learning: Appendices

A PROOFS FOR SECTION 3

This section is dedicated to proving the main theorem on the paper, Theorem 1. Before that, we introduce and prove a more general result from which Theorem 1 can be deduced as a corollary.

Let $s_1, ..., s_n$ denote iid draws from an arbitrary distribution $\nu \in \mathcal{P}(\mathcal{S})$ and $(e_i)_{i=1}^S \subset \mathbb{R}^S$ the standard basis.

Assumption 1. *We assume that $\nu(s) > 0$ for all state state $s \in \{1, ..., S\}$.*

Let $N := \mathbb{E}_{i\sim\nu}[e_i e_i^\intercal]$, and let $\|x\|_{\nu,2} := \|N^{1/2}x\|_2$ for $x \in \mathbb{R}^S$. Put $\underline{\nu} := \min_{i=1,...,S} \nu_i > 0$. Let $w^* := (\Phi^\intercal N\Phi)^{-1}\Phi^\intercal NV$, and also define $\Xi := \Phi^\intercal N\Phi$. Ξ is the steady-state feature covariance matrix. w^* represents the best k-dimensional model. Since we assume that $\underline{\nu} > 0$, we have that Ξ is positive definite.

The excess risk $\mathcal{E}(V_{\phi,w})$ of a hypothesis $V_{\phi,w} : \mathcal{S} \to \mathbb{R}$ is defined as:

$$\mathcal{E}(V_{\phi,w}) := \mathbb{E}_{s_i\sim\nu}(V_{\phi,w}(s_i) - V(s_i))^2.$$

For any $\hat{w} \in \mathbb{R}^k$, we have the decomposition:

$$\mathcal{E}(V_{\phi,\hat{w}}) = \|\Phi\hat{w} - V\|_{\nu,2}^2 = \|\Phi(\hat{w} - w^*)\|_{\nu,2}^2 + \|\Phi w^* - V\|_{\nu,2}^2.$$

Note we have the identity:

$$\|\Phi w^* - V\|_{\nu,2}^2 = \|P^\perp_{N^{1/2}\Phi} N^{1/2} V\|_2^2.$$

Theorem 2. *Fix any $\delta \in (0,1)$. Suppose that $n \geqslant 8 d_{\text{eff}}(\Phi)\log(6k/\delta)$. Under Assumption 1, with probability at least $1-\delta$, the empirical risk minimizer $V_{\phi,\hat{w}}$ satisfies:*

$$\begin{aligned}\mathcal{E}(V_{\phi,\hat{w}}) = \|P^\perp_{N^{1/2}\Phi} N^{1/2} V\|_2^2 &+ 384\frac{d_{\text{eff}}(\Phi)}{\underline{\nu} nS}\|P^\perp_{N^{1/2}\Phi} N^{1/2} V\|_2^2 \log(3/\delta) \\ &+ 48\frac{\sigma^2}{n}[2k + 3\log(3/\delta)] + \frac{64}{3}\frac{d_{\text{eff}}(\Phi)}{\underline{\nu} n^2 S}\|N^{-1/2} P^\perp_{N^{1/2}\Phi} N^{1/2} V\|_\infty^2 \log^2(3/\delta).\end{aligned}$$

where $\|\cdot\|_\infty$ denotes the usual supremum norm.

Proof. The empirical risk minimizer $\hat{w} \in \mathbb{R}^k$ is defined as the random vector $\hat{w} = (E_n\Phi)^\dagger Y$. Next, we write:

$$N^{1/2}\Phi(\hat{w} - w^*) = N^{1/2}\Phi(E_n\Phi)^\dagger(E_n V + \eta) - N^{1/2}\Phi w^*$$

Therefore, assuming $E_n\Phi$ has full column rank (which will be the case by Lemma 2),

$$\begin{aligned}&N^{1/2}\Phi(E_n\Phi)^\dagger E_n V - N^{1/2}\Phi w^* \\ &= N^{1/2}\Phi(E_n\Phi)^\dagger E_n V - P_{N^{1/2}\Phi} N^{1/2} V \\ &= N^{1/2}\Phi(\Phi^\intercal E_n^\intercal E_n\Phi)^{-1}\Phi^\intercal E_n^\intercal E_n V - P_{N^{1/2}\Phi} N^{1/2} V \\ &= N^{1/2}\Phi(\Phi^\intercal E_n^\intercal E_n\Phi)^{-1}\Phi^\intercal E_n^\intercal E_n N^{-1/2}(P_{N^{1/2}\Phi} + P^\perp_{N^{1/2}\Phi}) N^{1/2} V - P_{N^{1/2}\Phi} N^{1/2} V \\ &= N^{1/2}\Phi(\Phi^\intercal E_n^\intercal E_n\Phi)^{-1}\Phi^\intercal E_n^\intercal E_n N^{-1/2} P^\perp_{N^{1/2}\Phi} N^{1/2} V \\ &\quad + N^{1/2}\Phi(\Phi^\intercal E_n^\intercal E_n\Phi)^{-1}\Phi^\intercal E_n^\intercal E_n N^{-1/2} P_{N^{1/2}\Phi} N^{1/2} V - P_{N^{1/2}\Phi} N^{1/2} V \\ &= N^{1/2}\Phi(\Phi^\intercal E_n^\intercal E_n\Phi)^{-1}\Phi^\intercal E_n^\intercal E_n N^{-1/2} P^\perp_{N^{1/2}\Phi} N^{1/2} V \\ &= N^{1/2}\Phi\Xi^{-1/2}(\Xi^{-1/2}\Phi^\intercal E_n^\intercal E_n\Phi\Xi^{-1/2})^{-1}\Xi^{-1/2}\Phi^\intercal E_n^\intercal E_n N^{-1/2} P^\perp_{N^{1/2}\Phi} N^{1/2} V.\end{aligned}$$

Similarly,

$$\begin{aligned} N^{1/2}\Phi(E_n\Phi)^\dagger\eta &= N^{1/2}\Phi(\Phi^\mathsf{T}E_n^\mathsf{T}E_n\Phi)^{-1}\Phi^\mathsf{T}E_n^\mathsf{T}\eta \\ &= N^{1/2}\Phi\Xi^{-1/2}(\Xi^{-1/2}\Phi^\mathsf{T}E_n^\mathsf{T}E_n\Phi\Xi^{-1/2})^{-1}\Xi^{-1/2}\Phi^\mathsf{T}E_n^\mathsf{T}\eta. \end{aligned}$$

We first claim that $\|N^{1/2}\Phi\Xi^{-1/2}\|_{\mathrm{op}} \leqslant 1$. To see this, observe that:

$$\|N^{1/2}\Phi\Xi^{-1/2}\|_{\mathrm{op}}^2 = \lambda_{\max}(N^{1/2}\Phi(\Phi^\mathsf{T}N\Phi)^{-1}\Phi^\mathsf{T}N^{1/2}) = \lambda_{\max}(P_{N^{1/2}\Phi}) \leqslant 1.$$

Hence:

$$\|N^{1/2}\Phi(E_n\Phi)^\dagger E_nV - N^{1/2}\Phi w_*\|_2 \leqslant \frac{\|\Xi^{-1/2}\Phi^\mathsf{T}E_n^\mathsf{T}E_nN^{-1/2}P^\perp_{N^{1/2}\Phi}N^{1/2}V\|_2}{\lambda_{\min}(\Xi^{-1/2}\Phi^\mathsf{T}E_n^\mathsf{T}E_n\Phi\Xi^{-1/2})},$$

and similarly

$$\|N^{1/2}\Phi(E_n\Phi)^\dagger\eta\|_2 \leqslant \frac{\|\Xi^{-1/2}\Phi^\mathsf{T}E_n^\mathsf{T}\eta\|_2}{\lambda_{\min}(\Xi^{-1/2}\Phi^\mathsf{T}E_n^\mathsf{T}E_n\Phi\Xi^{-1/2})}.$$

Therefore,

$$\|N^{1/2}\Phi(\hat{w}-w^*)\|_2 \leqslant \frac{1}{\lambda_{\min}(\Xi^{-1/2}\Phi^\mathsf{T}E_n^\mathsf{T}E_n\Phi\Xi^{-1/2})}[\|\Xi^{-1/2}\Phi^\mathsf{T}E_n^\mathsf{T}E_nN^{-1/2}P^\perp_{N^{1/2}\Phi}N^{1/2}V\|_2 + \|\Xi^{-1/2}\Phi^\mathsf{T}E_n^\mathsf{T}\eta\|_2]$$

By Lemma 2, as long as $n \geqslant \frac{8d_{\mathrm{eff}}(\Phi)}{\underline{\nu}S}\log(6k/\delta)$, then with probability at least $1-\delta/3$,

$$\frac{n}{2}I_k \preccurlyeq \Xi^{-1/2}\Phi^\mathsf{T}E_n^\mathsf{T}E_n\Phi\Xi^{-1/2} \preccurlyeq 4nI_k.$$

Furthermore, by Lemma 3, with probability at least $1-\delta/3$,

$$\begin{aligned} &\|\Xi^{-1/2}\Phi^\mathsf{T}E_n^\mathsf{T}E_nN^{-1/2}P^\perp_{N^{1/2}\Phi}N^{1/2}V\|_2 \\ &\leqslant 2\sqrt{\frac{8nd_{\mathrm{eff}}(\Phi)}{\underline{\nu}S}\|P^\perp_{N^{1/2}\Phi}N^{1/2}V\|_2^2\log\left(\frac{3}{\delta}\right)} + \frac{4}{3}\sqrt{\frac{d_{\mathrm{eff}}(\Phi)}{\underline{\nu}S}}\|N^{-1/2}P^\perp_{N^{1/2}\Phi}N^{1/2}V\|_\infty\log\left(\frac{3}{\delta}\right). \end{aligned}$$

Finally, by Lemma 4, with probability at least $1-\delta/3$,

$$\mathbf{1}\left\{\Xi^{-1/2}\Phi^\mathsf{T}E_n^\mathsf{T}E_n\Phi\Xi^{-1/2} \preccurlyeq 4nI_k\right\}\cdot\|\Xi^{-1/2}\Phi^\mathsf{T}E_n^\mathsf{T}\eta\|_2 \leqslant \sqrt{\sigma^2n[8k+12\log(3/\delta)]}.$$

Therefore, by a union bound, with probability at least $1-\delta$,

$$\begin{aligned} \|N^{1/2}\Phi(\hat{w}-w^*)\|_2 &\leqslant \frac{2}{n}\left[2\sqrt{\frac{8nd_{\mathrm{eff}}(\Phi)}{\underline{\nu}S}\|P^\perp_{N^{1/2}\Phi}N^{1/2}V\|_2^2\log\left(\frac{3}{\delta}\right)}\right] \\ &\quad + \frac{2}{n}\left[\frac{4}{3}\sqrt{\frac{d_{\mathrm{eff}}(\Phi)}{\underline{\nu}S}}\|N^{-1/2}P^\perp_{N^{1/2}\Phi}N^{1/2}V\|_\infty\log\left(\frac{3}{\delta}\right)\right] + \frac{2}{n}\left[\sqrt{\sigma^2n[8k+12\log(3/\delta)]}\right] \\ &= 4\sqrt{8}\sqrt{\frac{d_{\mathrm{eff}}(\Phi)}{\underline{\nu}nS}\log(3/\delta)}\|P^\perp_{N^{1/2}\Phi}N^{1/2}V\|_2 + 4\sqrt{\frac{\sigma^2}{n}[2k+3\log(3/\delta)]} \\ &\quad + \frac{8}{3}\frac{\sqrt{\frac{d_{\mathrm{eff}}(\Phi)}{\underline{\nu}S}}}{n}\|N^{-1/2}P^\perp_{N^{1/2}\Phi}N^{1/2}V\|_\infty\log\left(\frac{3}{\delta}\right). \end{aligned}$$

Now, from the inequality $(a+b+c)^2 \leqslant 3\left(a^2+b^2+c^2\right)$ for any $a,b,c\in\mathbb{R}$, it follows that

$$\begin{aligned} \mathcal{E}(V_{\phi,\hat{w}}) &= \|P^\perp_{N^{1/2}\Phi}N^{1/2}V\|_2^2 + 384\frac{d_{\mathrm{eff}}(\Phi)}{\underline{\nu}nS}\|P^\perp_{N^{1/2}\Phi}N^{1/2}V\|_2^2\log(3/\delta) \\ &\quad + 48\frac{\sigma^2}{n}[2k+3\log(3/\delta)] + \frac{64}{3}\frac{d_{\mathrm{eff}}(\Phi)}{\underline{\nu}n^2S}\|N^{-1/2}P^\perp_{N^{1/2}\Phi}N^{1/2}V\|_\infty^2\log^2(3/\delta). \end{aligned}$$

□

Lemma 2. *Let $\Phi \in \mathbb{R}^{S\times k}$. Let ν denote a distribution over $\{1,...,S\}$ satisfying Assumption 1 and $(e_i)_{i=1}^S \subset \mathbb{R}^S$ the standard basis. Let $s_1,...,s_n$ denote iid draws from ν. Define $Y_n \in \mathbb{R}^{k\times k}$ as:*

$$Y_n = \sum_{i=1}^n \Xi^{-1/2}\Phi^\mathsf{T} e_{s_i} e_{s_i}^\mathsf{T} \Phi \Xi^{-1/2}.$$

Fix any $\delta \in (0,1)$. As long as $n \geqslant \frac{8 d_{\text{eff}}(\Phi)}{\underline{\nu} S} \log(2k/\delta)$, with probability at least $1-\delta$,

$$\frac{n}{2} I_k \preccurlyeq Y_n \preccurlyeq 4n I_k.$$

where for two symmetric matrices, $A \preccurlyeq B$ means that the matrice $B - A$ is positive semi-definite.

Proof. This is an application of the Matrix Chernoff inequality. First, we see that $\mathbb{E}[Y_n] = nI_k$. Next, we have:

$$\begin{aligned}
\max_{i=1,...,S} \lambda_{\max}(\Xi^{-1/2}\Phi^\mathsf{T} e_i e_i^\mathsf{T} \Phi \Xi^{-1/2}) &= \max_{i=1,...,S} \|\Xi^{-1/2}\Phi^\mathsf{T} e_i\|_2^2 \\
&= \max_{i=1,...,S} \|(\Phi^\mathsf{T} N \Phi)^{-1/2}\Phi^\mathsf{T} e_i\|_2^2 \\
&\leqslant \frac{1}{\underline{\nu}} \max_{i=1,...,S} \|P_\Phi e_i\|_2^2 \\
&\leqslant \frac{d_{\text{eff}}(\Phi)}{\underline{\nu} S}.
\end{aligned}$$

We now make two applications of the Matrix Chernoff inequality (see Theorem 5.1.1 in Tropp (2015)). Denoting e as Euler's number, for the upper tail, we have that for any $t \geqslant e$,

$$\mathbb{P}(\lambda_{\max}(Y_n) \geqslant tn) \leqslant k(e/t)^{tn\underline{\nu} S / d_{\text{eff}}(\Phi)}.$$

Setting $t = 4$, we conclude that as long as $n \geqslant \frac{1}{4\log(4/e)} \frac{d_{\text{eff}}(\Phi)}{\underline{\nu} S} \log(2k/\delta)$, then we have that with probability at least $1-\delta/2$, $\lambda_{\max}(Y_n) \leqslant 4n$. For the lower tail, we have that for any $t \in (0,1)$,

$$\mathbb{P}(\lambda_{\min}(Y_n) \leqslant tn) \leqslant k \exp\left(-(1-t)^2 \frac{n}{2} \frac{\underline{\nu} S}{d_{\text{eff}}(\Phi)}\right).$$

Setting $t = 0.5$, we see that as long as $n \geqslant 8 \frac{d_{\text{eff}}(\Phi)}{\underline{\nu} S} \log(2k/\delta)$, then $\lambda_{\min}(Y_n) \geqslant n/2$ with probability at least $1-\delta/2$. Taking a union bound yields the claim. □

Lemma 3. *Put $z_n := \Xi^{-1/2}\Phi^\mathsf{T} E_n^\mathsf{T} E_n N^{-1/2} P^\perp_{N^{1/2}\Phi} N^{1/2} V$. Fix any $\delta \in (0, e^{-1/8})$. With probability at least $1-\delta$,*

$$\|z_n\|_2 \leqslant 2\sqrt{\frac{8n d_{\text{eff}}(\Phi)}{\underline{\nu} S}} \|P^\perp_{N^{1/2}\Phi} N^{1/2} V\|_2 \sqrt{\log(1/\delta)} + \frac{4}{3}\sqrt{\frac{d_{\text{eff}}(\Phi)}{\underline{\nu} S}} \|N^{-1/2} P^\perp_{N^{1/2}\Phi} N^{1/2} V\|_\infty \log(1/\delta).$$

Proof. Define $q_i := \Xi^{-1/2}\Phi^\mathsf{T} e_{s_i} e_{s_i}^\mathsf{T} N^{-1/2} P^\perp_{N^{1/2}\Phi} N^{1/2} V$. We have that $\mathbb{E}[q_i] = 0$. Next,

$$\begin{aligned}
\mathbb{E}[\|q_i\|_2^2] &= \mathbb{E}[\|\Xi^{-1/2}\Phi^\mathsf{T} e_{s_i}\|_2^2 \langle e_{s_i}, N^{-1/2} P^\perp_{N^{1/2}\Phi} N^{1/2} V\rangle^2] \\
&\leqslant \frac{d_{\text{eff}}(\Phi)}{\underline{\nu} S} \mathbb{E}[\langle e_{s_i}, N^{-1/2} P^\perp_{N^{1/2}\Phi} N^{1/2} V\rangle^2] \\
&= \frac{d_{\text{eff}}(\Phi)}{\underline{\nu} S} \|P^\perp_{N^{1/2}\Phi} N^{1/2} V\|_2^2.
\end{aligned}$$

Finally, we have the following almost sure bound:

$$\|q_i\|_2 \leqslant \sqrt{\frac{d_{\text{eff}}(\Phi)}{\underline{\nu} S}} \|N^{-1/2} P^\perp_{N^{1/2}\Phi} N^{1/2} V\|_\infty.$$

Put $z_n := \sum_{i=1}^n q_i$. By the vector Bernstein inequality, for all $t > 0$,

$$\mathbb{P}\left(\|z_n\|_2 > \sqrt{\frac{nd_{\text{eff}}(\Phi)}{\underline{\nu} S}}\|P^{\perp}_{N^{1/2}\Phi} N^{1/2} V\|_2^2(1+\sqrt{8t}) + \frac{4}{3}\sqrt{\frac{d_{\text{eff}}(\Phi)}{\underline{\nu} S}}\|N^{-1/2} P^{\perp}_{N^{1/2}\Phi} N^{1/2} V\|_\infty t\right) \leqslant e^{-t}.$$

The claim now follows by setting $t = \log(1/\delta)$. □

Lemma 4. *Let $\mathcal{G}$ be the event:*

$$\mathcal{G} := \left\{\Xi^{-1/2}\Phi^\mathsf{T} E_n^\mathsf{T} E_n \Phi \Xi^{-1/2} \preccurlyeq 4nI_k\right\}$$

With probability at least $1-\delta$, we have:

$$\mathbf{1}\{\mathcal{G}\} \cdot \|\Xi^{-1/2}\Phi^\mathsf{T} E_n^\mathsf{T}\eta\|_2^2 \leqslant \sigma^2 n[8k + 12\log(1/\delta)].$$

Proof. Put $M := \mathbf{1}\{\mathcal{G}\} \cdot E_n \Phi \Xi^{-1}\Phi^\mathsf{T} E_n^\mathsf{T}$. Because η is assumed to be independent of E_n, we can condition on E_n and apply the Hanson-Wright inequality (Hsu et al., 2012a) to conclude that for any $t > 0$,

$$\mathbb{P}(\eta^\mathsf{T} M\eta > \sigma^2(\text{tr}(M) + 2\sqrt{\text{tr}(M^2)t} + 2\|M\|_{\text{op}} t)) \mid E_n) \leqslant e^{-t}.$$

We now compute upper bounds on $\text{tr}(M)$, $\text{tr}(M^2)$, and $\|M\|_{\text{op}}$. First, we have:

$$\text{tr}(M) = \mathbf{1}\{\mathcal{G}\}\,\text{tr}(E_n\Phi\Xi^{-1}\Phi^\mathsf{T} E_n^\mathsf{T}) = \mathbf{1}\{\mathcal{G}\}\,\text{tr}(\Xi^{-1/2}\Phi^\mathsf{T} E_n^\mathsf{T} E_n \Phi\Xi^{-1/2}) \leqslant 4nk.$$

Next,

$$\begin{aligned}
\text{tr}(M^2) &= \mathbf{1}\{\mathcal{G}\}\,\text{tr}(E_n\Phi\Xi^{-1}\Phi^\mathsf{T} E_n^\mathsf{T} E_n\Phi\Xi^{-1}\Phi^\mathsf{T} E_n^\mathsf{T}) \\
&= \mathbf{1}\{\mathcal{G}\}\,\text{tr}(\Xi^{-1/2}\Phi^\mathsf{T} E_n^\mathsf{T} E_n\Phi\Xi^{-1/2} \cdot \Xi^{-1/2}\Phi^\mathsf{T} E_n^\mathsf{T} E_n\Phi\Xi^{-1/2}) \\
&\overset{(a)}{\leqslant} \mathbf{1}\{\mathcal{G}\}\,\text{tr}(\Xi^{-1/2}\Phi^\mathsf{T} E_n^\mathsf{T} E_n \Xi^{-1/2}\Phi)\|\Xi^{-1/2}\Phi^\mathsf{T} E_n^\mathsf{T} E_n\Phi\Xi^{-1/2}\|_{\text{op}} \\
&\leqslant 4nk \cdot 4n = 16n^2 k.
\end{aligned}$$

Above, (a) follows from Hölder's inequality. Finally,

$$\|M\|_{\text{op}} = \mathbf{1}\{\mathcal{G}\}\|E_n\Phi\Xi^{-1}\Phi^\mathsf{T} E_n^\mathsf{T}\|_{\text{op}} = \mathbf{1}\{\mathcal{G}\}\|\Xi^{-1/2}\Phi^\mathsf{T} E_n^\mathsf{T} E_n\Phi\Xi^{-1/2}\|_{\text{op}} \leqslant 4n.$$

We now plug these bounds in along with the choice of $t = \log(1/\delta)$, which tells us that conditioned on E_n, with probability at least $1-\delta$,

$$\begin{aligned}
\eta^\mathsf{T} M\eta &\leqslant \sigma^2\left[4nk + 8n\sqrt{k\log(1/\delta)} + 8n\log(1/\delta)\right] \\
&\leqslant \sigma^2\left[8nk + 12n\log(1/\delta)\right] \\
&= \sigma^2 n\left[8k + 12\log(1/\delta)\right].
\end{aligned}$$

We now remove the conditioning on E_n. Let $\bar{t} := \sigma^2 n\left[8k + 12\log(1/\delta)\right]$. By the tower property,

$$\mathbb{P}(\eta^\mathsf{T} M\eta \geqslant \bar{t}) = \mathbb{E}[\mathbf{1}\{\eta^\mathsf{T} M\eta \geqslant \bar{t}\}] = \mathbb{E}[\mathbb{E}[\mathbf{1}\{\eta^\mathsf{T} M\eta \geqslant \bar{t}\} \mid E_n]] = \mathbb{E}[\mathbb{P}(\eta^\mathsf{T} M\eta \geqslant \bar{t} \mid E_n)] \leqslant \mathbb{E}[\delta] = \delta.$$

□

Theorem 1 is a corollary of Theorem 2 in the case where the distribution ν is uniform.

Theorem 1 (Excess risk)**.** *Fix any $\delta \in (0,1)$. Suppose that $n \geqslant 8d_{\text{eff}}(\Phi)\log(6k/\delta)$. With probability at least $1-\delta$, the empirical risk minimizer $V_{\phi,\hat{w}}$ satisfies:*

$$\begin{aligned}
\mathcal{E}(V_{\phi,\hat{w}}) &\leqslant \|P^{\perp}_{\Phi} V\|^2_{S,2} + 384c\frac{d_{\text{eff}}(\Phi)}{n}\|P^{\perp}_{\Phi} V\|^2_{S,2} \\
&+ 48\sigma^2\frac{2k+3c}{n} + \frac{64}{3}\frac{d_{\text{eff}}(\Phi)}{n^2}\|P^{\perp}_{\Phi} V\|^2_\infty c^2,
\end{aligned}$$

where $c = \log(3/\delta)$ and $\|\cdot\|_\infty$ denotes the usual supremum norm.

Proof. ν being uniform, we have $\underline{\nu} = S$. The result follows by plugging $\underline{\nu}$ in Theorem 2. □

B PROOFS FOR SECTION 4

Lemma 1. *Let $P \in \mathbb{R}^{|\mathcal{S}|\times|\mathcal{S}|}$ be a symmetric row stochastic matrix, and let $\gamma \in (0,1)$. Let $\sigma(\cdot)$ denote the set of singular values of a matrix. We have that:*

$$\sigma((I-\gamma P)^{-1}) \subseteq \left[\tfrac{1}{1+\gamma}, \tfrac{1}{1-\gamma}\right].$$

Proof. Let $\lambda(\cdot)$ denote the eigenvalues of a matrix. Because P is symmetric, we have that:

$$\sigma((I-\gamma P)^{-1}) = \left\{\frac{1}{1-\gamma\lambda} : \lambda \in \lambda(P)\right\}.$$

Because P is a row stochastic matrix, we have that the spectral radius of P satisfies $\rho(P) = 1$, and therefore $\lambda(P) \subseteq [-1,1]$. Hence:

$$\frac{1}{1-\gamma\lambda} \in [1/(1+\gamma), 1/(1-\gamma)].$$

□

Eigenstructure of the Star Graph (Subsection 4.2)
A random walk on the Star graph induces a rank-two transition matrix $P_\pi \in \mathbb{R}^S$. We may write $P_\pi = v_1 e_S^\mathsf{T} + \frac{e_S v^\mathsf{T}}{S-1}$ where v is an all-ones vector except on its last coordinate where it takes value 0 and e_S a one-hot vector taking value 1 on its last coordinate. It is easy to prove by induction that

- for any $k \geqslant 1, P_\pi^{2k} = \frac{vv^\mathsf{T}}{S-1} + e_S e_S^\mathsf{T}$
- for any $k \geqslant 0, P_\pi^{2k+1} = P_\pi$

From this, it follows that

$$\begin{aligned}
(I-\gamma P_\pi)^{-1} &= I + \sum_{t=1}^{\infty}(\gamma P_\pi)^t \\
&= I + \sum_{2k\geqslant 2}\gamma^{2k}P_\pi^{2k} + \sum_{2k+1\geqslant 1}\gamma^{2k+1}(P_\pi)^{2k+1} \\
&= I + \sum_{2k\geqslant 2}\gamma^{2k}\left(\frac{vv^\mathsf{T}}{S-1} + e_S e_S^\mathsf{T}\right) + \sum_{2k+1\geqslant 1}\gamma^{2k+1}P_\pi \\
&= I + \frac{\gamma^2}{1-\gamma^2}\left(\frac{vv^\mathsf{T}}{S-1} + e_S e_S^\mathsf{T}\right) + \frac{\gamma}{1-\gamma^2}P_\pi.
\end{aligned}$$

Define $\eta := \frac{\gamma}{1-\gamma^2}$. The non-zero singular values of $(I-\gamma P_\pi)^{-1}$ are the square roots of the eigenvalues of $A = (I-\gamma P_\pi)^{-1}\left((I-\gamma P_\pi)^{-1}\right)^\mathsf{T}$. We have

$$\begin{aligned}
A = (I-\gamma P_\pi)^{-1}\left((I-\gamma P_\pi)^{-1}\right)^\mathsf{T} &= \left(I + \gamma\eta P_\pi^2 + \eta P_\pi\right)\left(I + \gamma\eta (P_\pi^2)^\mathsf{T} + \eta P_\pi^\mathsf{T}\right) \\
&= I + B,
\end{aligned}$$

where $B := avv^\mathsf{T} + be_S e_S^\mathsf{T} + c(e_S v^\mathsf{T} + ve_S^\mathsf{T})$ with $a = \frac{2\eta\gamma+\eta^2\gamma^2}{S-1} + \eta^2$, $b = 2\eta\gamma + \eta^2\gamma^2 + \frac{\eta^2}{S-1}$ and $c = (\eta + \eta^2\gamma)\frac{S}{S-1}$.
Moreover, if $\{\lambda_1, ..., \lambda_k\}$ are the eigenvalues of B then the eigenvalues of A are $\{1+\lambda_1, ..., 1+\lambda_k\}$.

Consider the basis $\{e_S, v\}$. For any a_1, a_2,

$$\begin{aligned}
B(a_1 e_S + a_2 v) &= avv^\mathsf{T}(a_1 e_S + a_2 v) + be_S e_S^\mathsf{T}(a_1 e_S + a_2 v) + c(e_S v^\mathsf{T} + ve_S^\mathsf{T})(a_1 e_S + a_2 v) \\
&= a_1 a\langle v, e_S\rangle v + a_2 a\|v\|_2^2 v + a_1 b e_S + a_2 b\langle v, e_S\rangle e_S + c(a_1\langle v, e_S\rangle e_S + a_1 v + a_2\|v\|_2^2 e_S + a_2\langle v, e_S\rangle v) \\
&= (a_1 b + ca_1\langle v, e_S\rangle + a_2 b\langle v, e_S\rangle + a_2 c\|v\|_2^2)e_S + (a_1 a\langle v, e_S\rangle + ca_1 + a_2\langle v, e_S\rangle + a_2 a\|v\|_2^2)v.
\end{aligned}$$

Since $\|v\|_2^2 = S-1$ and $\langle v, e_S \rangle = 0$, B has the representation in $\{e_S, v\}$ as:

$$\begin{bmatrix} b & c(S-1) \\ c & a(S-1) \end{bmatrix} = \begin{bmatrix} 2\eta\gamma + \eta^2\gamma^2 + \frac{\eta^2}{S-1} & (\eta + \eta^2\gamma)S \\ (\eta + \eta^2\gamma)\frac{S}{S-1} & 2\eta\gamma + \eta^2\gamma^2 + \eta^2(S-1) \end{bmatrix} = \begin{bmatrix} \frac{\eta^2}{S-1} & (\eta + \eta^2\gamma)S \\ (\eta + \eta^2\gamma)\frac{S}{S-1} & \eta^2(S-1) \end{bmatrix} + (2\eta\gamma + \eta^2\gamma^2)I$$
$$= C + (2\eta\gamma + \eta^2\gamma^2)I$$

Hence, the eigenvalues of C are given by $\frac{1}{2}\left(\eta^2\left((S-1) + \frac{1}{S-1}\right) \pm \sqrt{\eta^4\left((S-1) + \frac{1}{S-1}\right)^2 + 4(\eta + \eta^2\gamma)^2\frac{S^2}{S-1} - 4\eta^4}\right)$.

The non-zero singular values of $(I - \gamma P_\pi)^{-1}$ are thus 1 with multiplicity $S-2$ and

$$\sqrt{\frac{1}{2}\left(\eta^2\left((S-1) + \frac{1}{S-1}\right) \pm \sqrt{\eta^4\left((S-1) + \frac{1}{S-1}\right)^2 + 4(\eta + \eta^2\gamma)^2\frac{S^2}{S-1} - 4\eta^4}\right) + 2\eta\gamma + \eta^2\gamma^2 + 1}$$

For $\gamma = 0.99$ and $S = 400$, we can check numerically that the two extreme singular values are equal to 996 and 0.05 respectively which matches the spectrum obtained for the Star graph in Figure 2.

C EMPIRICAL EVALUATION: ADDITIONAL DETAILS

C.1 Graphical Structures

In this section, we study the generalization characteristics of the representations induced by the SVD of the successor representation for several environment transition structures. We illustrate the different graphs over which we define a random walk, studied in Subsection 4.2 as well as some new ones, in Figure 6.

Our experiment consists in evaluating the value function on these different transition structures when $S = 400$ states. We consider three different reward vectors $r_\pi \in \mathbb{R}^S$: the all ones vector, the one-hot feature vector e_S, and a vector whose entries are drawn from zero-mean Gaussian distribution and normalized such that $\|r_\pi\|_\infty = 1$. We then sampled a dataset D of $n = 300$ pairs (s_i, y_i) where we performed a Monte Carlo rollout to obtain the returns $(y_i)_{i=1}^n$. The targets are the value functions induced by the random walk.

We are interested in comparing our generalization bound to the empirical excess risk on these domains. Our bound looks at the regime $n \geqslant d_{\text{eff}}(F_k)$. We choose $k \leqslant \frac{n}{2}$ as an heuristic way of achieving this. We report in Figure 7 the approximation error (Figure 7 Left), the empirical excess risk (Figure 7 Middle) and the theoretical excess risk (Figure 7 Right) obtained when using the representation $\phi = F_k$ on these different graph structures.

Star: Baird's star graph (Baird, 1995) consists in $S-1$ states which are the star corners and a state S which is the star center. A random walk on this star graph induces a transition function such that all star corners

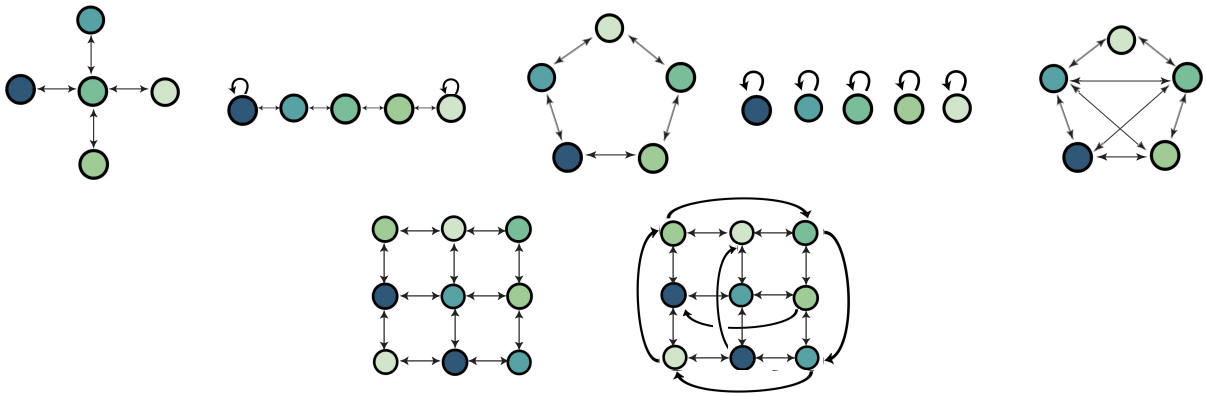

Figure 6: **Top:** Different graphical structures with $S = 5$ states from left to right, Star, Chain, Torus1d, Disconnected, Fullyconnected. **Bottom:** Two-dimensional graphical structures with $S = 9$ states: from left to right, Openroom and Torus2d.

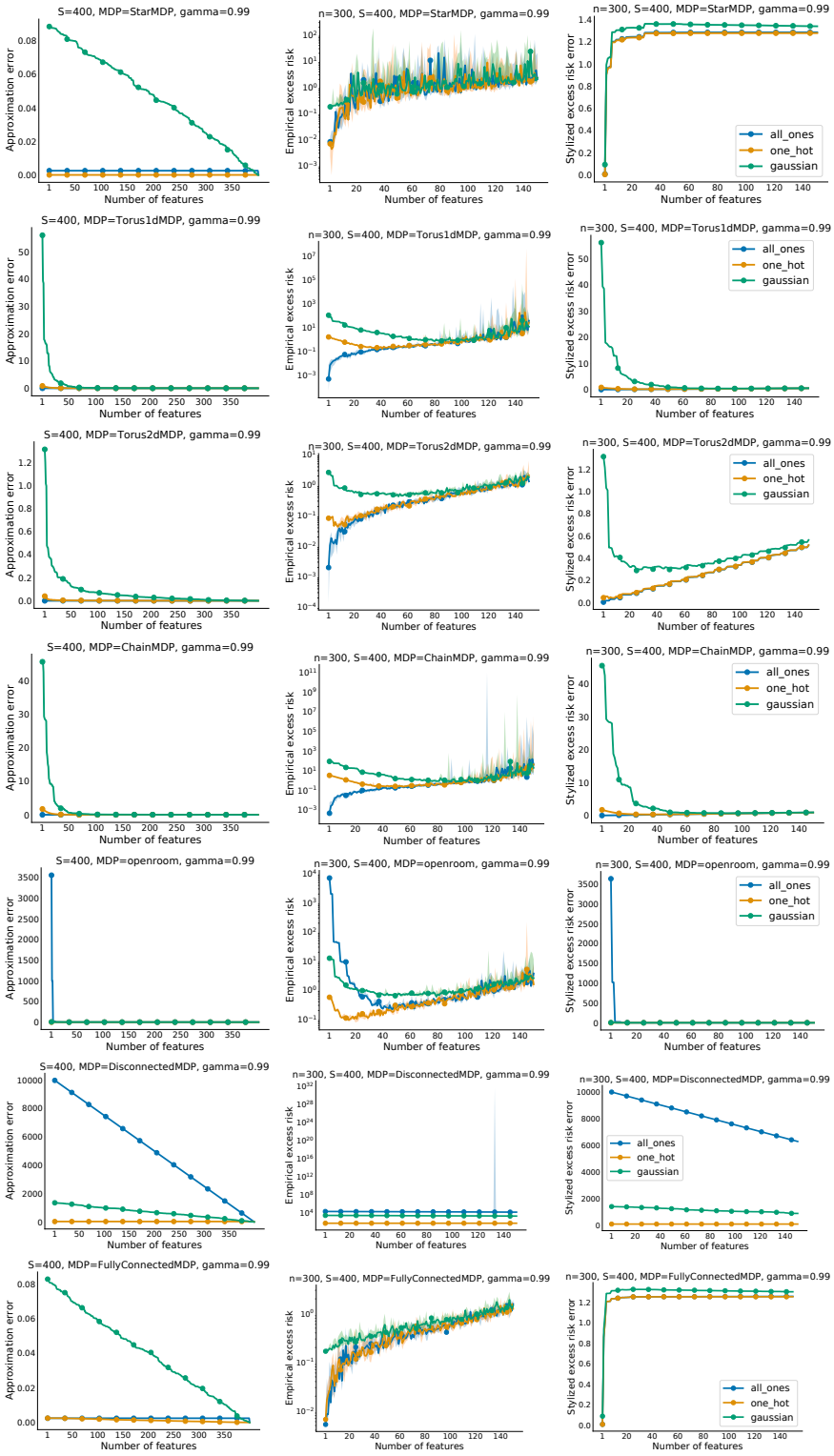

Figure 7: **Left:** Approximation error $\|P_{F_k} V^\pi\|$ given a one-hot, all-ones and Gaussian reward vector and for MDPs with different graphical structures. **Middle:** Median empirical excess risk $\mathcal{E}(V_{F_k,\hat{w}})$ given a one-hot, all-ones and Gaussian reward vector. **Right** Theoretical excess risk for a representation $\Phi_k = F_k$ and a one-hot, all-ones and Gaussian reward vector. The median is over 5 random seeds and shading gives 95% confidence intervals.

transition to the star center and the star center goes to the star corners. There are two extreme cases in terms of rewards: either the reward is the same for all s_i, $i \neq S$, (e.g. the all ones reward vector or the one-hot vector e_S) or not. If the reward is the same for all $s_i, i \neq S$, then this is effectively a 2 state structure, so we only really need 1 feature to distinguish between the value of the star corners and the value of the star center. However, if the reward is different for all $s_i (i \neq S)$ then we effectively have $(S-1)$ tuples (s_i, s_S) which can be thought of as independent graphical structures and we thus expect to need all the features to distinguish between their values. We can see this in Figure 7 that for the all ones reward vector and the one-hot reward vector e_S, the error with $k = 1$ is very good but for the Gaussian reward, the error with $k = 1$ is high.

Chain: This is a S-state connected graph with 2 pendant states and $(n-2)$ states of degree two. The shapes of the curves are similar to the Torus1d but we can notice that the errors are larger for each feature dimension k. This is intuitive as for instance in the case of an all ones reward vector, the values are not the same for each state due to the two end states of the chain, implying that more than one feature is needed to generalize the value function.

Openroom: This is a two-dimensional grid with S states. States strictly inside the grid have four neighbours. States belonging to one (reps. two) edges are of degree three (resp. two). As we observed in Figure 2, the Openroom domain does not generalizes as well as the Torus2d which can be explained by their difference in effective dimension.

Torus1d: This is a wrap-around version of the Chain. State i transitions to state $(i+1)$ mod S and state $(i-1)$ mod S. We can see that the curve showing the empirical excess risk (Middle) corresponding to the Gaussian reward vector has a sweet spot which is also predicted by our theory. Moreover, when all states have the same reward, their values are identical. Hence, in that case, only one feature is enough to have very low error which is shown both empirically and by our theoretical bound on Figure 7.

Torus2d: It is a wrap-around version of the Openroom domain such that each state has four different neighbors. We can see in Figure 2 that the Torus1d and Torus2d have similar effective dimension but the decay of the singular values is faster in the case of Torus2d translating into smaller approximation errors in Figure 7 (Middle). This results in overall lower excess risk for the Torus2d indicating it generalizes in general better than its one-dimensional counterpart. Just like for the Torus1d, in the case of the Gaussian reward vector, there is a non trivial optimal number of features k minimizing the excess risk, which we can notice is smaller than for the Torus1d.

Disconnected: This graph consists of S states that self-transition. We do not expect the successor representation to generalize well within this MDP as we cannot leverage knowledge from one feature state to another. This idea was already captured by the effective dimension shown in Figure 2. The plots in Figure 7 corroborates this both empirically and theoretically showing that its excess risk is indeed the highest across all transition structures considered.

Fullyconnected: This is a connected graph of S states where each state can transition to $(S-1)$ states. The first singular vector, which is the constant vector, is very good in terms of effective dimension but the second vector has high effective dimension. When the rewards are the same in each state, their values are identical. In that case, one feature is enough to distinguish between the S states leading to good generalization in that case. Additional features must be misleading as the excess risks rises significantly from a number of features $k = 2$.

C.2 Full Atari Results

For all experiments, we used the hyperparameters provided by Dopamine (Castro et al., 2018).

Compute. For our experiments on Atari, we used Tesla V100 GPUs and P100 for all runs. To obtain the pretrained deep representations for each deep RL agent, we ran a total of 5 runs / game $\times$ 60 games / algorithm $\times$ 5 algorithms = 1500 runs. Each of these runs takes around 5 days. Additionally, for the auxiliary loss experiment, we ran a total of 5 runs / game $\times$ 5 games / algorithm $\times$ 2 algorithms = 50 runs. In this setting, each run takes around 1 day. Overall, the amount of compute is of 7050 days of GPU training.

We provide a per-game comparison of the effective dimension of the representations induced by DQN, DQN (Adam), Rainbow, IQN and M-IQN throughout training in Figure 9 for all 60 Atari games in the online setting to complement the results presented in Figure 4 in the main part of the paper.

For the offline experiment presented in Figure 5, we use the same mini-batch sampled for the temporal-difference loss $\mathcal{L}_{\text{TD}}$ for computing the auxiliary loss $\mathcal{L}_\phi$. Our combined loss is then $\mathcal{L}_\alpha = (1-\alpha)\mathcal{L}_{\text{TD}} + \alpha\mathcal{L}_\phi$. We ran a hyperparameter sweep over α on the five games displayed in Figure 8 and found that a value of $\alpha = 0.1$ worked well. We provide per-game training curves for IQN agents for 17 Atari games in Figure 10 as well as the effective dimension (see Figure 11) of their induced representations computed with a batch size of 2^{15}. We also complement these results with the rank of these representations as a function of training in Figure 12 and Figure 13 as a proxy for the approximation error.

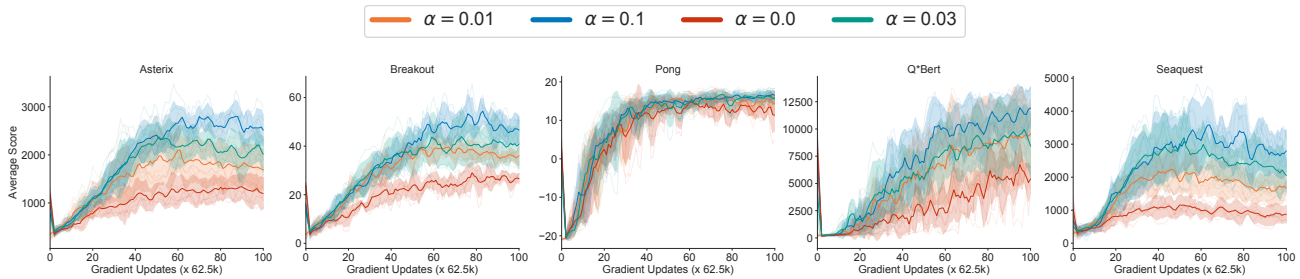

Figure 8: Sweeping over various values of α when adding the auxiliary loss $\mathcal{L}_\phi$ to IQN.

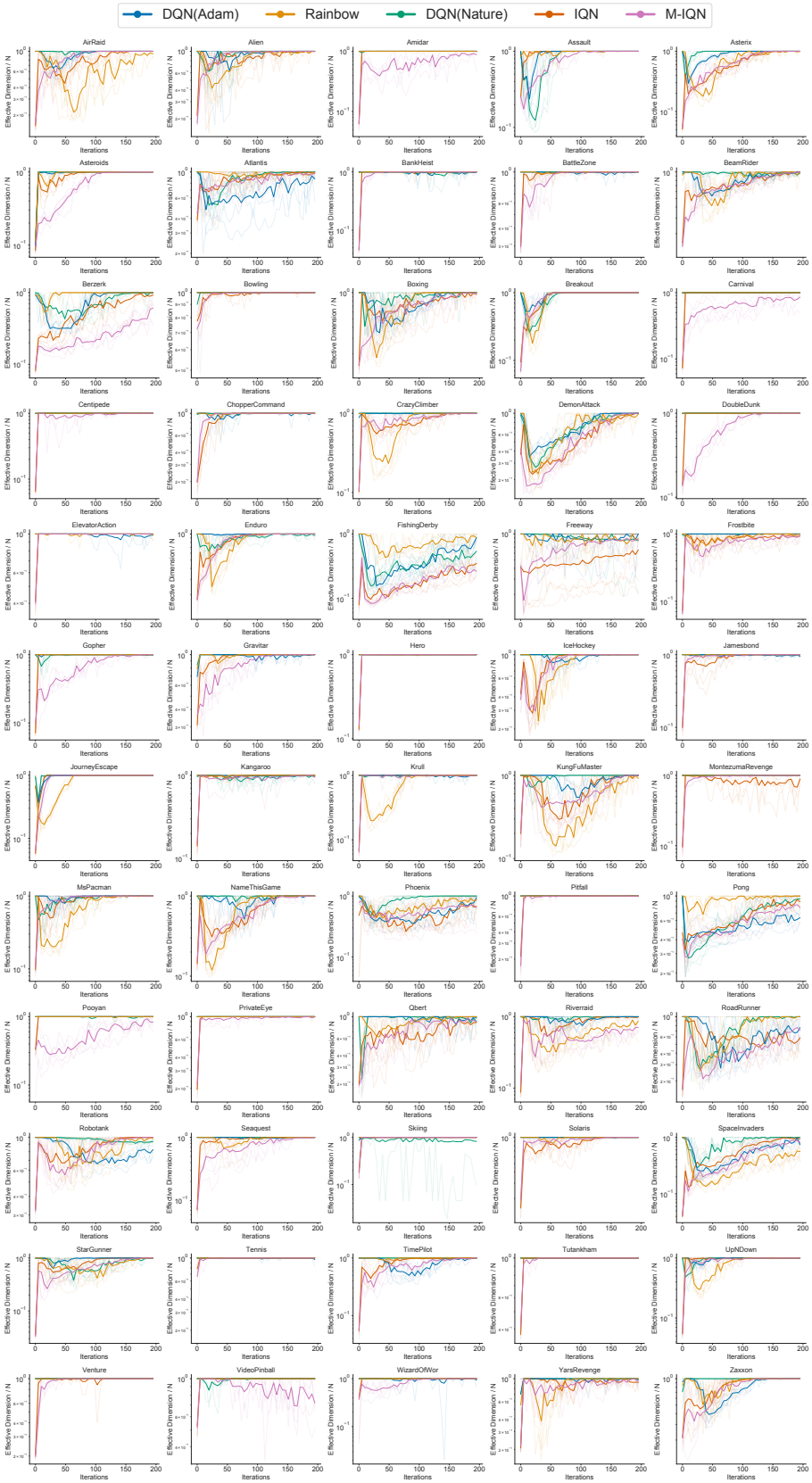

Figure 9: Average estimate (darker color) of the effective dimension normalized by the batch size used $N = 2^{15}$ on DQN(Nature), DQN(Adam), Rainbow, IQN and M-IQN on all 60 Atari games computed using 5 independent runs. Individual runs are shown with a lighter color.

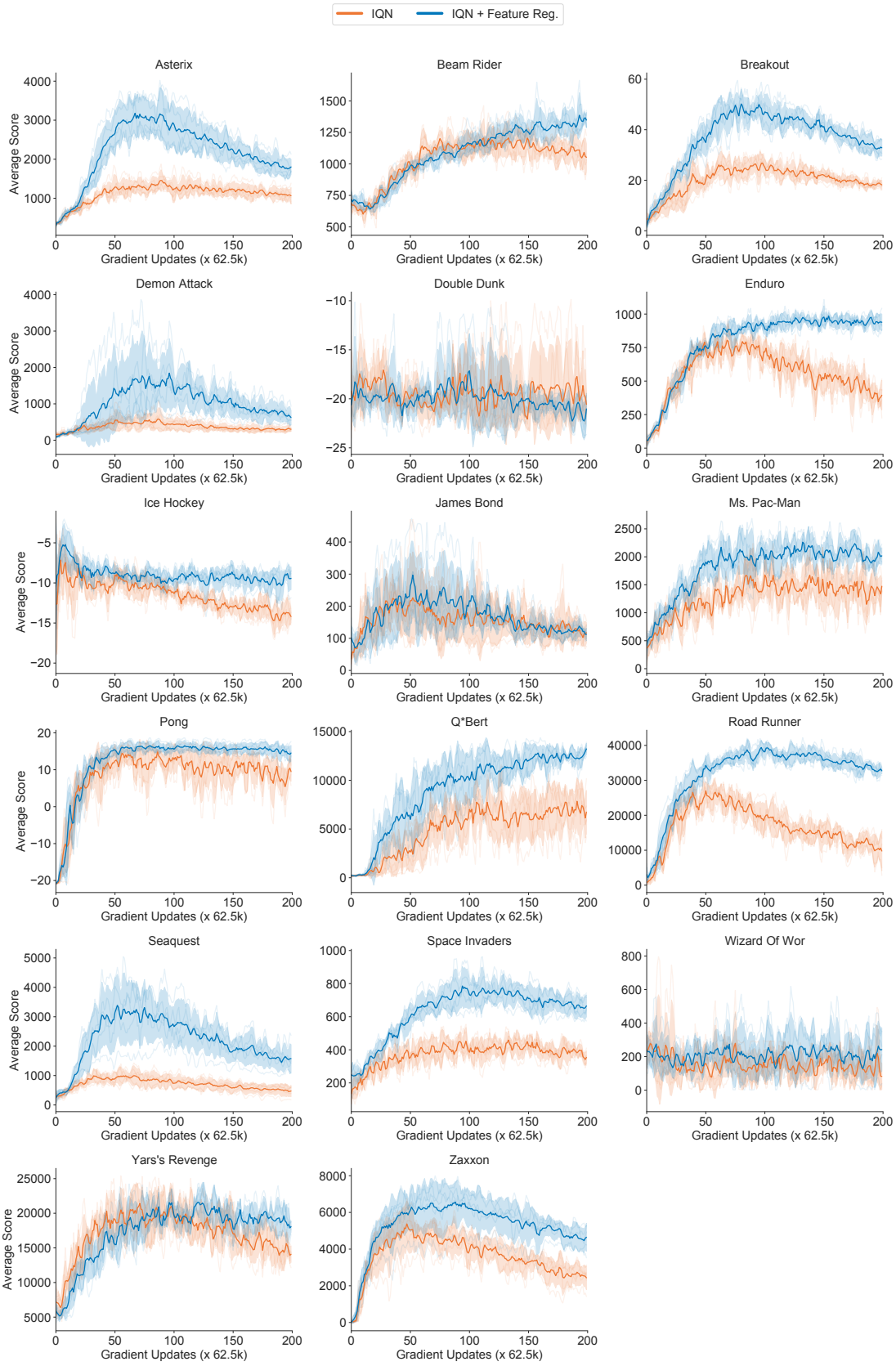

Figure 10: Per-game learning curves of IQN and IQN with feature regularization L_ϕ on 17 Atari games in the offline RL setting.

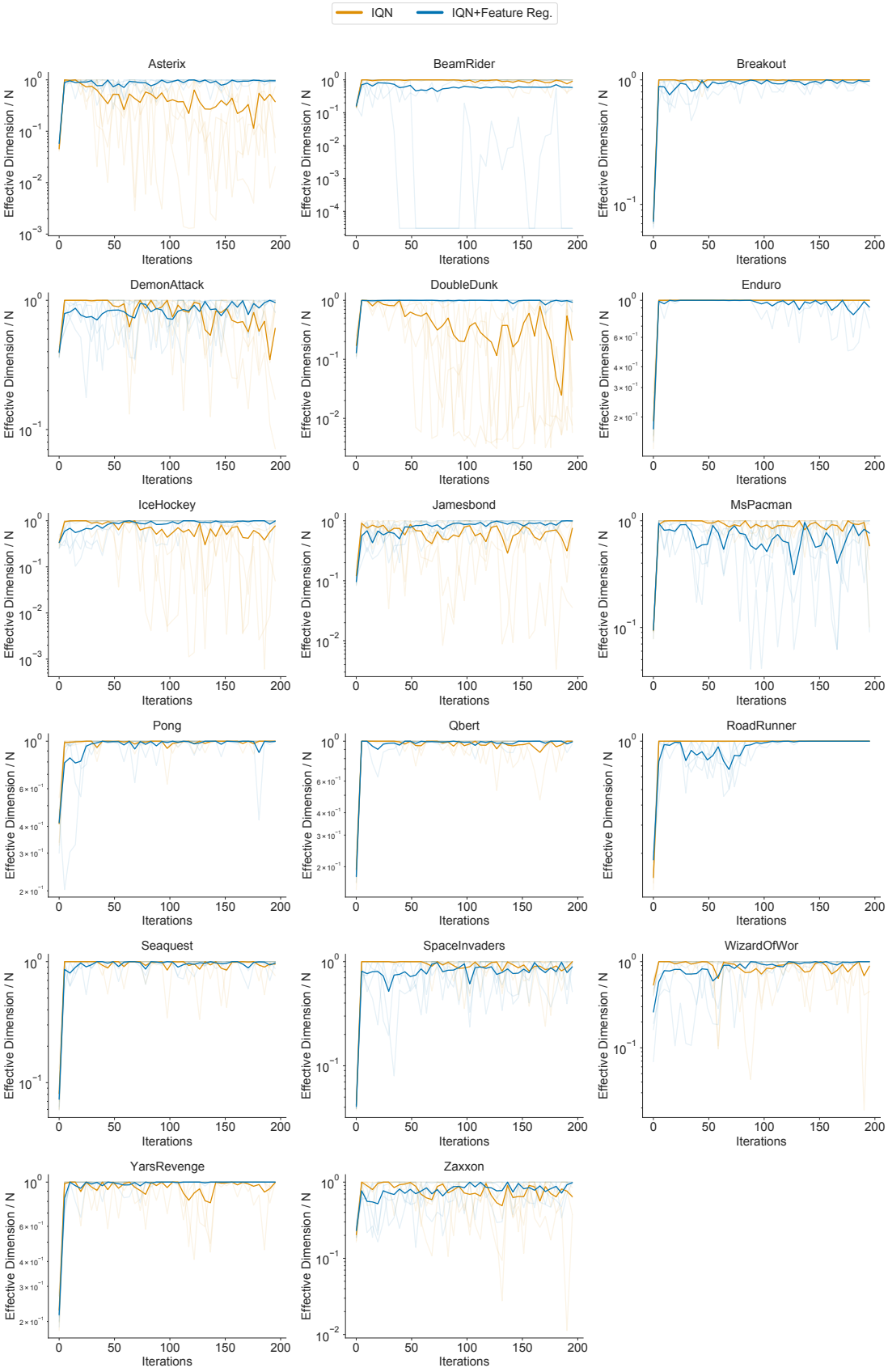

Figure 11: Per-game effective dimension normalized by the batch size $N = 2^{15}$ of IQN and IQN with feature regularization L_ϕ on 17 Atari games in the offline RL setting, using 5 independent runs. Individual runs are shown with a lighter color.

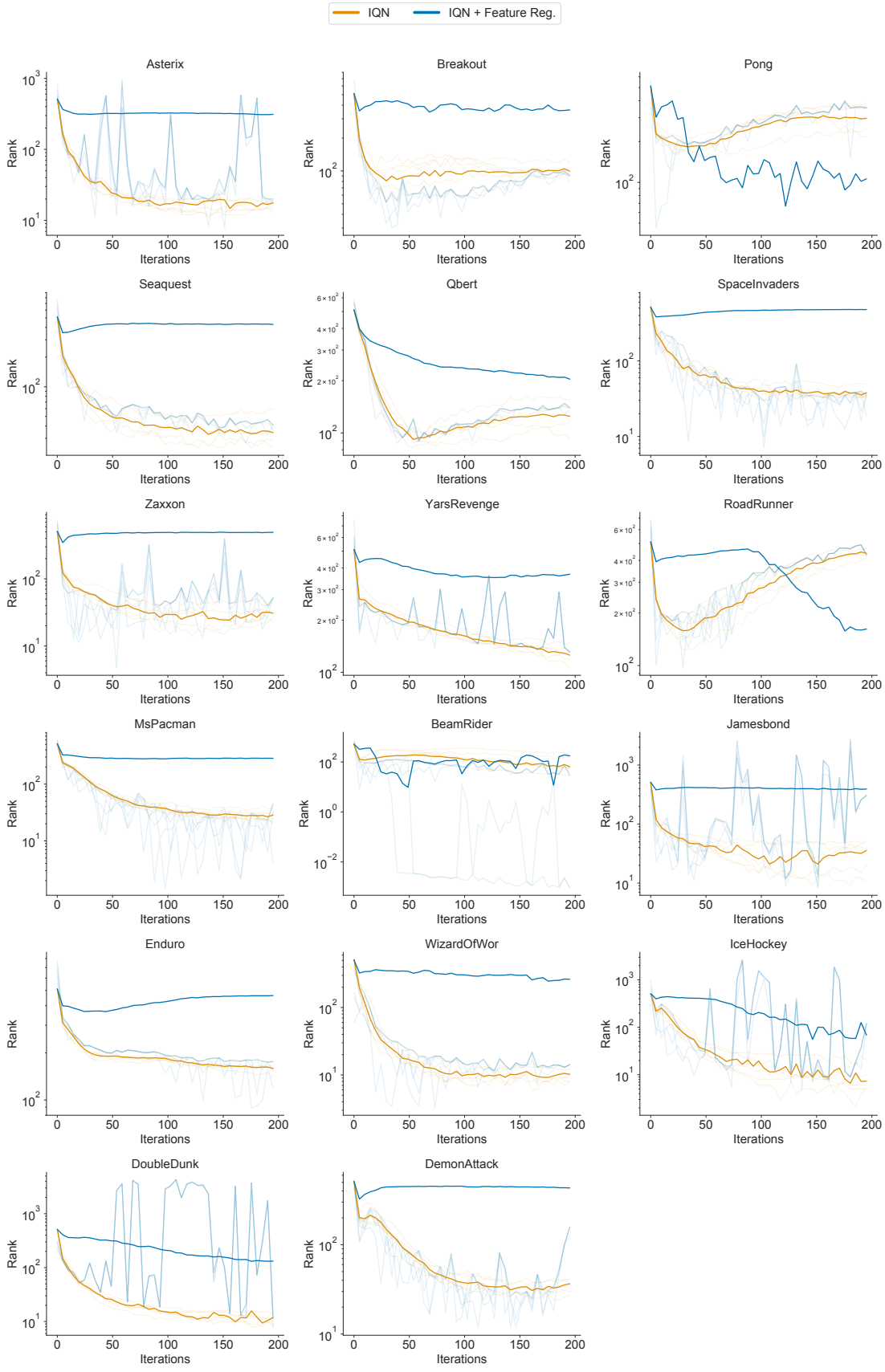

Figure 12: Per-game rank of IQN and IQN with feature regularization L_ϕ computed with a batch size $N = 2^{15}$ on 17 Atari games in the offline RL setting, using 5 independent runs. Individual runs are shown with a lighter color.

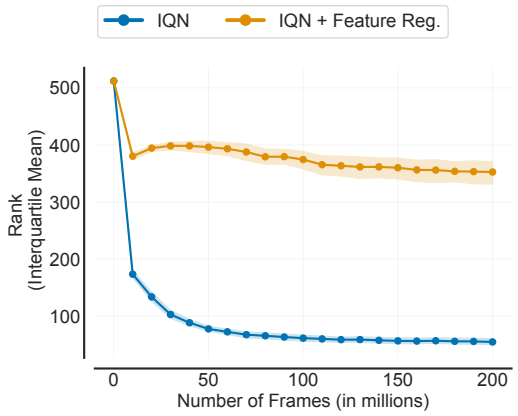

Figure 13: Interquartile mean (IQM) (Agarwal et al., 2021b) for the rank of representations induced by IQN and IQN with feature regularization L_ϕ computed with a batch size $N = 2^{15}$ on 17 Atari games in the offline setting.

D SOCIETAL IMPACT

This paper contributes to the fundamental understanding of state representations, characterizing their generalization capacity. Our work suggests that algorithms making use of representations minimized by the excess risk bound from Theorem 1 can improve their performance. However, when making the choice of such a representation, we did not focus on other important factors like the computational cost of learning these representations, their scalability or the biases these representations can propagate resulting into possible discriminatory outcomes or dangerous behaviours. We suggest that practitioners should not only consider our generalization characterization of representations but also ethical deliberations.