Understanding Deep Neural Function Approximation in Reinforcement Learning via ϵ-Greedy Exploration

Fanghui Liu,[*] Luca Viano, Volkan Cevher
Laboratory for Information and Inference Systems
École Polytechnique Fédérale de Lausanne (EPFL), Switzerland
{first}.{last}@epfl.ch

Abstract

This paper provides a theoretical study of deep neural function approximation in reinforcement learning (RL) with the ϵ-greedy exploration under the online setting. This problem setting is motivated by the successful deep Q-networks (DQN) framework that falls in this regime. In this work, we provide an initial attempt on theoretical understanding deep RL from the perspective of function class and neural networks architectures (e.g., width and depth) beyond the "linear" regime. To be specific, we focus on the value based algorithm with the ϵ-greedy exploration via deep (and two-layer) neural networks endowed by Besov (and Barron) function spaces, respectively, which aims at approximating an α-smooth Q-function in a d-dimensional feature space. We prove that, with T episodes, scaling the width $m = \widetilde{\mathcal{O}}(T^{\frac{d}{2\alpha+d}})$ and the depth $L = \mathcal{O}(\log T)$ of the neural network for deep RL is sufficient for learning with sublinear regret in Besov spaces. Moreover, for a two layer neural network endowed by the Barron space, scaling the width $\Omega(\sqrt{T})$ is sufficient. To achieve this, the key issue in our analysis is how to estimate the temporal difference error under deep neural function approximation as the ϵ-greedy exploration is not enough to ensure "optimism". Our analysis reformulates the temporal difference error in an $L^2(\mathrm{d}\mu)$-integrable space over a certain averaged measure μ, and transforms it to a generalization problem under the non-iid setting. This might have its own interest in RL theory for better understanding ϵ-greedy exploration in deep RL.

1 Introduction

Efficient reinforcement learning (RL) under the large (or even infinite) state space and action space setting is increasingly important and relevant challenge [1, 2, 3]. One of the first successful approaches towards this problem is the deep Q-network (DQN) [4, 5] framework, which deploys powerful nonlinear function approximation techniques via Deep Neural Networks (DNNs) [6] to concisely approximate state and action spaces. Despite its impressive practical success, there is still a gap between practical uses and theoretical understanding on deep RL with regard to the function class and the employed ϵ-greedy policy.

In the perspective of function class, many theoretical works center around linear function approximation [7, 8] and linear mixtures [9, 10]. Existing non-linear function approximation results on RL are largely based on neural tangent kernel (NTK) [11, 12], Bellman rank [13, 14], and Eluder dimension [15, 16, 17]. Nevertheless, these approaches fail in truly capturing the highly non-linear properties of deep RL. For example, NTK (or lazy training [18]) essentially works in a "linear" regime [19, 20, 21], and can not efficiently learn even a single ReLU neuron [22, 23, 24] as it requires $\Omega(\varepsilon^{-d})$ samples

[*] Correspondence to: Fanghui Liu <fanghui.liu@epfl.ch> and Luca Viano<luca.viano@epfl.ch>.

16th European Workshop on Reinforcement Learning (EWRL 2023).

to achieve ε approximation error, where d is the (original) or transformed feature dimension input; the Bellman rank is normally difficult to be estimated for neural networks as suggested by [25]; the Eluder dimension is at least in an exponential order [26, 27] even for two-layer neural networks. The above general function approximation schemes appear difficult to fully demonstrate the success of practical deep RL both theoretically and empirically.

In the perspective of exploration schemes, DQN is directly equipped with the ϵ-greedy policy instead of confidence-bound based scheme that are commonly used in RL theory. The ϵ-greedy exploration is theoretically demonstrated to have exponential sample complexity in the worst case [28] but is still popular in practical deep RL due to its simple implementation. In this case, theoretical analyses of ϵ-greedy in deep RL are still required. Besides, to ensure a sublinear regret, under the NTK regime, the width of neural networks is required to be $m = \Omega(T^{13})$ [12], where T is the number of episodes. This does not match deep RL in practice with small width/depth under large episodes [4, 29].

To bridge the large theory-practice gap, we study the value iteration algorithm with deep neural function approximation and the ϵ-greedy policy under the online setting, which broadly captures the key features of DQN. Our analysis framework is based on DNNs (as well as two-layer neural networks) where the target Q function lies in the Besov space [30] or the Barron space [31], respectively. These function classes can fully capture the properties of Q-functions, e.g., smoothness by neural networks. Our results demonstrate that the sublinear regret can be achieved for deep neural function approximation under the ϵ-greedy exploration with reasonably finite width and depth in practice. Besides, the relationship between the problem-dependent smoothness of Q-function and regret bounds is also developed. These results could also motivate practitioners to consider different architectures of implementations of deep RL.

1.1 Technical challenges and contributions

Most previous RL theory results on function approximation in the online setting work with "optimism in the face of uncertainty" principle for exploration, leading to a series of upper confidence bound (UCB)-type algorithms to ensure the temporal difference (TD) error smaller than zero.

Conceptually, optimism is sometimes too aggressive and UCB-style algorithms can suffer exponential sample complexity even for nonlinear bandits [27]. Technically, UCB-type algorithms in linear/kernel function approximation [7, 12, 32] depend on a known feature mapping or the NTK kernel, which appears invalid for deep neural function approximation beyond the "linear" regime. This is because, the used confidence ellipsoid and elliptical potential lemma are not applicable for data-dependent feature mapping of DNNs. To avoid explicitly designing a bonus function, Thompson sampling [33, 34] appears promising in a Bayesian perspective by using randomized (i.e., perturbed) versions of the estimated model or value function [35]. Nevertheless, the bonus function is still implicitly included in confidence estimate of perturbations.

In this work, we center around deep neural function approximation with the ϵ-greedy exploration. Since this exploration scheme is not enough to ensure the TD error smaller than zero, the technical challenge in our analysis is how to estimate it to ensure the sublinear regret. In our proof framework, by a measure transform, the TD error is analysed in an $L^2(\mathrm{d}\bar{\mu})$-integrable space, where $\bar{\mu}$ is the averaged measure wrt a mini-batch of historical state-action pairs. To break the dependence between the episodes for neural networks training, we utilize the *experience replay* scheme [36] from DQN, and then transform the TD error estimation to generalization error under the independent but non-identically distributed data setting and approximation error in the respective function spaces. Note that in practice, experience replay makes observations to be (nearly) iid, but our analysis only requires the independence of observations, that is weaker than iid. Such generalization problem can be addressed by uniform convergence via (local) Rademacher complexity of the Besov/Barron spaces under the averaged measure. This considered function spaces in this work is more general than Hölder spaces used in offline RL [37].

Our results show that (*i*) the problem-dependent smoothness of Q-function affects the efficiency of learning with deep RL, which can be improved by increasing the model capacity (width and depth). We use α as a parameter indicating the smoothness degree of Q-function. A larger α indicates smoother functions, easier RL tasks, and smaller exploration times, which coincides with our theory. (*ii*) for deep neural networks under the Besov space, the width $m = \widetilde{\mathcal{O}}(T^{\frac{d}{2\alpha+d}})$ and the depth $L = \widetilde{\mathcal{O}}(1)$ are enough for sublinear regret under the ϵ-greedy policy, where $\widetilde{\mathcal{O}}(\cdot)$ omits the $\log$

terms. (*iii*) for two-layer neural networks under the Barron space, the width $m = \Omega(\sqrt{T})$ suffices to ensure sublinear regret. Furthermore, our regret bounds can be independent of the feature dimension, supporting the premise of practical, high-dimensional data in RL.

1.2 Related work

Recent work on neural network function approximation beyond NTK (or the Eluder dimension) mainly restrict on the generative setting [25, 38] by assuming a simulator in which the agent can require any state and action, and the offline setting [37, 39]. In sequel, we review RL with function approximation under the online setting that DQN falls into this regime. We also mention that, theoretical understanding of DQN can be conducted by from the perspective of neural fitted Q-iteration algorithm [40, 37, 41], and Q learning [42] in the perspective of understanding the target network [43] and experience replay [44, 45, 46] with linear function approximation. Note that, for notational consistency with previous work, in this subsection, T denotes the total number of steps (i.e., interactions with the environment) instead of the number of episodes in our paper.

RL with linear/kernel function approximation: RL with linear function approximation achieves a sublinear regret bound with $\widetilde{\mathcal{O}}(\sqrt{d^3H^3T})$ under a low-rank MDP in a model-free setting [7] and $\widetilde{\mathcal{O}}(dH^2\sqrt{T})$ in a model-based setting [32], where H is the length of each episode. The regret can be improved to $\widetilde{\mathcal{O}}(dH\sqrt{T})$ under a low inherent Bellman error by assuming a global planning oracle [47] or under a Bernstein-type exploration bonus and controlling extra uniform convergence cost [48]. This nearly optimal regret can be also achieved under the linear mixtures setting [10]. In the kernel regime, the regret can be achieved with $\widetilde{\mathcal{O}}(\delta_{\mathcal{F}}\sqrt{H^3T})$ [32, 12], where $\delta_{\mathcal{F}}$ is the intrinsic complexity (e.g., effective dimension) of the function class RKHS $\mathcal{F}$. The above bounds are based on confidence ellipsoid to quantify the uncertainty in an explicit bonus function by feature mapping/kernel function; while Thompson sampling [34, 33] utilizes an implicit bonus function in probability estimation on uncertainty quantification, which leads to an $\widetilde{\mathcal{O}}(d^2H^2\sqrt{T})$ [35] regret in linear function approximation.

RL with general function approximation: One prototypical scheme uses the Eluder dimension [15], which measures the degree of dependence among action rewards, resulting in an $\widetilde{\mathcal{O}}(\texttt{poly}(\delta_{\mathcal{F}}H)\sqrt{T})$ regret [16, 17], where the complexity $\delta_{\mathcal{F}}$ depends on the Eluder dimension. Using this metric, the sublinear regret under the ϵ-greedy exploration can be achieved by [49]. Besides, the low Bellman rank assumption [13], where the Bellman error "matrix" admits a low-rank factorization, can be also used general function approximation [14] by measuring the error of the function class under the Bellman operator. Combining Bellman rank and Eluder dimension results in a new metric, Bellman Eluder dimension [50], achieving $\widetilde{\mathcal{O}}(H\sqrt{\delta_{\mathcal{F}}T})$-regret, where $\delta_{\mathcal{F}}$ depends on this metric.

Overall, the above metrics are difficult to the nonlinear spaces of DNNs beyond "linear" regime that concern us.

2 Background and preliminaries

In this section, we introduce the necessary background and definitions with respect to online reinforcement learning based on episodic Markov decision processes (MDPs) and function spaces of deep (and two-layer) ReLU neural networks.

Notation: We denote by $a(n) \lesssim b(n)$: there exists a positive constant c independent of n such that $a(n) \leqslant cb(n)$; $a(n) \asymp b(n)$: there exists two positive constant c_1 and c_2 independent of n such that $c_1 b(n) \leqslant a(n) \leqslant c_2 b(n)$. We use the shorthand $[n] := \{1, 2, \dots, n\}$ for some positive n and $\lceil x \rceil$ denotes the smallest integer exceeding x. Let $\mathcal{X} = [0,1]^d$ be a domain of the functions, we denote the L^p-integrable space by $L^p(\mathcal{X})$ endowed by the norm $\|f\|_{L^p(\mathcal{X})} = \big(\int_{\mathcal{X}} |f(\boldsymbol{x})|^p \mathrm{d}\boldsymbol{x}\big)^{1/p}$, and the μ-integrable L^p space by $L^p(\mathrm{d}\mu)$ for a probability measure μ on $\mathcal{X}$ and the norm is given by $\|f\|_{L^p(\mathrm{d}\mu)} = \big(\int_{\mathcal{X}} |f(\boldsymbol{x})|^p \mathrm{d}\mu\big)^{1/p}$.

2.1 Episodic Markov decision processes

A (finite-horizon) episodic MDPs is denoted as MDP$(\mathcal{S}, \mathcal{A}, H, \mathbb{P}, r)$, where $\mathcal{S}$ is the state space with possibly infinite states; $\mathcal{A}$ is the finite action space; H is the number of steps in each episode;

$\mathbb{P} := \{\mathbb{P}_h\}_{h=1}^H$ is the Markov transition kernel with the transition probability $\mathbb{P}_h(\cdot|s,a)$ on action a taken at state $s \in \mathcal{S}$ in the h-th step; the reward functions $r := \{r_h\}_{h=1}^H$ are assumed to be deterministic. For notational simplicity, denote $\mathcal{X} = \mathcal{S} \times \mathcal{A}$ and $\boldsymbol{x} = (s,a)$, we assume $\mathcal{X} = [0,1]^d$ as a compact space of $\mathbb{R}^d$ and $r_h : \mathcal{S} \times \mathcal{A} \to [0,1]$ at h-th step.

A non-stationary policy π is a collection of H functions $\pi := \{\pi_h : \mathcal{S} \to \mathcal{A}\}_{h=1}^H$. Given a policy π, the (state) value function $V_h^\pi \colon \mathcal{S} \to [0,H]$ is defined as the expected cumulative reward of the MDP starting from step $h \in [H]$, i.e., $V_h^\pi(s) = \mathbb{E}_\pi\big[\sum_{h'=h}^H r_{h'}(s_{h'}, a_{h'})\big|s_h = s\big], \forall s \in \mathcal{S}, h \in [H]$ where $\mathbb{E}_\pi[\cdot]$ denotes the expectation with respect to the randomness of the trajectory $\{(s_h, a_h)\}_{h=1}^H$ obtained by the policy π. Likewise, the action-value function $Q_h^\pi : \mathcal{S} \times \mathcal{A} \to [0,H]$ is defined as $Q_h^\pi(s,a) = \mathbb{E}_\pi\big[\sum_{h'=h}^H r_{h'}(s_{h'}, a_{h'}) \,\big|\, s_h = s,\, a_h = a\big]$.

Moreover, since the action space and episode length are both finite, there always exists an optimal policy $\pi^\star$ [51] such that $V_h^\star(s) = \sup_\pi V_h^\pi(s)$ for all $s \in \mathcal{S}$ and $h \in [H]$. To simplify the notation, denote $(\mathbb{P}_h V)(s,a) := \mathbb{E}_{s' \sim \mathbb{P}_h(\cdot|s,a)}[V(s')]$ and the Bellman operator $(\mathbb{T}_h V)(s,a) = r_h(s,a) + (\mathbb{P}_h V)(s,a)$ for any measurable function $V \colon \mathcal{S} \to [0,H]$. Using this notation, the Bellman equation associated with a policy π can be formulated as

$$Q_h^\pi(s,a) = (\mathbb{T}_h V_{h+1}^\pi)(s,a), \qquad V_h^\pi(s) = \langle Q_h^\pi(s,\cdot), \pi_h(\cdot|s)\rangle_{\mathcal{A}}, \qquad V_{H+1}^\pi(s) = 0\,. \tag{1}$$

Similarly, the Bellman optimality equation is given by

$$Q_h^\star(s,a) = (\mathbb{T}_h V_{h+1}^\star)(s,a), \qquad V_h^\star(s) = \max_{a \in \mathcal{A}} Q_h^\star(s,a), \qquad V_{H+1}^\star(s) = 0\,. \tag{2}$$

Accordingly, the optimal policy $\pi^\star$ is the greedy policy with respect to $\{Q_h^\star\}_{h=1}^H$. Hence the Bellman optimality operator $\mathbb{T}_h^\star$ is defined as

$$(\mathbb{T}_h^\star Q)(s_h, a_h) = r_h(s_h, a_h) + \mathbb{E}_{s_{h+1} \sim \mathbb{P}_h(\cdot|s_h, a_h)}[\max_{a \in \mathcal{A}} Q(s_{h+1}, a)], \qquad \forall\, Q : \mathcal{S} \times \mathcal{A} \to [0,H]\,.$$

By definition, the Bellman equation in Eq. (2) is equivalent to $Q_h^\star = \mathbb{T}_h^\star Q_{h+1}^\star$, $\forall h \in [H]$.

In the **online setting**, the goal is to learn the optimal policy $\pi^\star$ by minimizing the cumulative regret under the interaction with the environment over a number of episodes. For any policy π, the difference between V_1^π and $V_1^\star$ quantifies its sub-optimality. Thus, after T (fixed but large) episodes, the total (expected) regret is defined as $\mathrm{Regret}(T) = \sum_{t=1}^T \big[V_1^\star(s_1^t) - V_1^{\tilde{\pi}^t}(s_1^t)\big]$, where $\tilde{\pi}^t$ is the policy executed in the t-th episode and s_1^t is the initial state.

2.2 Function spaces

We give an overview of Besov spaces for deep neural networks and the Barron space for two-layer neural networks. More details refer to Appendix A. For description simplicity, we focus on the ReLU activation function in this work.

Besov spaces: Previous work in approximation theory focuses on the "smoothness" of the function, e.g., Hölder spaces [52, 37] and Sobolev spaces [53]. Here we consider the concept of α-smooth from modulus of smoothness [30], *cf.*, Appendix A.

Based on this, we consider a more general function space beyond Hölder spaces and Sobolev spaces, i.e., Besov spaces [54, 30], which allows for spatially inhomogeneous smoothness with spikes and jumps. The Besov space is defined by $\mathcal{B}_{p,q}^\alpha(\mathcal{X}) = \{f \in L^p(\mathcal{X}) \mid \|f\|_{\mathcal{B}_{p,q}^\alpha} < \infty\}$, where the Besov norm is $\|f\|_{\mathcal{B}_{p,q}^\alpha} := \|f\|_{L^p(\mathcal{X})} + |f|_{\mathcal{B}_{p,q}^\alpha}$. The smoothness parameter α indicates which function at a certain smoothness degree can be represented. For example, if $\alpha > d/p$, then the related Besov space is continuously embedded in the set of the continuous functions; if $\alpha < d/p$, then the functions in the Besov space are no longer continuous. The formal definition and relations to Hölder spaces and Sobolev spaces are deferred to Appendix A.

Barron spaces: A two-layer neural network with m neurons can be represented as $f(\boldsymbol{x}) = \frac{1}{m}\sum_{k=1}^m b_k \sigma(\boldsymbol{w}_k^\top \boldsymbol{x} + c_k)$ with the ReLU activation function $\sigma(\cdot)$ used in this work and the neural network parameters $\{b_k, \boldsymbol{w}_k, c_k\}_{k=1}^m$. It admits the integral representation $f(\boldsymbol{x}) = \int_\Omega b\sigma\left(\boldsymbol{w}^\top \boldsymbol{x} + c\right) \rho(\mathrm{d}b, \mathrm{d}\boldsymbol{w}, \mathrm{d}c)$, $\boldsymbol{x} \in \mathcal{X}$, where $\Omega = \mathbb{R} \times \mathbb{R}^d \times \mathbb{R}$ and ρ is a probability measure over Ω. Then the Barron space [31] endowed by the Barron norm is defined as

$$\widetilde{\mathcal{P}} = \left\{ f \ \text{ admits Eq. (9)} : \|f\|_{\widetilde{\mathcal{P}}} = \inf_\rho \{\mathbb{E}_\rho |b| (\|\boldsymbol{w}\|_1 + |c|)\} < \infty \right\}.$$

The Barron space $\widetilde{\mathcal{P}}$ [31] can be (roughly) equipped with the ℓ_1-path norm, i.e., $\|f\|_{\widetilde{\mathcal{P}}} \leqslant \|f\|_{\mathcal{P}} := \frac{1}{m}\sum_{k=1}^{m} |b_k|(\|\boldsymbol{w}_k\|_1 + c_k) \leqslant 2\|f\|_{\widetilde{\mathcal{P}}}$. Accordingly, it is natural to use $\|f\|_{\mathcal{P}}$ to denote the Barron norm, as the discrete version.

The Barron space [31] can be regarded as the *largest* function space for two-layer ReLU neural networks. Here the "largest" terminology [31, 55] means that the approximation ability can avoid *curse of dimensionality*, i.e., 1) any function in Barron spaces can be efficiently approximated by two-layer neural networks with bounded norm; 2) any continuous function that can be efficiently approximated by two-layer neural networks with bounded norm belongs to a Barron space.

We remark that, avoiding curse of dimensionality is important in theory for practical high-dimensional data in RL. However, Besov spaces are too large and thus do not enjoy this property for deep ReLU neural networks.

3 Algorithm: Value iteration via DNNs under ϵ-greedy exploration

In this section, we lay out our algorithm 1 via value iteration by DNNs under the ϵ-greedy policy. Though our value iteration algorithm is different from one gradient-step for deep Q-learning in DQN, it still shares the key spirit with DQN in terms of function approximation via DNNs, ϵ-greedy exploration, and experience replay.

Function class: We define the function class $\mathcal{F}$ given by $\mathcal{F} = \mathcal{F}_1 \times \cdots \times \mathcal{F}_H$, including $\mathcal{F}_{\texttt{SNN}}$ for two-layer (Shallow) ReLU neural networks and $\mathcal{F}_{\texttt{DNN}}$ for deep ReLU neural networks as below

$$\mathcal{F}_{\texttt{SNN}} = \left\{ f : [0,1]^d \to [0,H] \middle| f(\boldsymbol{x}) = \frac{1}{m}\sum_{k=1}^{m} b_k \sigma(\boldsymbol{w}_k^{\top}\boldsymbol{x} + c_k), \|f\|_{\mathcal{P}} \leqslant B \right\}, \tag{3}$$

where $B > 0$ is the ℓ_1-path norm constraint parameter, and deep ReLU neural networks [30] as

$$\begin{aligned} \mathcal{F}_{\texttt{DNN}}(L, m, S, B) := \Bigg\{ f : [0,1]^d \to [0,H] \Bigg| f(\boldsymbol{x}) = (\boldsymbol{W}^{(L)}\sigma(\cdot) + b^{(L)}) \circ \cdots \circ (\boldsymbol{W}^{(1)}\boldsymbol{x} + \boldsymbol{b}^{(1)}), \\ \sum_{i=1}^{L}(\|\boldsymbol{W}^{(i)}\|_0 + \|\boldsymbol{b}^{(i)}\|_0) \leqslant S, \ \max_i(\|\boldsymbol{W}^{(i)}\|_\infty \vee \|\boldsymbol{b}^{(i)}\|_\infty) \leqslant B \Bigg\}, \end{aligned} \tag{4}$$

where the weight parameters are $\boldsymbol{W}^{(1)} \in \mathbb{R}^{m \times d}$, $\boldsymbol{W}^{(l)} \in \mathbb{R}^{m \times m}$, $\forall l \in \{2, 3, \dots, L-1\}$, and $\boldsymbol{W}^{(L)} \in \mathbb{R}^{m}$; the bias parameter are $\boldsymbol{b}^{(l)} \in \mathbb{R}^{m}$, $\forall l \in [L-1]$ and $b^{(L)} \in \mathbb{R}$. Such sparsely-connected neural networks require most of the network parameters to be zero or non-active, which can be verified [56]. The depth L, the width m, the sparsity parameter S and the norm parameter B can be determined later in our proof to achieve good approximation and estimation performance.

Experience replay: In our setting, after initialization, at t-th episode, at h-th time step, we have observed $t-1$ transition tuples, $\{(s_h^\tau, a_h^\tau, s_{h+1}^\tau)\}_{\tau=1}^{t-1}$ and attempt to estimate $\{Q_h^\star\}_{h=1}^{H}$ via DNNs. Note that, at each time step h, these $t-1$ transition tuples are neither independent nor identically distributed due to the interaction with value functions and stochastic transition. To pursue the independence among the transition tuples that is required in our analysis, we follow the *experience replay* scheme [36] that is successfully applied in DQN [4]. The intuition behind experience replay is to break (or weaken) the temporal dependency among the observations for neural networks training. When the replay memory is large (e.g., 10^6 in DQN [4]), experience replay is close to sampling independent transitions. To be specific, at t-th episode, we store transition $\{(s_h^t, a_h^t, r_h, s_{h+1}^t)\}_{h=1}^{H}$ in the replay memory $\mathcal{D}$, and then sample a mini-batch of *independent* observations from $\mathcal{D}$ with $\{(s_h^{\tau_j}, a_h^{\tau_j}, s_{h+1}^{\tau_j})\}_{(j,h)\in[\tilde{t}]\times[H]}$ for DNNs training. Here the number of mini-batch is denoted as $\tilde{t} := \lceil \varrho t \rceil$ with the mini-batch ratio $\varrho \in (0,1)$, and $\{\tau_j\}_{j=1}^{\tilde{t}}$ is the index for the mino-batch of $\tilde{t}$ independent samples. Note that such independence assumption from experience replay is also used in RL theory, e.g., [37, 44] and theoretically demonstrated to be a good de-correlator [57]. In fact, our analysis only requires independence via experience replay, which is still weaker than the standard iid assumption.

Value iteration via neural networks: In our algorithm, we apply the classical least squares value iteration via neural networks for value function learning [28]. We solve the following least squares

Algorithm 1 Value Iteration via DNNs under ϵ-greedy exploration with experience replay

```
Input: Function class $\mathcal{F}$, the number of episodes $T$, the $\epsilon$-greedy parameter $\epsilon \in (0,1)$, mini-batch ratio $\varrho \in (0,1)$.
Initialize replay memory $\mathcal{D}$.
for episode $t = 1, \ldots, T$ do
    Receive the initial state $s_1^t$.
    Set $V_{H+1}^t$ as the zero function.
    Set the minibatch size $\tilde{t} := \lceil \varrho t \rceil$ for experience replay.
    for step $h = H, \ldots, 1$ do
        Obtain $\widehat{Q}_h^t := \operatorname{argmin}_{f \in \mathcal{F}} \sum_{j=1}^{\tilde{t}} \left[ f(s_h^{\tau_j}, a_h^{\tau_j}) - r_h(s_h^{\tau_j}, a_h^{\tau_j}) - V_{h+1}^t(s_{h+1}^{\tau_j}) \right]^2$.
        Obtain $Q_h^t := \widehat{Q}_h^t$ and $V_h^t(\cdot) = \max_{a \in \mathcal{A}} Q_h^t(\cdot, a)$.
    end for
    //$\epsilon$-greedy for exploration
    Take the policy $\{\tilde{\pi}_h^t\}_{h=1}^H$ to be greedy policy with probability $1 - \epsilon$ or any policy with probability $\epsilon$.
    for step $h = 1, \ldots, H$ do
        Take $a_h^t \sim \tilde{\pi}_h^t(\cdot|s_h^t)$ .
        Observe the reward $r_h(s_h^t, a_h^t)$ and obtain the next state $s_{h+1}^t$.
    end for
    //experience replay
    Store transition $\{(s_h^t, a_h^t, r_h, s_{h+1}^t)\}_{h=1}^H$ in $\mathcal{D}$.
    Sample random mini-batch of transitions from $\mathcal{D}$ with $\tilde{t}$ pairs $\{(s_h^{\tau_j}, a_h^{\tau_j}, s_{h+1}^{\tau_j})\}_{(j,h) \in [\tilde{t}] \times [H]}$.
end for
```

regression problem via $\tilde{t}$ independent samples

$$\widehat{Q}_h^t = \operatorname*{argmin}_{f \in \mathcal{F}} \widehat{\mathcal{E}}_h^t(f) := \frac{1}{\tilde{t}} \sum_{j=1}^{\tilde{t}} \left[f(s_h^{\tau_j}, a_h^{\tau_j}) - r_h(s_h^{\tau_j}, a_h^{\tau_j}) - V_{h+1}^t(s_{h+1}^{\tau_j}) \right]^2 . \tag{5}$$

For ease of simplicity for analyses, we directly assume that the global minima solution of problem (5) can be obtained, that follows [58, 52, 59] in deep learning theory. Nevertheless, our result could be extended to allow small optimization error in each episode that will be discussed in Appendix B.

Besides, we also need the expectation version of $\widehat{\mathcal{E}}_h^t$ in problem (5) for our analysis. Formally, we assume each state-action pair in the mini-batch is sampled from a respective (unknown) probability measure, i.e., $(s_h^{\tau_j}, a_h^{\tau_j}) \sim \mu_h^{\tau_j}, \forall j \in [\tilde{t}]$, where $\mu_h^{\tau_j} \in \mathscr{P}(\mathcal{S} \times \mathcal{A})$ is from the collection of all probability distribution on $\mathcal{S} \times \mathcal{A}$. Taking the averaged measure $\bar{\mu}_h^{\tilde{t}} := \frac{1}{\tilde{t}} \sum_{j=1}^{\tilde{t}} \mu_h^{\tau_j}$, the expectation of $\widehat{\mathcal{E}}_h^t$ is defined as

$$\mathcal{E}_h^t(f) = \mathbb{E}_{(s_h, a_h) \sim \bar{\mu}_h^{\tilde{t}}, s_{h+1} \sim \mathbb{P}_h(\cdot|s_h, a_h)} \left[f(s_h, a_h) - r_h(s_h, a_h) - V_{h+1}^t(s_{h+1}) \right]^2 . \tag{6}$$

Note that, $\widehat{Q}_h^t$ in Eq. (5) is not an unbiased estimator of the squared Bellman error minimizer [60, 61]. Indeed, $\mathcal{E}_h^t$ differs from the squared Bellman error because of an extra variance term caused by the stochastic transition [62]. This biased estimation issue can be avoided (or alleviated) in practice by introducing target networks in DQN [63]. Some variants [64] of DQN can also reduce the biased estimate and performs well without target networks. Nevertheless, in our analysis, we center around the uniform bound $\sup_{f \in \mathcal{F}} |\mathcal{E}_h^t(f) - \widehat{\mathcal{E}}_h^t(f)|$ instead of the Bellman error.

ϵ-greedy exploration: In order to work in the online setting, we need to ensure that the learner visits "good" state action pairs in the sense that are almost maximizers of the value function for unseen state, *a.k.a.*, exploration. In RL theory, a classical way is to design an optimistic estimate of the value function via a bonus function b_h^t [12, 65] such that $Q_h^t = \min\{\widehat{Q}_h^t + b_h^t, H\}^+$. Instead, we employ the ϵ-greedy exploration that follows DQN-like algorithms. Using the ϵ-greedy exploration will ensure each state-action pair can be visited with positive probability and favor independence among samples. In our algorithm, we directly set $Q_h^t := \min\{\widehat{Q}_h^t, H\}^+$, and then naturally incorporate the truncation operation in neural networks training, see Eqs. (3) and (4).

Based on the above description, our algorithm centers around deep neural function approximation via value iteration under the ϵ-greedy exploration and experience replay under the online setting. This problem setting matches the spirit of practical DQN, which allows for better understanding deep RL.

4 Main results

This section presents our results for value iteration under deep (as well as two-layer) ReLU neural networks via the Besov spaces and Barron spaces, respectively. Our theory is based on the independence assumption via experience replay and achieves sublinear regret under the ϵ-greedy exploration.

4.1 Efficient value iteration via DNNs in Besov spaces

In this setting, we consider $\widehat{Q}_h^t = \operatorname{argmin}_{f \in \mathcal{F}_{\mathtt{DNN}}} \widehat{\mathcal{E}}_h^t(f)$ in Eq. (5), where $\mathcal{F}_{\mathtt{DNN}}$ is the function space of deep ReLU neural networks defined in Eq. (4). We make the following assumption on the Besov space $\mathcal{B}$, similar to [7, 12], where the Bellman optimality operator maps any bounded value function to a bounded Besov space ball.

Assumption 1. *Let $\widetilde{R}$ be a fixed constant. Define $\mathcal{B}_{\widetilde{R}} = \{f \in \mathcal{B}^{\alpha}_{p,q}(\mathcal{X}) : \|f\|_{\mathcal{B}} \leqslant \widetilde{R}\}$ in the Besov space and assume that for any $h \in [H]$ and $Q \colon \mathcal{S} \times \mathcal{A} \to [0, H]$, we have $\mathbb{T}_h^{\star} Q \in \mathcal{B}_{\widetilde{R}}$.*

Remark: Due to $Q \in [0, H]$, the radius $\widetilde{R}$ in fact depends on H, i.e., $\widetilde{R} \asymp H$.

Based on this assumption, we have the following theorem on the regret bound in the Besov space for deep RL under the ϵ-greedy exploration.

Theorem 1. *Under Assumption 1 with the smoothness parameter $\alpha > d(1/p - 1/4)_+$ in the Besov space $\mathcal{B}^{\alpha}_{p,q}(\mathcal{X})$, considering value function learning (5) via DNNs defined by Eq. (4) in Algorithm 1 under the ϵ-greedy exploration and the mini-batch ratio $\varrho \in (0, 1)$, and taking*

$$\text{the depth } L \asymp \tfrac{d}{2\alpha+d} \log T, \qquad \text{the width } m \asymp \tfrac{d}{2\alpha+d} T^{\frac{d}{2\alpha+d}} \log T\,, \tag{7}$$

then given a MDP-dependent constant $K \in [1, H]$, for any $\delta \in (0, 1)$, the total regret can be upper bounded with probability at least $1 - \delta$

$$\begin{aligned} \textit{Regret}(T) &\lesssim \left(\frac{\epsilon}{A}\right)^{-\frac{K}{2}} \frac{1}{\sqrt{\varrho}} \left(H^{\frac{3}{2}} T^{\frac{\alpha+d}{2\alpha+d}} \log^3 T + H^2 \sqrt{T} \sqrt{\log\left(\frac{2}{\delta}\right) \log T} \right) + \epsilon H T + \sqrt{T H^3 \log\left(\frac{4}{\delta}\right)} \\ &\lesssim \widetilde{\mathcal{O}}\big(H^{\frac{H+4}{H+2}} K^{\frac{2}{K+2}} A^{\frac{K}{K+2}} T^{\frac{\alpha K + (\alpha+d)(K+2)}{(2\alpha+d)(K+2)}}\big)\,, \quad \textit{taking } \epsilon = \mathcal{O}\big((HK)^{\frac{2}{K+2}} A^{\frac{K}{K+2}} T^{-\frac{2\alpha}{(2\alpha+d)(K+2)}}\big)\,. \end{aligned} \tag{8}$$

Remark: We make the following remarks.
i) The constant K describes the "myopic" level of MDPs under the ϵ-greedy policy, e.g., the worst case ($K := H$) under the sparse rewards setting; the benign case $K := c$ (for some small constant c) under the helpful dense rewards setting as discussed in [49]. The exponential dependence on H (in the worst case for any MDP) can be avoided at an additional cost of worsening T dependence. In fact, whether in the benign/worst case, the sublinear regret is always achieved under some certain ϵ values in Eq. (8), which theoretically demonstrates the efficiency of deep RL. Note that the chosen $\epsilon \in (0, 1)$ is always satisfied under a large episode T.
ii) Clearly, the regret bound is a non-increasing function of the smoothness parameter α, which shows that an easier task (i.e., the target Q function is more smooth) leads to regret bounds with faster rates. Specially, if we take $\alpha \to \infty$ (i.e., the target Q function is sufficiently smooth), which holds for linear function approximation

$$\text{Regret}(T) \lesssim \widetilde{\mathcal{O}}\big(H^{\frac{H+4}{H+2}} K^{\frac{2}{K+2}} A^{\frac{K}{K+2}} T^{\frac{K+1}{K+2}}\big)\,,$$

which recovers the regret bound $\widetilde{\mathcal{O}}(T^{\frac{K+1}{K+2}})$ in [49, Theorem 3] via Eluder dimension. In the best case ($K = 1$), our regret bound implies $\widetilde{\mathcal{O}}(H^{\frac{4}{3}} A^{\frac{1}{3}} T^{\frac{2}{3}})$ with $H \geqslant 4$, which matches the optimal regret bound for the contextual bandits problem in terms of dependence on T or A under the ϵ-greedy exploration [66]. In the worst case ($K := H$), we can still obtain the sublinear regret at a certain $\widetilde{\mathcal{O}}(T^{\frac{H+1}{H+2}})$ rate.

Theorem 1 demonstrates that the sublinear regret can be achieved by choosing $\mathcal{O}(\log T)$ depth and $\widetilde{\mathcal{O}}(T^{\frac{d}{2\alpha+d}})$ width, but the sublinear regret bound $\widetilde{\mathcal{O}}(T^{\frac{\alpha K+(\alpha+d)(K+2)}{(2\alpha+d)(K+2)}})$ heavily depends on the feature dimension d, failing in the *curse of dimensionality*, which appears ineffective on high dimensional data in deep RL. In the next, we consider the Barron spaces, i.e., the "largest" function space for two-layer neural networks to avoid the curse of dimensionality. In this case, the rate of the sublinear regret can get rid of d, which is useful for high dimensional data in practical RL.

4.2 Efficient value iteration via two-layer neural networks in Barron spaces

As mentioned before, Barron spaces are the "largest" function space for two-layer neural networks. In this setting, we consider $\widehat{Q}_h^t = \operatorname{argmin}_{f\in\mathcal{F}_{\texttt{SNN}}} \widehat{\mathcal{E}}_h^t(f)$ in Eq. (5), where $\mathcal{F}_{\texttt{SNN}}$ is the function space of two-layer ReLU neural networks defined in Eq. (3). We give a similar assumption on the Bellman optimality operator in the Barron space.

Assumption 2. *Let $\widetilde{R} > 0$ be a fixed constant. Define $\mathcal{P}_{\widetilde{R}} = \{f \in \mathcal{P} : \|f\|_{\mathcal{P}} \leq \widetilde{R}\}$ in the Barron space, and assume that for any $h \in [H]$ and $Q\colon \mathcal{S}\times\mathcal{A}\to[0,H]$, we have $\mathbb{T}_h^\star Q \in \mathcal{P}_{\widetilde{R}}$.*

Based on this assumption, we have the following regret bounds for two-layer ReLU neural networks.

Theorem 2. *Under Assumption 2, considering value function learning* (5) *by two-layer ReLU neural networks with width m and bounded ℓ_1 norm B defined by Eq.* (3) *in Algorithm 1 under the ϵ-greedy exploration and the mini-batch ratio $\varrho \in (0,1)$, then given a MDP-dependent constant $K \in [1,H]$, for any $\delta \in (0,1)$, the total regret can be upper bounded with probability at least $1-\delta$*

$$\begin{aligned}\textit{Regret}(T) &\lesssim \left(\frac{\epsilon}{A}\right)^{-\frac{K}{2}}\left(\frac{H^2T^{\frac{3}{4}}}{\sqrt{\varrho}}\left[B(\log d)^{\frac{1}{4}} + \log^{\frac{1}{4}}\left(\frac{4}{\delta}\right)\right] + \frac{H^2T}{\sqrt{m}}\right) + \epsilon HT + \sqrt{TH^3\log\left(\frac{4}{\delta}\right)}\\ &\lesssim \widetilde{\mathcal{O}}(H^{\frac{K+4}{K+2}}T^{\frac{2K+3}{2K+4}}), \quad \textit{by taking } m=\Omega(\sqrt{T}) \textit{ and } \epsilon = \mathcal{O}\left(H^{\frac{2}{K+2}}T^{-\frac{1}{2(K+2)}}\right).\end{aligned}$$

Remark: In our result, taking $m = \Omega(\sqrt{T})$ is suffice to achieve the sublinear regret bound $\widetilde{\mathcal{O}}(T^{\frac{2K+3}{2K+4}})$, which also gets rid of the feature dimension d, allowing for high-dimensional image data in practice.

Proof outline: As mentioned before, the technical challenge in our analysis is how to estimate the TD error without bonus function design. Apart from the regret decomposition, our proof framework includes two main parts: 1) transformation of TD error estimation to generalization bounds and 2) generalization bounds on non-iid data in certain Besov/Barron spaces for TD error analysis.

To bound the TD error, we first prove $\bar{\mu}_h^{\tilde{t}}(\mathcal{C}) > 0$, where the event $\mathcal{C}$ denotes that all state-action pairs have been visited at all time steps under the ϵ-greedy policy. Then the TD error is formulated in the $L^2(\mathrm{d}\bar{\mu}_h^{\tilde{t}})$-integrable space, and thus transformed to generalization error and approximation error in the respective Barron or Besov spaces.

The key part left is to bound the generalization error on non-i.i.d data for the TD error estimation, and we prove that the classical uniform convergence (e.g., Rademacher complexity, covering number) is still valid under our setting. In our proof, we firstly verify that the maximum error in estimating the mean of any function $f \in \mathcal{F}$ can be still bounded by the Rademacher complexity of $\mathcal{F}$, and then generalization bounds by Rademacher complexity still holds via the averaged measure $\bar{\mu}_h^{\tilde{t}}$, which only requires the data to be independent. These results can be easily extended to local Rademacher complexity.

Regret bounds effected by optimization error: Here we briefly discuss the regret bound affected by a solution (denoted as $\widetilde{Q}_h^t$) that is not a global minimum of problem (5). Assume that the optimization error is small in the functional view, i.e., $\|\widetilde{Q}_h^t - \widehat{Q}_h^t\|_{L^2(\mathrm{d}\bar{\mu}_h^{\tilde{t}})} \leqslant \varepsilon_{\mathrm{opt}}$, that will appear in our analysis, and accordingly the TD error incurs in an extra regret bound $\mathcal{O}(H^2\log T)$ if we take $\varepsilon_{\mathrm{opt}} := H/\sqrt{\tilde{t}}$. This condition is fair and reasonable as the optimization error decreases with the mini-batch size $\tilde{t}$ for neural network training but requires a refined analysis under non-iid data [67, 68].

5 Discussion on architecture guidelines in deep RL

In this section, we present a detailed discussion on how our results provide the architecture guidelines in practical deep RL, in the perspective of the width, the depth, and problem-dependent smoothness of the Q function.

Width-depth and DQN: According to Theorem 1, the $\mathcal{O}(\log T)$ depth and $\widetilde{O}(T^{\frac{d}{2\alpha+d}})$ width are enough for sublinear regret in deep RL. Interestingly, we notice that this result is closely matching practical implementation of DQN. For example, the choices of [4] $m = 512$ and $L = 5$ can be explained by our theory, indeed $\log(512) \approx 6$. Specially, when taking $\alpha \to \infty$, this setting holds for linear function approximation. For two-layer neural networks endowed by the Barron space, the curse of dimensionality in terms of width and regret bound can be avoided in Theorem 2, supporting the premise of practical, high-dimensional RL.

Problem-dependent smoothness and exploration: The problem-dependent smoothness, determined by α, largely affects our regret bounds. The difficulty of a task in deep RL can be defined in two views: one is the smoothness of the target Q function; and the other is the degree of exploration. Intuitively speaking, if a RL task is difficult, then the target Q function is often complicated, and thus admits a relative lower smoothness; or we need conduct more exploration in a complex scenario. Our results coincide with these two views. One hand, the regret bound in Theorem 1 is a non-increasing function of the smoothness parameter α. A more difficult task in deep RL (i.e., a smaller α) leads to a slower rate of the sublinear regret, which indicates that more episodes are required. On the other hand, Theorem 1 shows that the parameter ϵ is also a non-increasing function of α. That means, a more difficult task in deep RL requires a larger ϵ, i.e., we need conduct exploration more frequently.

Besides, the exploration parameter ϵ is also affected by K for MDPs with different situations. For example, compared to the best case $K = 1$, more frequent exploration (a larger α) is required in MDPs under difficult cases, which coincides with our certain ϵ value in Theorems 1 and 2.

Width and depth trade-off: Under a limit parameter budget, according to the width-depth ratio $m/L = T^{\frac{d}{2\alpha+d}}$ in Theorem 1, our theory indicates that less problem-dependent smoothness of Q-function requires DNNs to be wider. In practice, if we work in the limited budget of parameters N in neural networks, e.g., $N \asymp m^2 L$, our theory implies that there is a tradeoff between the depth and width on smoothness, i.e., the depth $L := N^{1/3} T^{-\frac{2d}{3(2\alpha+d)}}$ increasing with α (or T) and the width $m = N^{1/3} T^{\frac{d}{3(2\alpha+d)}}$ decreasing with α (or T).

Besides, according to the width-depth ratio, it can be found that, the change of α leads to less changes on the depth but more changes on the width. This shows that width and depth admit different levels of parameter sensitivity under the change of problem-dependent smoothness.

6 Conclusion

This paper provides an in-depth understanding on neural network function approximation with the ϵ-greedy exploration under the online setting beyond the "linear" regime. Our results provide theoretical guarantees of sublinear regret bounds, and shed light on some guidelines for understanding deep RL in the perspective of the width-depth configuration and the problem-dependent smoothness of RL tasks.

The analysis of this work is built on the ϵ-greedy policy for exploration, which are satisfied in practical cases when employing DQN. Nevertheless, designing a provably efficient exploration mechanism for deep RL could be an interesting future direction in both practice and theory. Besides, our theory requires state-action pairs to be independent, which (approximately) holds via experience replay and could be improved by reverse experience replay [69]. Furthermore, our work is built on the value iteration based algorithm, which is different from practical DQN that adapts Q-learning via one-step gradient descent. Towards a better understanding DQN in terms of Q learning and target networks [43, 44] would be an interesting direction.

Acknowledgement

This work was supported by SNF project – Deep Optimisation of the Swiss National Science Foundation (SNSF) under grant number 200021_205011; the Enterprise for Society Center (E4S);

the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement n°725594 - time-data).

References

[1] Csaba Szepesvári. Algorithms for reinforcement learning. *Synthesis lectures on artificial intelligence and machine learning*, 4(1):1–103, 2010. 1

[2] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018. 1

[3] Scott Fujimoto, Herke Hoof, and David Meger. Addressing function approximation error in actor-critic methods. In *International conference on machine learning*, pages 1587–1596. PMLR, 2018. 1

[4] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015. 1, 2, 5, 9

[5] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of Go with deep neural networks and tree search. *Nature*, 529(7587):484–489, 2016. 1

[6] LeCun Yann, Bengio Yoshua, and Hinton Geoffrey. Deep learning. *Nature*, 521(7553):436–444, 2015. 1

[7] Chi Jin, Zhuoran Yang, Zhaoran Wang, and Michael I Jordan. Provably efficient reinforcement learning with linear function approximation. In *Conference on Learning Theory*, pages 2137–2143. PMLR, 2020. 1, 2, 3, 7, 16

[8] Yining Wang, Ruosong Wang, Simon Shaolei Du, and Akshay Krishnamurthy. Optimism in reinforcement learning with generalized linear function approximation. In *International Conference on Learning Representations*, 2020. 1

[9] Alex Ayoub, Zeyu Jia, Csaba Szepesvari, Mengdi Wang, and Lin Yang. Model-based reinforcement learning with value-targeted regression. In *International Conference on Machine Learning*, pages 463–474. PMLR, 2020. 1

[10] Dongruo Zhou, Quanquan Gu, and Csaba Szepesvari. Nearly minimax optimal reinforcement learning for linear mixture markov decision processes. In *Conference on Learning Theory*, pages 4532–4576. PMLR, 2021. 1, 3

[11] Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks. In *Advances in Neural Information Processing Systems*, pages 8571–8580, 2018. 1

[12] Zhuoran Yang, Chi Jin, Zhaoran Wang, Mengdi Wang, and Michael I Jordan. On function approximation in reinforcement learning: Optimism in the face of large state spaces. In *Advances in Neural Information Processing Systems*, 2020. 1, 2, 3, 6, 7, 16, 17, 18, 19

[13] Nan Jiang, Akshay Krishnamurthy, Alekh Agarwal, John Langford, and Robert E Schapire. Contextual decision processes with low bellman rank are pac-learnable. In *International Conference on Machine Learning*, pages 1704–1713. PMLR, 2017. 1, 3

[14] Kefan Dong, Jian Peng, Yining Wang, and Yuan Zhou. Root-n-regret for learning in markov decision processes with function approximation and low bellman rank. In *Conference on Learning Theory*, pages 1554–1557. PMLR, 2020. 1, 3

[15] Daniel Russo and Benjamin Van Roy. Eluder dimension and the sample complexity of optimistic exploration. *Advances in Neural Information Processing Systems*, 26, 2013. 1, 3

[16] Ruosong Wang, Russ R Salakhutdinov, and Lin Yang. Reinforcement learning with general value function approximation: Provably efficient approach via bounded eluder dimension. In *Advances in Neural Information Processing Systems*, volume 33, pages 6123–6135, 2020. 1, 3

[17] Haque Ishfaq, Qiwen Cui, Viet Nguyen, Alex Ayoub, Zhuoran Yang, Zhaoran Wang, Doina Precup, and Lin Yang. Randomized exploration in reinforcement learning with general value function approximation. In *International Conference on Machine Learning*, pages 4607–4616. PMLR, 2021. 1, 3

[18] Lenaic Chizat, Edouard Oyallon, and Francis Bach. On lazy training in differentiable programming. In *Advances in Neural Information Processing Systems*, pages 2933–2943, 2019. 1

[19] Jaehoon Lee, Lechao Xiao, Samuel Schoenholz, Yasaman Bahri, Roman Novak, Jascha Sohl-Dickstein, and Jeffrey Pennington. Wide neural networks of any depth evolve as linear models under gradient descent. In *Advances in Neural Information Processing Systems*, pages 8570–8581, 2019. 1

[20] Blake Woodworth, Suriya Gunasekar, Jason D Lee, Edward Moroshko, Pedro Savarese, Itay Golan, Daniel Soudry, and Nathan Srebro. Kernel and rich regimes in overparametrized models. In *Conference on Learning Theory*, pages 3635–3673. PMLR, 2020. 1

[21] Mario Geiger, Stefano Spigler, Arthur Jacot, and Matthieu Wyart. Disentangling feature and lazy training in deep neural networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2020(11):113301, 2020. 1

[22] Francis Bach. Breaking the curse of dimensionality with convex neural networks. *Journal of Machine Learning Research*, 18(1):629–681, 2017. 1

[23] Gilad Yehudai and Ohad Shamir. On the power and limitations of random features for understanding neural networks. In *Advances in Neural Information Processing Systems*, pages 6594–6604, 2019. 1

[24] Michael Celentano, Theodor Misiakiewicz, and Andrea Montanari. Minimum complexity interpolation in random features models. *arXiv preprint arXiv:2103.15996*, 2021. 1

[25] Baihe Huang, Kaixuan Huang, Sham Kakade, Jason D Lee, Qi Lei, Runzhe Wang, and Jiaqi Yang. Going beyond linear rl: Sample efficient neural function approximation. In *Advances in Neural Information Processing Systems*, 2021. 2, 3

[26] Gene Li, Pritish Kamath, Dylan J Foster, and Nathan Srebro. Eluder dimension and generalized rank. *arXiv preprint arXiv:2104.06970*, 2021. 2

[27] Kefan Dong, Jiaqi Yang, and Tengyu Ma. Provable model-based nonlinear bandit and reinforcement learning: Shelve optimism, embrace virtual curvature. In *Advances in Neural Information Processing Systems*, volume 34, 2021. 2

[28] Ian Osband, Benjamin Van Roy, Daniel J Russo, Zheng Wen, et al. Deep exploration via randomized value functions. *Journal of Machine Learning Research*, 20(124):1–62, 2019. 2, 5

[29] Todd Hester, Matej Vecerik, Olivier Pietquin, Marc Lanctot, Tom Schaul, Bilal Piot, Dan Horgan, John Quan, Andrew Sendonaris, Ian Osband, et al. Deep q-learning from demonstrations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018. 2

[30] Taiji Suzuki. Adaptivity of deep ReLU network for learning in Besov and mixed smooth Besov spaces: optimal rate and curse of dimensionality. In *International Conference on Learning Representations*, 2019. 2, 4, 5, 15, 17, 26

[31] Weinan E, Chao Ma, and Lei Wu. The barron space and the flow-induced function spaces for neural network models. *Constructive Approximation*, pages 1–38, 2021. 2, 4, 5, 16, 29, 30

[32] Lin Yang and Mengdi Wang. Reinforcement learning in feature space: Matrix bandit, kernels, and regret bound. In *International Conference on Machine Learning*, pages 10746–10756. PMLR, 2020. 2, 3

[33] Daniel J Russo, Benjamin Van Roy, Abbas Kazerouni, Ian Osband, Zheng Wen, et al. A tutorial on thompson sampling. *Foundations and Trends® in Machine Learning*, 11(1):1–96, 2018. 2, 3

[34] Shipra Agrawal and Navin Goyal. Analysis of thompson sampling for the multi-armed bandit problem. In *Conference on Learning Theory*, pages 1–39. JMLR Workshop and Conference Proceedings, 2012. 2, 3

[35] Andrea Zanette, David Brandfonbrener, Emma Brunskill, Matteo Pirotta, and Alessandro Lazaric. Frequentist regret bounds for randomized least-squares value iteration. In *International Conference on Artificial Intelligence and Statistics*, pages 1954–1964. PMLR, 2020. 2, 3

[36] Long-Ji Lin. Self-improving reactive agents based on reinforcement learning, planning and teaching. *Machine learning*, 8(3):293–321, 1992. 2, 5

[37] Jianqing Fan, Zhaoran Wang, Yuchen Xie, and Zhuoran Yang. A theoretical analysis of deep q-learning. In *Learning for Dynamics and Control*, pages 486–489, 2020. 2, 3, 4, 5

[38] Jihao Long, Jiequn Han, et al. An l^2 analysis of reinforcement learning in high dimensions with kernel and neural network approximation. *arXiv preprint arXiv:2104.07794*, 2021. 3

[39] Thanh Nguyen-Tang, Sunil Gupta, Hung Tran-The, and Svetha Venkatesh. Sample complexity of offline reinforcement learning with deep relu networks. *arXiv preprint arXiv:2103.06671*, 2021. 3

[40] Martin Riedmiller. Neural fitted Q iteration–first experiences with a data efficient neural reinforcement learning method. In *European Conference on Machine Learning*, pages 317–328. Springer, 2005. 3

[41] Pan Xu and Quanquan Gu. A finite-time analysis of q-learning with neural network function approximation. In *International Conference on Machine Learning*, pages 10555–10565. PMLR, 2020. 3

[42] Chi Jin, Zeyuan Allen-Zhu, Sebastien Bubeck, and Michael I Jordan. Is q-learning provably efficient? In *Advances in neural information processing systems*, 2018. 3

[43] Andrea Zanette and Martin J Wainwright. Stabilizing Q-learning with linear architectures for provably efficient learning. *arXiv preprint arXiv:2206.00796*, 2022. 3, 9

[44] Diogo Carvalho, Francisco S Melo, and Pedro Santos. A new convergent variant of Q-learning with linear function approximation. In *Advances in Neural Information Processing Systems*, volume 33, pages 19412–19421, 2020. 3, 5, 9

[45] Naman Agarwal, Syomantak Chaudhuri, Prateek Jain, Dheeraj Nagaraj, and Praneeth Netrapalli. Online target Q-learning with reverse experience replay: Efficiently finding the optimal policy for linear mdps. *arXiv preprint arXiv:2110.08440*, 2021. 3

[46] Liran Szlak and Ohad Shamir. Convergence results for Q-learning with experience replay. *arXiv preprint arXiv:2112.04213*, 2021. 3

[47] Andrea Zanette, Alessandro Lazaric, Mykel Kochenderfer, and Emma Brunskill. Learning near optimal policies with low inherent bellman error. In *International Conference on Machine Learning*, pages 10978–10989. PMLR, 2020. 3

[48] Pihe Hu, Yu Chen, and Longbo Huang. Nearly minimax optimal reinforcement learning with linear function approximation. In *International Conference on Machine Learning*, pages 8971–9019, 2022. 3

[49] Chris Dann, Yishay Mansour, Mehryar Mohri, Ayush Sekhari, and Karthik Sridharan. Guarantees for epsilon-greedy reinforcement learning with function approximation. In *International Conference on Machine Learning*, pages 4666–4689, 2022. 3, 7, 19

[50] Chi Jin, Qinghua Liu, and Sobhan Miryoosefi. Bellman eluder dimension: New rich classes of rl problems, and sample-efficient algorithms. *Advances in Neural Information Processing Systems*, 34, 2021. 3

[51] Martin L Puterman. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014. 4

[52] Minshuo Chen, Haoming Jiang, Wenjing Liao, and Tuo Zhao. Efficient approximation of deep ReLU networks for functions on low dimensional manifolds. *Advances in neural information processing systems*, 32:8174–8184, 2019. 4, 6

[53] Ahmed Abdeljawad and Philipp Grohs. Approximations with deep neural networks in sobolev time-space. *arXiv preprint arXiv:2101.06115*, 2020. 4

[54] Yoshihiro Sawano. *Theory of Besov spaces*, volume 56. Springer, 2018. 4, 15

[55] Weinan E and Stephan Wojtowytsch. Representation formulas and pointwise properties for barron functions. *arXiv preprint arXiv:2006.05982*, 2020. 5, 16

[56] Boris Hanin and David Rolnick. Deep relu networks have surprisingly few activation patterns. *Advances in Neural Information Processing Systems*, 32, 2019. 5

[57] Shirli Di-Castro, Shie Mannor, and Dotan Di Castro. Analysis of stochastic processes through replay buffers. In *International Conference on Machine Learning*, pages 5039–5060. PMLR, 2022. 5

[58] Johannes Schmidt-Hieber. Nonparametric regression using deep neural networks with ReLU activation function. *Annals of Statistics*, 48(4):1875–1897, 2020. 6

[59] Taiji Suzuki and Atsushi Nitanda. Deep learning is adaptive to intrinsic dimensionality of model smoothness in anisotropic besov space. In *Advances in Neural Information Processing Systems*, 2021. 6

[60] András Antos, Csaba Szepesvári, and Rémi Munos. Learning near-optimal policies with bellman-residual minimization based fitted policy iteration and a single sample path. *Machine Learning*, 71(1):89–129, 2008. 6

[61] Yaqi Duan, Chi Jin, and Zhiyuan Li. Risk bounds and rademacher complexity in batch reinforcement learning. In *International Conference on Machine Learning*, pages 2892–2902. PMLR, 2021. 6, 21

[62] Steven J Bradtke and Andrew G Barto. Linear least-squares algorithms for temporal difference learning. *Machine learning*, 22(1):33–57, 1996. 6

[63] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, 2013. 6

[64] Seungchan Kim, Kavosh Asadi, Michael Littman, and George Konidaris. Deepmellow: removing the need for a target network in deep q-learning. In *International Joint Conference on Artificial Intelligence*, 2019. 6

[65] Qi Cai, Zhuoran Yang, Chi Jin, and Zhaoran Wang. Provably efficient exploration in policy optimization. In *International Conference on Machine Learning*, pages 1283–1294, 2020. 6, 16, 17, 18, 19

[66] Tor Lattimore and Csaba Szepesvári. *Bandit algorithms*. Cambridge University Press, 2020. 7

[67] Suhas Kowshik, Dheeraj Nagaraj, Prateek Jain, and Praneeth Netrapalli. Streaming linear system identification with reverse experience replay. In *Advances in Neural Information Processing Systems*, pages 30140–30152, 2021. 8

[68] Ahmet Alacaoglu and Hanbaek Lyu. Convergence and complexity of stochastic subgradient methods with dependent data for nonconvex optimization. *arXiv preprint arXiv:2203.15797*, 2022. 8

[69] Naman Agarwal, Syomantak Chaudhuri, Prateek Jain, Dheeraj Nagaraj, and Praneeth Netrapalli. Online target Q-learning with reverse experience replay: Efficiently finding the optimal policy for linear MDPs. In *International Conference on Learning Representations*, 2022. 9

[70] Behnam Neyshabur, Ryota Tomioka, and Nathan Srebro. Norm-based capacity control in neural networks. In *Conference on Learning Theory*, pages 1376–1401. PMLR, 2015. 16

[71] Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of machine learning*. MIT press, 2018. 22

[72] Martin J Wainwright. *High-dimensional statistics: A non-asymptotic viewpoint*, volume 48. Cambridge University Press, 2019. 23, 29

[73] Shahar Mendelson. Improving the sample complexity using global data. *IEEE transactions on Information Theory*, 48(7):1977–1991, 2002. 23, 25

[74] Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge University Press, 2014. 23, 24, 25

[75] Peter L Bartlett, Olivier Bousquet, and Shahar Mendelson. Local rademacher complexities. *Annals of Statistics*, 33(4):1497–1537, 2005. 25

[76] Yunwen Lei, Lixin Ding, and Yingzhou Bi. Local rademacher complexity bounds based on covering numbers. *Neurocomputing*, 218:320–330, 2016. 25

The appendix is organized as follows

- Appendix A: preliminaries on Besov spaces and Barron spaces;
- Appendix B: an overview of our proof framework;
- Appendix C: proofs related to regret decomposition;
- Appendix D: proofs related to the temporal difference error and generalization error;
- Appendix E: proofs related to generalization bounds on non-iid data;
- Appendix F: proofs related to sublinear regret bounds for deep ReLU neural networks endowed by Besov spaces;
- Appendix G: proofs related to sublinear regret bounds for two-layer neural networks endowed by Barron spaces.

A Preliminaries: Besov spaces and Barron spaces

In this section, we give an overview of Besov spaces for deep ReLU neural networks and the Barron spaces for two-layer ReLU neural networks.

A.1 Besov spaces

Here we briefly introduce a general function space for deep ReLU neural networks according to the "smoothness" of the function, i.e., Besov spaces.

To define Besov functions, we need introduce the modulus of smoothness.

Definition 1. *[30, modulus of smoothness] For a function $f \in L^p(\mathcal{X})$ with some $p \in (0, \infty]$, the k-th modulus of smoothness of f is defined by*

$$w_{k,p}(f,t) = \sup_{\boldsymbol{h} \in \mathbb{R}^d:\ \|\boldsymbol{h}\|_2 \leqslant t} \|\Delta^k_{\boldsymbol{h}}(f)\|_p \,,$$

with

$$\Delta^k_{\boldsymbol{h}}(f)(\boldsymbol{x}) = \begin{cases} \sum_{j=0}^k \binom{k}{j} (-1)^{k-j} f(\boldsymbol{x} + j\boldsymbol{h}) & \text{if } \boldsymbol{x} \in \mathcal{X},\ \boldsymbol{x} + k\boldsymbol{h} \in \mathcal{X}\,, \\ 0 & \text{otherwise}. \end{cases}$$

The quantity $\Delta^k_{\boldsymbol{h}}(f)$ captures the local oscillation of function f that is not necessarily differentiable. Based on this, the Besov space is defined as below.

Definition 2. *[54, 30, Besov space $\mathcal{B}^\alpha_{p,q}(\mathcal{X})$] For $0 < p, q \leqslant \infty$, the smoothness parameter $\alpha > 0$, $k := \lfloor \alpha \rfloor + 1$, define the semi-norm $|\cdot|_{\mathcal{B}^\alpha_{p,q}}$ as*

$$|f|_{\mathcal{B}^\alpha_{p,q}} := \begin{cases} \left(\int_0^\infty (t^{-\alpha} w_{k,p}(f,t))^q \frac{\mathrm{d}t}{t} \right)^{\frac{1}{q}} & (q < \infty)\,, \\ \sup_{t>0} t^{-\alpha} w_{k,p}(f,t) & (q = \infty)\,. \end{cases}$$

The norm of the Besov space $\mathcal{B}^\alpha_{p,q}(\mathcal{X})$ is defined by $\|f\|_{\mathcal{B}^\alpha_{p,q}} := \|f\|_{L^p(\mathcal{X})} + |f|_{\mathcal{B}^\alpha_{p,q}}$, and the Besov space is $\mathcal{B}^\alpha_{p,q}(\mathcal{X}) = \{f \in L^p(\mathcal{X}) \mid \|f\|_{\mathcal{B}^\alpha_{p,q}} < \infty\}$.

The smoothness parameter α indicates which function at a certain smoothness degree can be represented. For example, if $\alpha > d/p$, then the related Besov space is continuously embedded in the set of the continuous functions. However, if $\alpha < d/p$, then the functions in the Besov space are no longer continuous. In particular, the Besov space reduces to the Hölder space $\mathtt{C}^\alpha$ when $p = q = \infty$ and α is a positive non-integer; degenerates to the Sobolev space $\mathtt{W}^\alpha_2$ when $p = q = 2$ and α is a positive integer. The Besov space is more general than these two spaces as it allows for spatially inhomogeneous smoothness with spikes and jumps. More properties of Besov spaces and relations to other function spaces refer to [30] for details.

$$\begin{array}{l}\text{Regret}\\ \text{decomp.}\\ \text{Lem. 1}\end{array}\begin{cases}\text{statistical error: Lem. 2 with } \mathcal{O}(\sqrt{H^3T})\\ \texttt{Term (i)} \Leftarrow \text{Lem. 5} \begin{cases}\text{generalization}\\ \text{approximation}\end{cases} \Leftarrow \text{Lem. 4: } \|\Gamma_h^t\|^2_{L^2(\mathrm{d}\bar{\mu}_h^{\tilde{t}})} \Leftarrow \text{Lem. 3: } \bar{\mu}_h^{\tilde{t}}(\mathcal{C})>0 \\ \epsilon\text{-greedy exploration: } \epsilon HT\end{cases}$$

Figure 1: Proof framework of regret decomposition and transformation of the TD error.

A.2 Barron spaces

The study for deep ReLU neural networks is endowed by Besov spaces, but the complete of function space for deep ReLU neural networks to avoid the *curse of dimensionality* is still open. Luckily, the complete of function space for two-layer neural networks can be conducted by Barron spaces. Here we briefly introduce the basic definition and property of Barron spaces [55, 31].

We consider a typical two-layer neural network $f(\boldsymbol{x}) = \frac{1}{m}\sum_{k=1}^{m} b_k \sigma(\boldsymbol{w}_k^\top \boldsymbol{x} + c_k)$, where m is the number of neurons in the hidden layer and $\sigma(x) = \max\{x, 0\}$ is the ReLU activation function used in this work. Accordingly, the two-layer neural network admits the following representation

$$f(\boldsymbol{x}) = \int_\Omega b\sigma\left(\boldsymbol{w}^\top \boldsymbol{x} + c\right)\rho(\mathrm{d}b, \mathrm{d}\boldsymbol{w}, \mathrm{d}c), \quad \boldsymbol{x} \in \mathcal{X}\,, \tag{9}$$

where $\Omega = \mathbb{R} \times \mathbb{R}^d \times \mathbb{R}$ and ρ is a probability measure over Ω. Then the Barron space [31] endowed by the p-Barron norm with $p \in [1, +\infty]$ is defined as

$$\widetilde{\mathcal{P}}_p = \left\{ f \;\text{ admits Eq. (9)} : \|f\|_{\widetilde{\mathcal{P}}_p} = \inf_\rho \left\{\mathbb{E}_\rho |b|^p (\|\boldsymbol{w}\|_1 + |c|)^p\right\}^{1/p} < \infty \right\}.$$

Specifically, when using ReLU, these function spaces under different p are the same, i.e., $\widetilde{\mathcal{P}}_1 = \widetilde{\mathcal{P}}_2 = \cdots = \widetilde{\mathcal{P}}_\infty$, and thus we directly use $\widetilde{\mathcal{P}}$ for short. The is the main reason why we study ReLU activation functions in this work. Besides, the Barron norm is close to the ℓ_1-path norm [70]

$$\|f\|_{\widetilde{\mathcal{P}}} \leqslant \|f\|_{\mathcal{P}} := \frac{1}{m}\sum_{k=1}^{m} |b_k|(\|\boldsymbol{w}_k\|_1 + |c_k|) \leqslant 2\|f\|_{\widetilde{\mathcal{P}}}\,.$$

Based on this, for description simplicity, we do not strictly distinguish the Barron norm and the ℓ_1-path norm, and regard $\|f\|_{\mathcal{P}}$ as the discrete version of the Barron norm.

As suggested by [55], Barron space can be regarded as the *largest* function space for two layer neural networks in two folds [31]: 1) *direct approximation:* Any function in Barron spaces can be efficiently approximated by two-layer neural networks with bounded ℓ_1 path norm at $\mathcal{O}(1/m)$ rate without *curse of dimensionality*; 2) *inverse approximation:* Any continuous function that can be efficiently approximated by two-layer neural networks with bounded ℓ_1-path norm belongs to a Barron space.

B Proof outline

In this section, we outline the proof of our theoretical results presented in Section 4. As mentioned before, the technical challenge in our analysis is how to estimate the TD error without bonus function design. Apart from the regret decomposition, our proof framework includes two main parts: transformation of TD error estimation to generalization bounds, see Figure 1; and generalization bounds on non-iid data in certain Besov/Barron spaces for TD error analysis, see Figure 2. The complete proof is reported in the appendix.

Regret decomposition: This part is standard and commonly studied in RL theory, e.g., [65, 7, 12]. We briefly include here for self-completeness. Define the temporal-difference (TD) error as

$$\Gamma_h^t(s,a) = r_h(s,a) + (\mathbb{P}_h V_{h+1}^t)(s,a) - Q_h^t(s,a) = (\mathbb{T}_h V_{h+1}^t)(s,a) - Q_h^t(s,a), \quad \forall (s,a) \in \mathcal{S}\times\mathcal{A}\,, \tag{10}$$

where Γ_h^t is a function on $\mathcal{S} \times \mathcal{A}$ for all $h \in [H]$ and $t \in [T]$. Accordingly, the regret can be decomposed into (*c.f.* Lemma 1)

$$\text{Regret}(T) \leqslant \underbrace{\sum_{t=1}^{T}\sum_{h=1}^{H}\Big(\mathbb{E}_{\pi^\star}[\Gamma_h^t(s_h,a_h)\,|\,s_1 = s_1^t] - \Gamma_h^t(s_h^t,a_h^t)\Big)}_{\texttt{Term (i)}} + \texttt{Term (ii)} + \epsilon HT\,, \tag{11}$$

$$\begin{cases}\text{Rademacher complexity on non-iid data: Lem. 7} \Leftarrow \text{Lem. 6} \\ \text{two-layer NNs: Thm. 2} \Leftarrow \mathcal{O}(H^2 B\sqrt{\log d/n}) \Leftarrow \text{Lem. 13: Rademacher complexity of Barron spaces} \\ \text{DNNs: Thm. 1} \Leftarrow \mathcal{O}(n^{-\frac{2\alpha}{2\alpha+d}}) \Leftarrow \text{Prop. 2} \Leftarrow \text{Lem. 12 on } \texttt{LRC} \text{ for Besov spaces}\end{cases}$$

Figure 2: Proof framework of the TD error via generalization bounds on n non-iid data. We denote LRC by local Rademacher complexity for short.

where the first term relates to the TD error and the second term is the statistical error based on the standard martingale difference sequences, which can be upper bounded by the Hoeffding-Azuma inequality with $\mathcal{O}(\sqrt{H^3T})$ regret (*c.f.* Lemma 2). The last term ϵHT is due to the ϵ-greedy exploration.

Transforming TD error to generalization bounds: To bound the TD error, we first introduce Lemma 3 with $\bar{\mu}_h^{\tilde{t}}(\mathcal{C}) > 0$, where the event $\mathcal{C}$ denotes that all state-action pairs have been visited at all time steps under the ϵ-greedy policy. Then we are able to build the connection between Term (i) and $\|\Gamma_h^t\|_{L^2(\mathrm{d}\bar{\mu}_h^{\tilde{t}})}$ in the $L^2(\mathrm{d}\bar{\mu}_h^{\tilde{t}})$-integrable space (*c.f.* Lemma 4). After analysis of $\mathcal{E}_h^t(f)$ in Proposition 1, we transform the estimation of $\|\Gamma_h^t\|_{L^2(\mathrm{d}\bar{\mu}_h^{\tilde{t}})}$ to the following two terms: generalization error and approximation error, respectively (*c.f.* Lemma 5)

$$\|\Gamma_h^t\|^2_{L^2(\mathrm{d}\bar{\mu}_h^{\tilde{t}})} \leqslant \left[\mathcal{E}_h^t(\widehat{Q}_h^t) - \min_{f\in\mathcal{F}} \mathcal{E}_h^t(f)\right] + \inf_{f\in\mathcal{F}} \|f - \mathbb{T}_h^\star Q_{h+1}^t\|^2_{L^2(\mathrm{d}\bar{\mu}_h^{\tilde{t}})} \,. \tag{12}$$

where the first term is the generalization error which we elucidate in the next and the second term is the approximation error and can be considered in an $L^p(\mathcal{X})$ space for Besov spaces in Corollary 1. For example, the approximation error in the Besov space admits the certain $\mathcal{O}(N^{-2\alpha/d})$ rate in [30] for deep ReLU networks with $L \asymp \log N,\ S \asymp N,\ m \asymp N\log N$.

Generalization bounds on non-iid data: The key part left is to bound the generalization error on non-i.i.d data for the TD error estimation, see the proof framework in Figure 2. In our proof, we firstly verify that the maximum error in estimating the mean of any function $f \in \mathcal{F}$ can be still bounded by the Rademacher complexity of $\mathcal{F}$ in Lemma 6, and then generalization bounds by Rademacher complexity still holds by Lemma 7 via the averaged measure $\bar{\mu}_h^{\tilde{t}}$, which only requires the data to be independent. These results can be easily extended to local Rademacher complexity.

For deep neural networks, by computing the local Rademacher complexity of $\mathcal{F}_{\texttt{DNN}}$ in Lemma 12 and choosing proper neural network parameters in Eq. (4), we derive the convergence rate of generalization bounds at a certain $\mathcal{O}(n^{-\frac{2\alpha}{2\alpha+d}})$ rate in Besov spaces (*c.f.* Proposition 2) with n non-iid data. Combining the result of approximation error and taking the depth and width in Eq. (7), Term (i) can be upper bounded with high probability. Finally we conclude the proof of Theorem 1 by combining with the statistical error.

For two-layer neural networks, by computing the Rademacher complexity of $\mathcal{F}_{\texttt{SNN}}$ in Lemma 13, we obtain the generalization error at a certain $\mathcal{O}(H^2 B\sqrt{\log d/n})$ convergence rate. Combining the result of approximation error in Barron spaces with other terms in the regret decomposition, we conclude the proof of Theorem 2.

C Regret decomposition

We present the regret decomposition under the ϵ-greedy policy by constructing the martingale difference sequence and giving error bounds for this. Apart from an extra ϵHT regret, this decomposition result appears in [65, 12], and we include them here just for self-completeness.

To establish the regret decomposition, we need some notations. Remember the definition of the regret, $\tilde{\pi}^t$ is the ϵ-greedy policy and π^t is the greedy policy at the t-th episode, and then we have

$$\begin{aligned}\text{Regret}(T) &= \sum_{t=1}^{T}\big[V_1^\star(s_1^t) - V_1^{\pi^t}(s_1^t)\big] + \sum_{t=1}^{T}\big[V_1^{\pi^t}(s_1^t) - V_1^{\tilde{\pi}^t}(s_1^t)\big] \\ &\leqslant \sum_{t=1}^{T}\big[V_1^\star(s_1^t) - V_1^{\pi^t}(s_1^t)\big] + \epsilon HT\,,\end{aligned}$$

where ϵHT stems from the fact that the return of greedy and ϵ-greedy policies can differ at most ϵH in each episode. In the next, we aim to estimate the first term in the above equation. It involves the greedy policy π^t at the t-th episode, which leads to a trajectory $\{(s_h^t, a_h^t)\}_{h=1}^H$. Note that this trajectory is different from Algorithm 1 that uses the ϵ-greedy policy but we use the same notation on state-action pairs for notational simplicity in this section.

Following [65, 12], we define two quantities $\zeta_{t,h}^1, \zeta_{t,h}^2 \in \mathbb{R}$ for any $h \in [H]$ and $t \in [T]$ based on the greedy policy

$$\begin{aligned} \zeta_{t,h}^1 &:= [V_h^t(s_h^t) - V_h^{\pi^t}(s_h^t)] - [Q_h^t(s_h^t, a_h^t) - Q_h^{\pi^t}(s_h^t, a_h^t)]\,, \\ \zeta_{t,h}^2 &:= [(\mathbb{P}_h V_{h+1}^t)(s_h^t, a_h^t) - (\mathbb{P}_h V_{h+1}^{\pi^t})(s_h^t, a_h^t)] - [V_{h+1}^t(s_{h+1}^t) - V_{h+1}^{\pi^t}(s_{h+1}^t)]\,. \end{aligned} \tag{13}$$

By definition, $\zeta_{t,h}^1$ depends on the randomness of choosing an action $a_h^t \sim \pi_h^t(\cdot|s_h^t)$; and $\zeta_{t,h}^2$ captures the stochastic transition, i.e., the randomness of drawing the next state s_{h+1}^t from $\mathbb{P}_h(\cdot|s_h^t, a_h^t)$. Based on the following definition of filtration, $\{\zeta_{t,h}^1, \zeta_{t,h}^2\}$ forms a bounded martingale difference sequence.

Definition 3. *[65, Filtration] For any $(t,h) \in [T] \times [H]$, define σ-algebras $\mathcal{M}_{t,h,1}$ and $\mathcal{M}_{t,h,2}$ generated by the following respective state-action sequence as*

$$\begin{aligned} \mathcal{M}_{t,h,1} &:= \sigma\big(\{(s_i^\tau, a_i^\tau)\}_{(\tau,i)\in[t-1]\times[H]} \cup \{(s_i^t, a_i^t)\}_{i\in[h]}\big), \\ \mathcal{M}_{t,h,2} &:= \sigma\big(\{(s_i^\tau, a_i^\tau)\}_{(\tau,i)\in[t-1]\times[H]} \cup \{(s_i^t, a_i^t)\}_{i\in[h]} \cup \{s_{h+1}^t\}\big)\,, \end{aligned} \tag{14}$$

where we identify $\mathcal{F}_{t,0,2}$ with $\mathcal{M}_{t-1,H,2}$ for all t $\geqslant 2$ and let $\mathcal{M}_{1,0,2}$ be the empty set. Further, for any $t \in [T]$, $h \in [H]$ and $m \in [2]$, we define the time-step index $\tau(t,h,m)$ as

$$\tau(t,h,m) = (t-1)\cdot 2H + (h-1)\cdot 2 + m\,, \tag{15}$$

which offers an partial ordering over the triplets $(t,h,m) \in [T] \times [H] \times [2]$. Moreover, according to Eq. (14)*, for any (t,h,m) and (t',h',m') satisfying $\tau(k,h,m) \leqslant \tau(k',h',m')$, it holds that $\mathcal{M}_{k,h,m} \subseteq \mathcal{M}_{k',h',m'}$. Thus, the sequence of σ-algebras $\{\mathcal{M}_{t,h,m}\}_{(t,h,m)\in[T]\times[H]\times[2]}$ forms a filtration.*

Accordingly, we have the following regret decomposition result.

Lemma 1 (Regret Decomposition [65, 12])**.** *Recall the definition of the temporal-difference error $\Gamma_h^t \colon \mathcal{S} \times \mathcal{A} \to$ in Eq.* (10) *for all $(t,h) \in [T] \times [H]$, then the regret can be decomposed as*

$$\begin{aligned} \textit{Regret}(T) \leqslant & \underbrace{\sum_{t=1}^T \sum_{h=1}^H \big[\mathbb{E}_{\pi^\star}[\Gamma_h^t(s_h, a_h) \,|\, s_1 = s_1^t] - \Gamma_h^t(s_h^t, a_h^t)\big]}_{\texttt{Term (i)}} + \underbrace{\sum_{t=1}^T \sum_{h=1}^H (\zeta_{t,h}^1 + \zeta_{t,h}^2)}_{\texttt{Term (ii)}} \\ & + \underbrace{\sum_{t=1}^T \sum_{h=1}^H \mathbb{E}_{\pi^\star}\big[\langle Q_h^t(s_h, \cdot), \pi_h^\star(\cdot \,|\, s_h) - \pi_h^t(\cdot|s_h)\rangle_{\mathcal{A}} \big| s_1 = s_1^t\big]}_{\texttt{Term (iii)}\leqslant 0} + \epsilon HT\,, \end{aligned} \tag{16}$$

where $\zeta_{t,h}^1$ and $\zeta_{t,h}^2$ are defined in Eq. (13)*.*

Proof. Remember the definition of the regret, $\tilde{\pi}^t$ is the ϵ-greedy policy and π^t is the greedy policy at the t-th episode, and then we have

$$\begin{aligned} \text{Regret}(T) &= \sum_{t=1}^T \big[V_1^\star(s_1^t) - V_1^{\pi^t}(s_1^t)\big] + \sum_{t=1}^T \big[V_1^{\pi^t}(s_1^t) - V_1^{\tilde{\pi}^t}(s_1^t)\big] \\ &\leqslant \sum_{t=1}^T \underbrace{V_1^\star(s_1^t) - V_1^t(s_1^t)}_{(*)} + \sum_{t=1}^T \underbrace{V_1^t(s_1^t) - V_1^{\pi^t}(s_1^t)}_{(**)} + \epsilon HT\,, \end{aligned} \tag{17}$$

where the first term (*) can be bounded by [65, 12]

$$V_1^\star(s_1^t) - V_1^t(s_1^t) = \sum_{h=1}^H \left[\mathbb{E}_{\pi^\star}[\Gamma_h^t(s_h, a_h) \,|\, s_1 = s_1^t]\right.$$
$$\underbrace{+ \sum_{h=1}^H \mathbb{E}_{\pi^\star}\left[\langle Q_h^t(s_h, \cdot), \pi_h^\star(\cdot \,|\, s_h) - \pi_h^t(\cdot|s_h)\rangle_{\mathcal{A}} \big| s_1 = s_1^t\right]}_{\leqslant 0}, \quad \forall t \in [T],$$

where we use the fact that π^t is the greedy policy with respect to Q_h^t for any $(t, h) \in [T] \times [H]$. The second term (**) is also bounded by [65, 12]

$$V_1^t(s_1^t) - V_1^{\pi^t}(s_1^t) = \sum_{h=1}^H (\zeta_{t,h}^1 + \zeta_{t,h}^2) - \sum_{h=1}^H \Gamma_h^t(s_h^t, a_h^t), \quad \forall t \in [T].$$

Finally, we conclude the proof. □

In the next, it is natural to employ Azuma-Hoeffding inequality for martingale difference sequences as below.

Lemma 2. *[65, statistical error] For $\zeta_{t,h}^1$ and $\zeta_{t,h}^2$ defined in Eq.* (13) *and for any $\delta \in (0, 1)$, with probability at least $1 - \delta$, we have*

$$\sum_{t=1}^T \sum_{h=1}^H (\zeta_{t,h}^1 + \zeta_{t,h}^2) \lesssim \sqrt{TH^3 \log(2/\delta)}.$$

D Proofs of transformation on the temporal difference error

In this section, we aim to transform the temporal difference error in $\texttt{Term (i)}$ to generalization bounds. This is the key part in our proof without bonus function design.

D.1 TD error under the averaged measure

Here we build the connection between $\texttt{Term (i)}$ in the regret decomposition and the TD error Γ_h^t in the $L^2(\mathrm{d}\bar{\mu}_h^{\tilde{t}})$-integrable space.

To this end, we need study the relationship between $L^2(\mathrm{d}\mu)$-norm and L^∞-norm, where μ can be any probability measure over $\mathcal{S} \times \mathcal{A}$. For any $f \in L^2(\mathrm{d}\mu)$ with $\delta \leqslant \|f\|_\infty$, denote

$$\mathcal{G}_\delta := \{(s, a) : |f(s, a)| \geqslant \|f\|_\infty - \delta\}, \ \forall (s, a) \in \mathcal{S} \times \mathcal{A}, \tag{18}$$

then we have the following lemma that $\bar{\mu}_h^{\tilde{t}}(\mathcal{G}_\delta)$ can be lower bounded under the ϵ-greedy policy.

Lemma 3. *Under the ϵ-greedy policy, considering the set in Eq.* (18) *and the averaged measure $\bar{\mu}_h^{\tilde{t}}$ based on a mini-batch of $\tilde{t}$ historical state-action pairs, we have*

$$\bar{\mu}_h^{\tilde{t}}(\mathcal{G}_\delta) \geqslant \Omega\left(\left(\frac{\epsilon}{A}\right)^H\right), \ \forall \epsilon \in (0, 1) \ \textit{and} \ \delta \geqslant 0.$$

Remark: Clearly, in the best case, we have $\bar{\mu}_h^{\tilde{t}}(\mathcal{G}_\delta) \geqslant \Omega\left(\frac{\epsilon}{A}\right)$. Accordingly, we denote $K \in [1, H]$ as a MDP-dependent constant to describe the "myopic" level of MDPs [49] such that $\bar{\mu}_h^{\tilde{t}}(\mathcal{G}_\delta) \geqslant \Omega\left((\epsilon/A)^K\right)$.

Proof. For any $f \in L^2(\mathrm{d}\mu)$ with $\delta \leqslant \|f\|_\infty$, we have

$$\|f\|_{L^2(\mathrm{d}\mu)} \geqslant \left(\int_{\mathcal{G}_\delta} (\|f\|_\infty - \delta)^2 \mathrm{d}\mu\right)^{1/2} = (\|f\|_\infty - \delta)[\mu(\mathcal{G}_\delta)]^{1/2}, \tag{19}$$

which is also valid to $\bar{\mu}_h^{\tilde{t}}$. Clearly, $\bar{\mu}_h^{\tilde{t}}(\mathcal{G}_\delta) \in [0, 1]$.

To prove $\bar{\mu}_h^{\tilde{t}}(\mathcal{G}_\delta) > 0$ with the lower bound, we consider **the worst case** with $\delta = 0$ and every time step taking non-greedy action with probability ϵ. That means, we need to find the optimal state-action pair in Eq. (18), which can be achieved by the fact that all state-action pairs have been visited at all time steps. It is clear that the cardinality of $\mathcal{G}_\delta$ is a non-decreasing function of δ. Accordingly, there exists $j \in [\tilde{t}]$ such that

$$\bar{\mu}_h^{\tilde{t}}(\mathcal{G}_\delta) \geqslant \bar{\mu}_h^{\tilde{t}}(\mathcal{G}_0) \geqslant \min_{(s,a,h)} \mu_h^{\pi_h^{\tau_j}} (s_h^{\tau_j}, a_h^{\tau_j})\,,$$

where $\mu_h^{\pi_h^{\tau_j}}$ is the occupancy measure of the policy $\pi_h^{\tau_j}$ at the h-step and t-th episode. Accordingly, $\mu_h^{\pi_h^{\tau_j}}$ admits the following representation

$$\mu_h^{\pi_h^{\tau_j}} \left(s_h^{\tau_j}, a_h^{\tau_j}\right) = \sum_{s_1^{\tau_j},\dots,s_{h-1}^{\tau_j}} \left(\prod_{i=1}^{h-1} \sum_{a \in \mathcal{A}} \Pr\left(\pi_h^{\tau}\left(s_i^{\tau_j}\right) = a\right) \mathbb{P}_i\left(s_{i+1}^{\tau_j} \mid s_i^{\tau_j}, a\right) \right) \Pr\left(\pi_h^{\tau_j}\left(s_h^{\tau_j}\right) = a_h^{\tau_j}\right)\,.$$

Accordingly, in the worst case, at every time step we take any one action with probability ϵ/A such that

$$\mu_h^{\pi_h^{\tau_j}} \left(s_h^{\tau_j}, a_h^{\tau_j}\right) \geqslant \Omega\left(\left(\frac{\epsilon}{A}\right)^h\right) \geqslant \Omega\left(\left(\frac{\epsilon}{A}\right)^H\right)\,,$$

which implies that

$$\bar{\mu}_h^{\tilde{t}}(\mathcal{G}_\delta) \geqslant \Omega\left(\left(\frac{\epsilon}{A}\right)^H\right)\,,$$

and accordingly we conclude the proof. □

Lemma 4. *Given a MDP-dependent constant $K \in [1, H]$, for the temporal-difference error Γ_h^t defined in Eq.* (10) *for all $(t,h) \in [T] \times [H]$, under the ϵ-greedy policy, then* $\texttt{Term (i)}$ *can be upper bounded by*

$$\texttt{Term (i)} \lesssim \left(\frac{\epsilon}{A}\right)^{-\frac{K}{2}} \sqrt{T} \sum_{h=1}^{H} \sqrt{\sum_{t=1}^{T} \|\Gamma_h^t\|^2_{L^2(\mathrm{d}\bar{\mu}_h^{\tilde{t}})}} + \mathcal{O}(H\sqrt{T})\,.$$

Proof. According to the definition of $\texttt{Term (i)}$ in Lemma 1, we have

$$\begin{aligned}
\texttt{Term (i)} &\leqslant \sum_{t=1}^{T}\sum_{h=1}^{H} \left(\mathbb{E}_{\pi^\star}\left[\left|\Gamma_h^t(s_h, a_h)\right| \middle| s_1 = s_1^t \right] + \left|\Gamma_h^t(s_h^t, a_h^t)\right| \right) \\
&\leqslant 2\sum_{t=1}^{T}\sum_{h=1}^{H} \|\Gamma_h^t\|_\infty \quad \text{[hold for any } (s,a) \in \mathcal{S} \times \mathcal{A}] \\
&\leqslant 2\sum_{t=1}^{T}\sum_{h=1}^{H} \left(\frac{\|\Gamma_h^t\|_{L^2(\mathrm{d}\bar{\mu}_h^{\tilde{t}})}}{\sqrt{\bar{\mu}_h^{\tilde{t}}(\mathcal{G}_\delta)}} + \delta \right) \quad \text{[taking } \mu := \bar{\mu}_h^{\tilde{t}} \text{ in Eq. (19)]}\,.
\end{aligned} \tag{20}$$

Furthermore, by taking $\delta := t^{-1/2}$ such that $\int_1^T t^{-1/2}\mathrm{d}t = \mathcal{O}(\sqrt{T})$, and using $\bar{\mu}_h^{\tilde{t}}(\mathcal{G}_\delta) \geqslant \Omega\left((\epsilon/A)^K\right)$ with $K \in [1, H]$ in Lemma 3, the above equation can be further expressed as

$$\begin{aligned}
\texttt{Term (i)} &\lesssim \left(\frac{\epsilon}{A}\right)^{-\frac{K}{2}} \sum_{t=1}^{T}\sum_{h=1}^{H} \|\Gamma_h^t\|_{L^2(\mathrm{d}\bar{\mu}_h^{\tilde{t}})} + \mathcal{O}(H\sqrt{T}) \quad \text{[using Lemma 3]} \\
&\leqslant \left(\frac{\epsilon}{A}\right)^{-\frac{K}{2}} \sum_{h=1}^{H} \sqrt{T} \sqrt{\sum_{t=1}^{T} \|\Gamma_h^t\|^2_{L^2(\mathrm{d}\bar{\mu}_h^{\tilde{t}})}} + \mathcal{O}(H\sqrt{T})\,, \quad \text{[using elementary inequality]}
\end{aligned}$$

which concludes the proof. □

D.2 Connection between the TD error and generalization bounds

Based on Lemma 4, the key issue left is to bound $\sum_{t=1}^{T} \|\Gamma_h^t\|^2_{L^2(\mathrm{d}\bar{\mu}_h^{\tilde{t}})} \lesssim o(T)$ for a sublinear regret. To this end, we build the connection between $\|\Gamma_h^t\|^2_{L^2(\mathrm{d}\bar{\mu}_h^{\tilde{t}})}$ and generalization bounds. We first the study the decomposition of $\mathcal{E}_h^t(f)$ in Eq. (6) by the following proposition: there exists an extra variance term in the expected risk $\mathcal{E}_h^t(f)$.

Proposition 1. *According to the definition of $\mathcal{E}_h^t(f)$ in Eq.* (6)*, then we have*

$$\mathcal{E}_h^t(f) = \underbrace{\|f - \mathbb{T}_h V_{h+1}^t\|^2_{L^2(\mathrm{d}\bar{\mu}_h^{\tilde{t}})}}_{:=\bar{\mathcal{E}}_h^t(f)} + \mathbb{E}_{(s_h,a_h)\sim\bar{\mu}_h^{\tilde{t}}, s_{h+1}\sim\mathbb{P}_h(\cdot|s_h,a_h)} \mathtt{Var}[V_{h+1}^t(s_{h+1})] \,, \tag{21}$$

where the variance $\mathtt{Var}[V_{h+1}^t(s_{h+1})] := \big[\mathbb{E}_{s_{h+1}}[V_{h+1}^t(s_{h+1})] - V_{h+1}^t(s_{h+1})\big]^2$.

Proof. Denote $s' := s_{h+1}$ for short, we expand $\mathcal{E}_h^t(f)$ as the following expression

$$\begin{aligned}
\mathcal{E}_h^t(f) &= \mathbb{E}_{(s_h,a_h)\sim\bar{\mu}_h^{\tilde{t}}, s'} \left[f(s_h,a_h) - r_h(s_h,a_h) - \mathbb{E}_{s'} V_{h+1}^t(s') + \mathbb{E}_{s'} V_{h+1}^t(s') - V_{h+1}^t(s')\right]^2 \\
&= \mathbb{E}_{(s_h,a_h)\sim\bar{\mu}_h^{\tilde{t}}} \left[f(s_h,a_h) - r_h(s_h,a_h) - \mathbb{E}_{s'} V_{h+1}^t(s')\right]^2 + \mathbb{E}_{(s_h,a_h)\sim\bar{\mu}_h^{\tilde{t}}, s'} \mathtt{Var}[V_{h+1}^t(s')] \\
&= \|f - \mathbb{T}_h V_{h+1}^t\|^2_{L^2(\mathrm{d}\bar{\mu}_h^{\tilde{t}})} + \mathbb{E}_{(s_h,a_h)\sim\bar{\mu}_h^{\tilde{t}}, s_{h+1}\sim\mathbb{P}_h(\cdot|s_h,a_h)} \mathtt{Var}[V_{h+1}^t(s_{h+1})] \,,
\end{aligned}$$

where we use $\mathbb{E}_{s'\sim\mathbb{P}_h(\cdot|s_h,a_h)} \left[\mathbb{E}_{s'}[V_{h+1}^t(s')] - V_{h+1}^t(s')\right] = 0$ and conclude the proof. □

According to the decomposition of $\mathcal{E}_h^t(f)$ Proposition 1, $\bar{\mathcal{E}}_h^t(f)$ in Eq. (21) is close to the squared Bellman error [61]. We are able to transform the estimation of the TD error to generalization error and approximation error as below.

Lemma 5. *For the temporal-difference error Γ_h^t defined in Eq.* (10) *for all $(t,h) \in [T] \times [H]$, it can be upper bounded in the $L^2(\mathrm{d}\mu_h^{\tilde{t}})$ space with*

$$\|\Gamma_h^t\|^2_{L^2(\mathrm{d}\bar{\mu}_h^{\tilde{t}})} \leqslant \left[\mathcal{E}_h^t(\widehat{Q}_h^t) - \min_{f\in\mathcal{F}} \mathcal{E}_h^t(f)\right] + \inf_{f\in\mathcal{F}} \|f - \mathbb{T}_h^\star Q_{h+1}^t\|^2_{L^2(\mathrm{d}\bar{\mu}_h^{\tilde{t}})} \,,$$

where the first term is the generalization error of $\widehat{Q}_h^t$, the second term is the approximation error in the function class $\mathcal{F}$.

Proof. According to the definition of the TD error Γ_h^t and taking $f := \widehat{Q}_h^t$ in Eq. (21) given by Proposition 1, we have

$$\begin{aligned}
\mathcal{E}_h^t(\widehat{Q}_h^t) &= \|\widehat{Q}_h^t - \mathbb{T}_h V_{h+1}^t\|^2_{L^2(\mathrm{d}\bar{\mu}_h^{\tilde{t}})} + \mathbb{E}_{(s_h,a_h)\sim\bar{\mu}_h^{\tilde{t}}, s_{h+1}\sim\mathbb{P}_h(\cdot|s_h,a_h)} \mathtt{Var}[V_{h+1}^t(s_{h+1})] \\
&= \frac{1}{\tilde{t}} \sum_{j=1}^{\tilde{t}} \|\widehat{Q}_h^t - \mathbb{T}_h V_{h+1}^t\|^2_{L^2(\mathrm{d}\mu_h^{\tau_j})} + \mathbb{E}_{(s_h,a_h)\sim\bar{\mu}_h^{\tilde{t}}, s_{h+1}\sim\mathbb{P}_h(\cdot|s_h,a_h)} \mathtt{Var}[V_{h+1}^t(s_{h+1})] \\
&= \frac{1}{\tilde{t}} \sum_{j=1}^{\tilde{t}} \|\Gamma_h^t\|^2_{L^2(\mathrm{d}\mu_h^{\tau_j})} + \mathbb{E}_{(s_h,a_h)\sim\bar{\mu}_h^{\tilde{t}}, s_{h+1}\sim\mathbb{P}_h(\cdot|s_h,a_h)} \mathtt{Var}[V_{h+1}^t(s_{h+1})] \,,
\end{aligned} \tag{22}$$

where the second equality holds by the definition of the averaged measure $\bar{\mu}_h^{\tilde{t}} = \frac{1}{\tilde{t}} \sum_{j=1}^{\tilde{t}} \mu_h^{\tau_j}$; and we use $Q_h^t = \widehat{Q}_h^t$ in the last equality as the truncation operation has been given in function classes, see Eqs. (3) and (4). Then, taking the infimum on both sides of Eq. (21), we have

$$\begin{aligned}
\min_{f\in\mathcal{F}} \mathcal{E}_h^t(f) &= \inf_{f\in\mathcal{F}} \|f - \mathbb{T}_h V_{h+1}^t\|^2_{L^2(\mathrm{d}\bar{\mu}_h^{\tilde{t}})} + \mathbb{E}_{(s_h,a_h)\sim\bar{\mu}_h^{\tilde{t}}, s_{h+1}\sim\mathbb{P}_h(\cdot|s_h,a_h)} \mathtt{Var}[V_{h+1}^t(s_{h+1})] \\
&= \inf_{f\in\mathcal{F}} \|f - \mathbb{T}_h^\star Q_{h+1}^t\|^2_{L^2(\mathrm{d}\bar{\mu}_h^{\tilde{t}})} + \mathbb{E}_{(s_h,a_h)\sim\bar{\mu}_h^{\tilde{t}}, s_{h+1}\sim\mathbb{P}_h(\cdot|s_h,a_h)} \mathtt{Var}[V_{h+1}^t(s_{h+1})] \,,
\end{aligned} \tag{23}$$

where the second equality holds by $V_{h+1}^t(s_{h+1}) = \max_{a\in\mathcal{A}} Q_{h+1}^t(s_{h+1}, a)$.

Combining Eqs. (22) and (23), we have

$$\|\Gamma_h^t\|^2_{L^2(\mathrm{d}\bar{\mu}_h^{\tilde{t}})} = \mathcal{E}_h^t(\widehat{Q}_h^t) - \min_{f\in\mathcal{F}} \mathcal{E}_h^t(f) + \inf_{f\in\mathcal{F}} \|f - \mathbb{T}_h^\star Q_{h+1}^t\|^2_{L^2(\mathrm{d}\bar{\mu}_h^{\tilde{t}})} \,, \tag{24}$$

which concludes the proof. □

Based on Lemma 5, we have the following corollary if we consider the approximation error in $L^p(\mathcal{X})$-integrable space, which is needed for our results on deep ReLU neural networks.

Corollary 1. *Under the same setting of Lemma 5, we have*

$$\|\Gamma_h^t\|^2_{L^2(\mathrm{d}\bar{\mu}_h^{\tilde{t}})} \lesssim \left[\mathcal{E}_h^t(\widehat{Q}_h^t) - \min_{f\in\mathcal{F}} \mathcal{E}_h^t(f)\right] + \inf_{f\in\mathcal{F}} \|f - \mathbb{T}_h^\star Q_{h+1}^t\|^2_{L^4(\mathcal{X})} \,.$$

Proof. Following the proof of Lemma 5, this result can be easily obtained by Cauchy-Schwartz inequality. To be specific, for any probability measure μ, we have

$$\|f\|_{L^p(\mathrm{d}\mu)} \leqslant \|f\|_{L^{2p}(\mathcal{X})} \left(\int_{\mathcal{X}} |g(\boldsymbol{x})|^2 \mathrm{d}\boldsymbol{x}\right)^{\frac{1}{2p}} \lesssim \|f\|_{L^{2p}(\mathcal{X})} \,,$$

where g is the probability density function associated with the probability measure μ. Note that the result here still holds true for the approximation error in $L^\infty(\mathcal{X})$ if we use Hölder inequality, but this condition is much stronger as it requires the target Q function to be continuous. □

E Generalization bounds on non-iid data

In this section, we prove that the traditional Rademacher complexity is still valid for independent but non-identically distributed data under a well-defined measure. Similarly, such result is also valid to local Rademacher complexity. The key fact is that, the classical Rademacher complexity [71] is still valid as McDiarmid's bound only requires the independent property.

For description simplicity, we consider a general setting beyond our reinforcement learning task, i.e., learning with n independent but non-identical distributed data $X = \{\boldsymbol{x}_i\}_{i=1}^n$ in $\mathbb{R}^d$ with $\boldsymbol{x}_i \sim \mu_i, \forall i \in [n]$. Define the average measure $\bar{\mu} := \frac{1}{n}\sum_{i=1}^n \mu_i$, we have

$$\mathbb{E}_{\boldsymbol{x}\sim\bar{\mu}}[f(\boldsymbol{x})] = \frac{1}{n}\sum_{i=1}^n \int_{\mathbb{R}^d} f(\boldsymbol{x})\mathrm{d}\mu_i(\boldsymbol{x}) = \frac{1}{n}\sum_{i=1}^n \mathbb{E}_{\boldsymbol{x}\sim\mu_i}[f(\boldsymbol{x})] \,. \tag{25}$$

Accordingly, the *empirical Rademacher complexity* of a function class $\mathcal{F}$ on the sample set X is defined as

$$\widehat{\mathcal{R}}_n(\mathcal{F}, X) = \frac{1}{n}\mathbb{E}_{\boldsymbol{\xi}}\left[\sup_{f\in\mathcal{F}} \sum_{i=1}^n \xi_i f(\boldsymbol{x}_i)\right] \,, \tag{26}$$

where the expectation is taken over $\boldsymbol{\xi} = \{\xi_1, \xi_2, \cdots, \xi_n\}$, i.e., Rademacher random variables, with $\Pr(\xi_i = 1) = \Pr(\xi_i = -1) = 1/2$. The related *Rademacher complexity* under our non-iid setting is defined as

$$\mathcal{R}_n(\mathcal{F}) = \mathbb{E}_{\boldsymbol{x}_1\sim\mu_1,\cdots,\boldsymbol{x}_n\sim\mu_n}\left[\frac{1}{n}\mathbb{E}_{\boldsymbol{\xi}}\left[\sup_{f\in\mathcal{F}} \sum_{i=1}^n \xi_i f\left(\boldsymbol{x}_i\right)\right]\right] \,,$$

where the expectation is taken over $\{\boldsymbol{x}_i\}_{i=1}^n$ with respect to each probability measure $\{\mu_i\}_{i=1}^n$. This definition follows the classical *Rademacher complexity* [71] on iid samples to intuitively indicates how expressive the function class is. Besides, in our proof, we also need a notation of *local Rademacher complexity* on a set of vectors, where "local" means that the class over which the Rademacher process is defined is a subset of the original class. Following the same style with Rademacher complexity, the local Rademacher complexity under the non-iid setting is defined as $\mathcal{R}_n\{f \in \mathcal{F} : \mathbb{E}_{\bar{\mu}} f^2 \leqslant R\}$, and the empirical local Rademacher complexity is defined as $\widehat{\mathcal{R}}_n\{f \in \mathcal{F} : P_n f^2 \leqslant R\}$, where we denote $P_n f := \frac{1}{n}\sum_{i=1}^n f(\boldsymbol{x}_i)$ for short.

Besides, Rademacher complexity is also related to covering number, a metric for estimation of a hypothesis space. Here we give the definition of covering number, that is also used in this work.

Definition 4. *[72, Definition 5.1, covering number] Let $(\mathcal{F}, \|\cdot\|)$ be a norm space. A δ-cover of the set $\mathcal{F}$ with respect to $\|\cdot\|$ is a set $\{\theta_1, \cdots, \theta_n\} \subseteq \mathcal{F}$ such that for each $\theta \in \mathcal{F}$, there exists some $i \in [n]$ such that $\|\theta - \theta_i\| \leqslant \delta$. The δ-covering number $\mathscr{N}(\delta, \mathcal{F}, \|\cdot\|)$ is the cardinality of the minimal δ-cover.*

In this work, we consider the covering number with two types of norms, one is $\mathscr{N}(\epsilon, \mathcal{F}, \|\cdot\|_\infty)$ and the other is $\mathscr{N}(\epsilon, \mathcal{F}, \|\cdot\|_2) := \sup_n \sup_{P_n} \mathscr{N}(\epsilon, \mathcal{F}, \|\cdot\|_{L_2(P_n)})$ [73].

E.1 Rademacher complexity on non-iid data

Based on the definition of Rademacher complexity and its empirical version, we have the following lemma.

Lemma 6. *Let $X = \{\boldsymbol{x}_i\}_{i=1}^n$ be an independent but non-identical distributed data set with $\boldsymbol{x}_i \sim \mu_i, \forall i \in [n]$, and $R_n(\mathcal{F})$ be the Rademacher complexity of the function class $\mathcal{F}$ on X, denote the averaged probability measure as $\bar{\mu} := \frac{1}{n}\sum_{i=1}^n \mu_i$, then we have*

$$\mathbb{E}_{\boldsymbol{x}_1\sim\mu_1,\cdots,\boldsymbol{x}_n\sim\mu_n}\left[\sup_{f\in\mathcal{F}}\left(\mathbb{E}_{\boldsymbol{x}\sim\bar{\mu}}[f(\boldsymbol{x})] - \frac{1}{n}\sum_{i=1}^n f(\boldsymbol{x}_i)\right)\right] \leq 2\mathcal{R}_n(\mathcal{F})\,.$$

Proof. The proof follows with the classical Rademacher complexity [74, Chapter 26] apart from the averaged measure. Take a copy of X, i.e., $X' = \{\boldsymbol{x}'_i\}_{i=1}^n$ such that X' is independent but $\boldsymbol{x}'_i \sim \mu_i, \forall i \in [n]$. According to Eq. (25), we have

$$\mathbb{E}_{\boldsymbol{x}\sim\bar{\mu}}[f(\boldsymbol{x})] = \mathbb{E}_{\boldsymbol{x}'_1\sim\mu_1,\cdots,\boldsymbol{x}'_n\sim\mu_n}\left[\frac{1}{n}\sum_{i=1}^n f(\boldsymbol{x}'_i)\right]\,. \tag{27}$$

Note that every possible configuration/value of $\boldsymbol{\xi}$ has probability of $1/2^n$ due to $\boldsymbol{\xi} \in \{-1,1\}^n$. Without loss of generality, we can permute any configuration of $\boldsymbol{\xi}$ of such that

$$\xi_{u_1} = \xi_{u_2} = \cdots = \xi_{u_k} = 1, \quad \xi_{u_{k+1}} = \xi_{u_{k+2}} = \cdots = \xi_{u_n} = -1,\;\; k \in \{0\}\cup[n]\,,$$

where $\boldsymbol{u} = \{u_1, u_2, \cdots, u_n\}$ is a permutation of $\{1, 2, \ldots, n\}$. Accordingly, for any configuration of $\boldsymbol{\xi}$, we have

$$\begin{aligned}
&\mathbb{E}_{\{\boldsymbol{x}_i\}_{i=1}^n}\left[\mathbb{E}_{\{\boldsymbol{x}'_i\}_{i=1}^n}\left[\sup_{f\in\mathcal{F}}\left(\frac{1}{n}\sum_{i=1}^n \xi_i\left(f\left(\boldsymbol{x}'_i\right) - f\left(\boldsymbol{x}_i\right)\right)\right)\right]\right] \\
&\quad = \mathbb{E}_{\{\boldsymbol{x}_i\}_{i=1}^n}\left[\mathbb{E}_{\{\boldsymbol{x}'_i\}_{i=1}^n}\left[\sup_{f\in\mathcal{F}}\left(\frac{1}{n}\left(\sum_{i=1}^k\left(f\left(\boldsymbol{x}'_{u_i}\right) - f\left(\boldsymbol{x}_{u_i}\right)\right) + \sum_{i=k+1}^n\left(f\left(\boldsymbol{x}_{u_i}\right) - f\left(\boldsymbol{x}'_{u_i}\right)\right)\right)\right)\right]\right] \\
&\quad = \mathbb{E}_{\{\boldsymbol{x}_i\}_{i=1}^n}\left[\mathbb{E}_{\{\boldsymbol{x}'_i\}_{i=1}^n}\left[\sup_{f\in\mathcal{F}}\left(\frac{1}{n}\sum_{i=1}^n\left(f\left(\boldsymbol{x}'_i\right) - f\left(\boldsymbol{x}_i\right)\right)\right)\right]\right]\,,
\end{aligned} \tag{28}$$

where we use the fact that $\boldsymbol{x}_{u_i}$ and $\boldsymbol{x}'_{u_i}$ are independent and symmetric. Based on this, we obtain

$$
\begin{aligned}
&\mathbb{E}_{\{\boldsymbol{x}_i\}_{i=1}^n}\left[\sup_{f\in\mathcal{F}}\left(\mathbb{E}_{\boldsymbol{x}\sim\bar{\mu}}[f(\boldsymbol{x})]-\frac{1}{n}\sum_{i=1}^n f\left(\boldsymbol{x}_i\right)\right)\right]\\
&\quad=\mathbb{E}_{\{\boldsymbol{x}_i\}_{i=1}^n}\left[\sup_{f\in\mathcal{F}}\left(\mathbb{E}_{\{\boldsymbol{x}'_i\}_{i=1}^n}\left[\frac{1}{n}\sum_{i=1}^n f\left(\boldsymbol{x}'_i\right)\right]-\frac{1}{n}\sum_{i=1}^n f\left(\boldsymbol{x}_i\right)\right)\right] \qquad \text{[using Eq. (27)]}\\
&\quad=\mathbb{E}_{\{\boldsymbol{x}_i\}_{i=1}^n}\left[\sup_{f\in\mathcal{F}}\left(\mathbb{E}_{\{\boldsymbol{x}'_i\}_{i=1}^n}\left[\frac{1}{n}\sum_{i=1}^n f\left(\boldsymbol{x}'_i\right)-\frac{1}{n}\sum_{i=1}^n f\left(\boldsymbol{x}_i\right)\right]\right)\right]\\
&\quad\leqslant\mathbb{E}_{\{\boldsymbol{x}_i\}_{i=1}^n}\left[\mathbb{E}_{\{\boldsymbol{x}'_i\}_{i=1}^n}\left[\sup_{f\in\mathcal{F}}\left(\frac{1}{n}\sum_{i=1}^n f\left(\boldsymbol{x}'_i\right)-\frac{1}{n}\sum_{i=1}^n f\left(\boldsymbol{x}_i\right)\right)\right]\right] \qquad \text{[Jensen's inequality]}\\
&\quad=\mathbb{E}_{\{\boldsymbol{x}_i\}_{i=1}^n}\left[\mathbb{E}_{\{\boldsymbol{x}'_i\}_{i=1}^n}\left[\mathbb{E}_{\boldsymbol{\xi}}\left[\sup_{f\in\mathcal{F}}\left(\frac{1}{n}\sum_{i=1}^n \xi_i\left(f\left(\boldsymbol{x}'_i\right)-f\left(\boldsymbol{x}_i\right)\right)\right)\right]\right]\right] \qquad \text{[using Eq. (28)]}\\
&\quad\leqslant\mathbb{E}_{\{\boldsymbol{x}_i\}_{i=1}^n}\left[\mathbb{E}_{\boldsymbol{\xi}}\left[\sup_{f\in\mathcal{F}}\left(\frac{1}{n}\sum_{i=1}^n \xi_i f\left(\boldsymbol{x}_i\right)\right)\right]\right]+\mathbb{E}_{\{\boldsymbol{x}'_i\}_{i=1}^n}\left[\mathbb{E}_{\boldsymbol{\xi}}\left[\sup_{f\in\mathcal{F}}\left(\frac{1}{n}\sum_{i=1}^n \xi_i f\left(\boldsymbol{x}'_i\right)\right)\right]\right]\\
&\quad=2\mathcal{R}_n(\mathcal{F})\,,
\end{aligned}
\tag{29}
$$

where the last inequality holds by the fact that ξ_i and $-\xi_i,\ i\in[n]$ admit the same distribution, and multiplying each term in the summation by a Rademacher variable ξ_i will not change the expectation due to $\mathbb{E}\xi_i=0$.

□

Based on the above lemma, we demonstrate that the Rademacher complexity can be well approximated by the empirical Rademacher complexity under our non-iid setting.

Lemma 7. *Under the same setting of Lemma 6, for any* $f\in\mathcal{F}$*, assume that* $|f(\boldsymbol{x})-f(\boldsymbol{x}')|\leqslant c,\ \forall\boldsymbol{x},\boldsymbol{x}'\in\operatorname{dom}(f)$ *for some constant* $c>0$*, for any* $\delta\in(0,1)$*, the following proposition holds with probability at least* $1-\delta$

$$
\Pr\left(\mathbb{E}_{\boldsymbol{x}\sim\bar{\mu}}(f(\boldsymbol{x}))-\frac{1}{n}\sum_{i=1}^n f\left(\boldsymbol{x}_i\right)\geqslant 2\widehat{\mathcal{R}}_n(\mathcal{F},X)+3\delta\right)\leqslant 2\exp\left(-\frac{2n\delta^2}{c^2}\right)\,. \tag{30}
$$

Proof. The proof follows with the classical Rademacher complexity [74, Chapter 26] apart from the averaged measure. Recall the definition of the empirical Rademacher complexity in Eq. (26), $\widehat{\mathcal{R}}_n(\mathcal{F},X)$ is a function of n random variables $\{\boldsymbol{x}_i\}_{i=1}^n$. Moreover, due to $|f(\boldsymbol{x})-f(\boldsymbol{x}')|\leqslant c$, $\widehat{\mathcal{R}}_n(\mathcal{F},X)$ satisfies the precondition for McDiarmid's inequality by at most c/n, which only requires independence of random variables without the identically distributed condition

$$
\Pr\left(\widehat{\mathcal{R}}_n(\mathcal{F},X)-\mathbb{E}_{\{\boldsymbol{x}_i\}_{i=1}^n}[\widehat{\mathcal{R}}_n(\mathcal{F},X)]\geqslant\delta\right)\leqslant\exp\left(-\frac{2n\delta^2}{c^2}\right)\,,
$$

which implies

$$
\Pr\left(\left|\widehat{\mathcal{R}}_n(\mathcal{F},X)-\mathcal{R}_n(\mathcal{F})\right|\geqslant\delta\right)\leqslant 2\exp\left(-\frac{2n\delta^2}{c^2}\right)\,. \tag{31}
$$

By Lemma 6, we have

$$
\mathbb{E}_{\{\boldsymbol{x}_i\}_{i=1}^n}\left[\mathbb{E}_{\boldsymbol{x}\sim\bar{\mu}}[f(\boldsymbol{x})]-\frac{1}{n}\sum_{i=1}^n f\left(\boldsymbol{x}_i\right)\right]\leqslant\mathbb{E}_{\{\boldsymbol{x}_i\}_{i=1}^n}\left[\sup_{f\in\mathcal{F}}\left(\mathbb{E}_{\boldsymbol{x}\sim\bar{\mu}}[f(\boldsymbol{x})]-\frac{1}{n}\sum_{i=1}^n f\left(\boldsymbol{x}_i\right)\right)\right]\leqslant 2\mathcal{R}_n(\mathcal{F})\,.
$$

Denote event $\mathtt{A}$ as

$$
\left[\mathbb{E}_{\boldsymbol{x}\sim\bar{\mu}}[f(\boldsymbol{x})]-\frac{1}{n}\sum_{i=1}^n f\left(\boldsymbol{x}_i\right)\right]-\mathbb{E}_{\{\boldsymbol{x}_i\}_{i=1}^n}\left[\mathbb{E}_{\boldsymbol{x}\sim\bar{\mu}}[f(\boldsymbol{x})]-\frac{1}{n}\sum_{i=1}^n f\left(\boldsymbol{x}_i\right)\right]\geqslant\delta\,,
$$

we use McDiarmid's inequality again to obtain $\Pr(\mathtt{A}) \leqslant e^{-2n\delta^2/c^2}$ since $\mathbb{E}_{\boldsymbol{x}\sim\bar{\mu}}[f(\boldsymbol{x})] - \frac{1}{n}\sum_{i=1}^{n} f(\boldsymbol{x}_i)$ can be regarded as a function of $\{\boldsymbol{x}_i\}_{i=1}^n$ and any variations of $\{\boldsymbol{x}_i\}_{i=1}^n$ would change the outcome by at most c/n. Denote event $\mathtt{B}$ as $\mathbb{E}_{\boldsymbol{x}\sim\bar{\mu}}[f(\boldsymbol{x})] - \frac{1}{n}\sum_{i=1}^{n} f(\boldsymbol{x}_i) - 2\mathcal{R}_n(\mathcal{F}) \geqslant \delta$, we have $\Pr(\mathtt{B}) \leqslant \Pr(\mathtt{A}) \leqslant e^{-2n\delta^2/c^2}$.

Further, denote the event $\mathtt{C}$ as $\widehat{\mathcal{R}}_n(\mathcal{F}, X) \geqslant \mathcal{R}_n(\mathcal{F}) - \delta$, we have $\Pr(\mathtt{C}) \geqslant 1 - \exp(-2n\delta^2/c^2)$ by Eq. (31). Denote the event $\mathtt{D}$ as $\mathbb{E}_{\boldsymbol{x}\sim\bar{\mu}}(f(\boldsymbol{x})) - \frac{1}{n}\sum_{i=1}^{n} f(\boldsymbol{x}_i) \geqslant 2\widehat{\mathcal{R}}_n(\mathcal{F}) + 3\delta$, we have

$$\begin{aligned}
\Pr\left(\mathbb{E}_{\boldsymbol{z}\sim\bar{\mu}}(f(\boldsymbol{x})) - \frac{1}{n}\sum_{i=1}^{n} f(\boldsymbol{x}_i) \geqslant 2\widehat{\mathcal{R}}_n(\mathcal{F}) + 3\delta\right) &= \Pr(\mathtt{D}) = \Pr(\mathtt{C}\cap\mathtt{D}) + \Pr(\mathtt{C}\cup\mathtt{D}) - \Pr(\mathtt{C}) \\
&\leqslant \Pr(\mathtt{B}) + 1 - \Pr(\mathtt{C}) \\
&= 2\exp\left(-2n\delta^2/c^2\right),
\end{aligned}$$

which concludes the proof.

□

Similar to the proof of Lemma 7, it is easy to verify that, the standard Massart's lemma and the Talagrand's Contraction Lemma (empirical Rademacher complexity of Lipschitz function class) in [74, Chapter 26] are valid to our independent but non-iid setting.

E.2 Local Rademacher complexity

Here we present some results on local Rademacher complexity [75] that is needed in this work. The used lemmas here are still valid for our independent but non-identically distributed data. Since the proof framework is similar to what we present for Rademacher complexity, we omit the proofs here.

When applying local Rademacher complexity, we need the following definition.

Definition 5. *A function $\psi : \mathbb{R}_+ \to \mathbb{R}_+$ is sub-root if it is non-negative, non-decreasing, and if $\psi(x)/\sqrt{x}$ is non-increasing.*

Lemma 8. *[76, Theorem 2] Let $\mathcal{F}$ be a function class with $\|f\|_\infty \leqslant b$, $\forall f \in \mathcal{F}$ and $\widetilde{F} := \{f - g : f, g \in \mathcal{F}\}$, and $P_n f := \frac{1}{n}\sum_{i=1}^{n} f(\boldsymbol{x}_i)$, then taking the average measure $\bar{\mu}$, we have*

$$\begin{aligned}
\mathcal{R}_n\{f \in \mathcal{F} : \mathbb{E}_{\bar{\mu}} f^2 \leqslant R\} \leqslant \inf_{\epsilon>0} \Bigg[& 2\mathcal{R}_n\{f \in \widetilde{\mathcal{F}} : P_n f^2 \leqslant \epsilon^2\} + \frac{8b\log(\epsilon/2, \mathcal{F}, \|\cdot\|_2)}{n} \\
& + \sqrt{\frac{2R\log\mathscr{N}(\epsilon/2, \mathcal{F}, \|\cdot\|_2)}{n}} \Bigg].
\end{aligned}$$

Lemma 9. *[75, Theorem 3.3, modified version] Let f be a class of functions with ranges in $[a, b]$ and assume that there exists some functional $T : \mathcal{F} \to \mathbb{R}^+$ and some constant B such that $\mathtt{Var}(f) \leqslant T(f) \leqslant BPf$ for every $f \in \mathcal{F}$. Let P_n be the empirical measure supported on the independent data points $\{\boldsymbol{x}_i\}_{i=1}^n$ with the averaged measure $\bar{\mu} := \frac{1}{n}\mu_i$, Let ψ be a sub-root function with the fixed point R^*. If for any $R \geqslant R^*$, ψ satisfies*

$$\psi(R) \geqslant B\mathcal{R}_n\{f \in \mathcal{F} : T(f) \leqslant R\},$$

then for any $J > 1$ and $\delta \in (0, 1)$, with probability at least $1 - \delta$, we have

$$\mathbb{E}_{\bar{\mu}} f \leqslant \frac{J}{J-1} P_n f + \frac{c_1 J}{B} R^* + (c_2(b-a) + c_3 BJ)\frac{\log(1/\delta)}{n},$$

where c_1, c_2, c_3 are some positive constants.

Lemma 10. *[73, Refined entropy integral] Let P_n be the empirical measure supported on the independent data points $\{\boldsymbol{x}_i\}_{i=1}^n$. For any function class $\mathcal{F}$ and any monotone sequence $\{\epsilon_k\}_{k=0}^\infty$ decreasing to 0 such that $\epsilon_0 \geqslant \sup_{f\in\mathcal{F}} \sqrt{P_n f^2}$, the following inequality holds for every non-negative integer N*

$$\widehat{\mathcal{R}}_n(\mathcal{F}, X) \leqslant 4\sum_{k=1}^{N} \epsilon_{k-1}\sqrt{\frac{\log\mathscr{N}(\epsilon_k, \mathcal{F}, \|\cdot\|_2)}{n}} + \epsilon_N. \tag{32}$$

F Proofs of regret bounds via deep ReLU neural networks

In this section, we give the proofs of regret bounds via deep ReLU neural networks according to the function class of $\mathbb{T}_h^\star Q$ in Besov spaces.

To conclude our proof, we need the following lemma that how well the functions in the Besov space can be approximated by deep neural networks with the ReLU activation. Here the approximation error is defined in the $L^4(\mathcal{X})$-integrable space (*c.f.* Corollary 1).

Lemma 11. *(Approximation error in Besov space) [30, Proposition 1, modified version] Assume that the smoothness parameter α satisfies*

$$\alpha > \eta := d(1/p - 1/4)_+ \,,$$

then there exists a deep neural network architecture $\mathcal{F}_{\texttt{DNN}}(L,m,S,B)$ with $\nu := (\alpha-\eta)/(2\eta)$ and a large N such that

$$L \asymp \log N, \quad S \asymp N, \quad m \asymp N \log N, \quad \textit{and } B \asymp N^{1/\nu + 1/d}\,, \tag{33}$$

then it holds that

$$\sup_{f^* \in \mathcal{B}^\alpha_{p,q}(\mathcal{X})} \inf_{f \in \mathcal{F}_{\texttt{DNN}}(L,m,S,B)} \|f - f^*\|_{L^4(\mathcal{X})} \lesssim N^{-\frac{\alpha}{d}}\,, \qquad \forall q > 0\,.$$

In our proof, we need the following result on local Rademacher complexity of deep ReLU neural networks.

Lemma 12. *Let $X = \{\boldsymbol{x}_i\}_{i=1}^n \subseteq [0,1]^d$ be an independent but non-identical distributed data set with $\boldsymbol{x}_i \sim \mu_i, \forall i \in [n]$, and $\mathcal{R}_n\{f \in \mathcal{F}_{\texttt{DNN}} : Pf^2 \leqslant R\}$ be the local Rademacher complexity of the function class $\mathcal{F}_{\texttt{DNN}}$ on X defined in Eq.* (4)*, denote the averaged measure as $\bar{\mu} := \frac{1}{n}\sum_{i=1}^n \mu_i$, then for a large N, we have*

$$\mathcal{R}_n\{f \in \mathcal{F}_{\texttt{DNN}} : \mathbb{E}_{\bar{\mu}} f^2 \leqslant R\} \lesssim \left(\frac{1}{n} + \sqrt{\frac{R}{n}}\right)\sqrt{N[(\log N)^2 + \log n]} + \frac{HN[(\log N)^2 + \log n]}{n}\,. \tag{34}$$

Remark: The parameter N depends on the number of the training data n, but it will be determined later.

Proof. According to [30, Lemma 3], the covering number of $\mathcal{F}_{\texttt{DNN}}$ can be bounded by

$$\begin{aligned} \log \mathscr{N}(\varepsilon, \mathcal{F}_{\texttt{DNN}}, \|\cdot\|_2) \leqslant \log \mathscr{N}(\varepsilon, \mathcal{F}_{\texttt{DNN}}, \|\cdot\|_\infty) &\le 2SL \log\left(\frac{L(B \vee 1)(m+1)}{\varepsilon}\right) \\ &\lesssim N\left[(\log N)^2 + \log\left(\frac{1}{\varepsilon}\right)\right]\,. \end{aligned}$$

Denote $\widetilde{\mathcal{F}}_{\texttt{DNN}} = \{f - g : f, g \in \mathcal{F}_{\texttt{DNN}}\}$, it satisfies

$$\begin{aligned} \log \mathscr{N}(\varepsilon, \widetilde{\mathcal{F}}_{\texttt{DNN}}, \|\cdot\|_2) \leqslant 2\log \mathscr{N}\left(\frac{\varepsilon}{2}, \mathcal{F}_{\texttt{DNN}}, \|\cdot\|_2\right) &\leqslant 2\log \mathscr{N}\left(\frac{\varepsilon}{2}, \mathcal{F}_{\texttt{DNN}}, \|\cdot\|_\infty\right) \\ &\lesssim N\left[(\log N)^2 + \log\left(\frac{2}{\varepsilon}\right)\right]\,. \end{aligned} \tag{35}$$

According to Lemma 10, taking $\varepsilon_j = 2^{-j}\varepsilon$, and using the inequality

$$\mathscr{N}(\varepsilon_j, \{f \in \widetilde{\mathcal{F}}_{\texttt{DNN}} : P_n f^2 \leqslant \varepsilon^2\}, \|\cdot\|_2) \leqslant \mathscr{N}(\varepsilon_j/2, \widetilde{\mathcal{F}}_{\texttt{DNN}}, \|\cdot\|_2)\,,$$

then the following inequality holds for any $J \in \mathbb{N}^+$:

$$
\begin{aligned}
\mathcal{R}_n\{f \in \widetilde{\mathcal{F}}_{\texttt{DNN}} : P_n f^2 \leqslant \varepsilon^2\} = \mathbb{E}\widehat{\mathcal{R}}_n\{f \in \widetilde{\mathcal{F}}_{\texttt{DNN}} : P_n f^2 \leqslant \varepsilon^2\} & \\
\leqslant 4\mathbb{E}\sum_{j=1}^{J} \varepsilon_{j-1}\sqrt{\frac{\log \mathscr{N}(\varepsilon_j/2, \widetilde{\mathcal{F}}_{\texttt{DNN}}, \|\cdot\|_2)}{n}} + \varepsilon_J & \\
\leqslant 4\mathbb{E}\sum_{j=1}^{J} 2^{-(j-1)}\varepsilon\sqrt{\frac{2\log \mathscr{N}(\frac{\varepsilon}{2^{(j+1)}}, \widetilde{\mathcal{F}}_{\texttt{DNN}}, \|\cdot\|_\infty)}{n}} + \varepsilon_J & \qquad (36) \\
\lesssim \frac{\varepsilon}{\sqrt{n}}\sum_{j=1}^{J} 2^{-(j-1)}\sqrt{2N\left[(\log N)^2 + \log\left(\frac{2^{j+1}}{\varepsilon}\right)\right]} + 2^{-J}\varepsilon & \\
\lesssim \frac{\varepsilon}{\sqrt{n}}\sqrt{N\left[(\log N)^2 + \log\left(\frac{1}{\varepsilon}\right)\right]}\,, \quad \text{[taking } J \to \infty] &
\end{aligned}
$$

where the first inequality holds by Lemma 10 and the second and third inequalities hold by Eq. (35). The last inequality uses the fact that $\sum_{j=0}^{\infty} \frac{\sqrt{j+1}}{2^{j-1}} < \infty$.

According to Lemma 8 with $\sup_{f \in \mathcal{F}_{\texttt{DNN}}} \|f\|_\infty \leqslant H$, we have

$$
\begin{aligned}
\mathcal{R}_n\{f \in \mathcal{F}_{\texttt{DNN}} : Pf^2 \leqslant R\} \lesssim \inf_{\varepsilon > 0}\Bigg[& 2\mathbb{E}\mathcal{R}_n\{f \in \widetilde{\mathcal{F}}_{\texttt{DNN}} : P_n f^2 \leqslant \varepsilon^2\} \\
& + \frac{8HN\left[(\log N)^2 + \log\left(\frac{1}{\varepsilon}\right)\right]}{n} + \sqrt{\frac{2rN\left[\log^2 N + \log\left(\frac{1}{\varepsilon}\right)\right]}{n}}\Bigg] \\
\lesssim \inf_{\varepsilon > 0} & \left[\frac{\epsilon + \sqrt{2R}}{\sqrt{n}}\sqrt{N\left[(\log N)^2 + \log\left(\frac{1}{\varepsilon}\right)\right]} + \frac{HN\left[\log^2 N + \log\left(\frac{1}{\varepsilon}\right)\right]}{n}\right] \\
\lesssim & \frac{n^{-1/2} + \sqrt{R}}{\sqrt{n}}\sqrt{N\left(\log^2 N + \log n\right)} + \frac{HN\left(\log^2 N + \log n\right)}{n} := \psi(R)\,, \qquad (37)
\end{aligned}
$$

where we choose $\varepsilon := n^{-1/2}$ in the last inequality, and then we conclude the proof. □

Based on the above result, we have the following proposition on generalization bounds in Besov spaces under non-iid state-action pairs.

Proposition 2. *Given the solution $\widehat{Q}_h^t = \operatorname{argmin}_{f \in \mathcal{F}_{\texttt{DNN}}} \widehat{\mathcal{E}}_h^t(f)$ in Eq. (5), then for a large N and any $\delta \in (0,1)$, with probability at least $1-\delta$, we have*

$$
\mathcal{E}_h^t(\widehat{Q}_h^t) - \min_{f \in \mathcal{F}_{\texttt{DNN}}} \mathcal{E}_h^t(f) \lesssim \frac{N\left[(\log N)^2 + \log n\right]}{n} + \frac{H\sqrt{N\left[(\log N)^2 + \log n\right]}}{n} + \frac{H^2\log(1/\delta)}{n}\,,
$$

where $n := \tilde{t}$ in our RL setting and N depends on t which needs further determined.

Proof. It is clear that $\psi(R)$ defined in Eq. (37) in Lemma 12 is a sub-root function. Therefore, the fixed point R^* of $\psi(R)$ can be analytically solved by the equation $R^* = \psi(R^*)$, which leads to

$$
R^* \lesssim \frac{\sqrt{N\left[(\log N)^2 + \log n\right]}}{n} + \frac{HN\left[(\log N)^2 + \log n\right]}{n}\,.
$$

Strictly speaking, there is an extra term $N\left[(\log N)^2 + \log n\right]^{\frac{3}{4}}/n$ in the above equation, but we can omit it as we only concern the smallest and largest order. By verifying the variance-expectation condition, we have

$$
\mathbb{E}[\mathcal{E}_h^t(\widehat{Q}_h^t) - \mathcal{E}_h^t(f_h^\star)]^2 \leqslant 16H^2\mathbb{E}[\mathcal{E}_h^t(\widehat{Q}_h^t) - \mathcal{E}_h^t(f_h^\star)]\,, \qquad (38)
$$

where $f_h^\star := \operatorname{argmin}_{f \in \mathcal{F}_{\texttt{DNN}}} \mathcal{E}_h^t(f)$ and we use the fact $\mathcal{E}_h^t(f)$ is $4H$-Lipschitz. Denote the function space $\widehat{\mathcal{F}_{\texttt{DNN}}}$ with the following function formulation for any $j \in [n]$

$$\hat{g}_h^t := \left[\widehat{Q}_h^t(s_h^{\tau_j}, a_h^{\tau_j}) - r_h(s_h^{\tau_j}, a_h^{\tau_j}) - V_{h+1}^t(s_{h+1}^{\tau_j})\right]^2 - \left[f_h^\star(s_h^{\tau_j}, a_h^{\tau_j}) - r_h(s_h^{\tau_j}, a_h^{\tau_j}) - V_{h+1}^t(s_{h+1}^{\tau_j})\right]^2 ,$$

we have $P_n \hat{g}_h^t = \widehat{\mathcal{E}}_h^t(\widehat{Q}_h^t) - \widehat{\mathcal{E}}_h^t(f_h^\star) \leqslant 0$ due to $\widehat{Q}_h^t = \operatorname{argmin}_{f \in \mathcal{F}} \widehat{\mathcal{E}}_h^t(f)$. Then using $\mathbb{E}g^2 \leqslant H^2 Pg$, for any $g \in \widehat{\mathcal{F}_{\texttt{DNN}}}$ by Eq. (38), according to Lemma 9, the following inequality holds with probability at least $1-\delta$

$$P\hat{g}_h^t \lesssim \frac{J}{H^2} R^* + \frac{(H^2 J + H)\log(1/\delta)}{n}, \quad \forall J > 1 ,$$

where which further implies

$$\mathcal{E}_h^t(\widehat{Q}_h^t) - \min_{f \in \mathcal{F}_{\texttt{DNN}}} \mathcal{E}_h^t(f) \lesssim \frac{N\left[(\log N)^2 + \log n\right]}{n} + \frac{H\sqrt{N\left[(\log N)^2 + \log n\right]}}{n} + \frac{H^2 \log(1/\delta)}{n} .$$

Finally, we conclude the proof. □

Proof of Theorem 1. Using the approximation error in $L^4(\mathcal{X})$ by Corollary 1, the smoothness parameter satisfies $\alpha > d(1/p - 1/4)_+$. By taking $\delta/2$ in Proposition 2, we have

$$\begin{aligned} \|\Gamma_h^t\|_{L^2(\mathrm{d}\bar{\mu}_h^{\tilde{t}})}^2 &\lesssim \left[\mathcal{E}_h^t(\widehat{Q}_h^t) - \min_{f \in \mathcal{F}_{\texttt{DNN}}} \mathcal{E}_h^t(f)\right] + \inf_{f \in \mathcal{F}_{\texttt{DNN}}} \|f - \mathbb{T}_h^\star Q_{h+1}^t\|_{L^4(\mathcal{X})}^2 \quad \text{[using Corollary 1]} \\ &\lesssim N^{-\frac{2\alpha}{d}} + \frac{N\left[(\log N)^2 + \log \tilde{t}\right]}{\tilde{t}} + \frac{H\sqrt{N\left[(\log N)^2 + \log \tilde{t}\right]}}{\tilde{t}} + \frac{H^2 \log(2/\delta)}{\tilde{t}} , \end{aligned} \tag{39}$$

where in the second inequality, taking $\alpha > d(1/p - 1/4)_+$, the approximation error can be estimated by Lemma 11

$$\inf_{f \in \mathcal{F}_{\texttt{DNN}}} \|f - \mathbb{T}_h^\star Q_{h+1}^t\|_{L^4(\mathcal{X})}^2 \lesssim N^{-2\alpha/d} .$$

Accordingly, the right hand side of Eq. (39) can be minimized by taking $N \asymp \tilde{t}^{\frac{d}{2\alpha+d}}$ up to $(\log \tilde{t})^3$-order in Eq. (33) for choosing suitable L, m, S, B. To make the architecture of deep RL independent of a variable $\tilde{t}$ (or t) during different episodes, here we directly choose $N \asymp T^{\frac{d}{2\alpha+d}} \log^3 T$, in this case, Eq. (39) can be formulated as

$$\|\Gamma_h^t\|_{L^2(\mathrm{d}\bar{\mu}_h^{\tilde{t}})}^2 \lesssim HT^{-\frac{2\alpha}{2\alpha+d}} \log^3 \tilde{t} + \frac{T^{\frac{d}{2\alpha+d}} \log^5 T}{\tilde{t}} + \frac{H^2 \log(2/\delta)}{\tilde{t}} ,$$

which requires the depth L and the width m up to

$$L \asymp \frac{d}{2\alpha+d} \log T, \quad m \asymp \frac{d}{2\alpha+d} T^{\frac{d}{2\alpha+d}} \log T .$$

Recall $\tilde{t} := \lceil \varrho t \rceil$, according to Lemma 4, if $\alpha > d(1/p - 1/4)_+$, then for any $\delta \in (0,1)$, with probability at least $1 - \delta/2$, the $\texttt{Term(i)}$ can be upper bounded by

$$\begin{aligned} \texttt{Term(i)} &\lesssim \left(\frac{\epsilon}{A}\right)^{-\frac{K}{2}} H\sqrt{T} \sqrt{\sum_{t=1}^{T} T^{-\frac{2\alpha}{2\alpha+d}} \log^3 \varrho t + \frac{T^{\frac{d}{2\alpha+d}} \log^5 T}{\varrho t} + \frac{H^2 \log(2/\delta)}{\varrho t}} + H\sqrt{T} \\ &\lesssim \left(\frac{\epsilon}{A}\right)^{-\frac{K}{2}} H\sqrt{T} \sqrt{HT^{\frac{d}{2\alpha+d}} \log^3 T + \frac{1}{\varrho} \int_1^{T+1} \left(\frac{T^{\frac{d}{2\alpha+d}} \log^5 T}{t} + \frac{H^2 \log(2/\delta)}{t}\right) \mathrm{d}t} + H\sqrt{T} \\ &\lesssim \left(\frac{\epsilon}{A}\right)^{-\frac{K}{2}} \frac{1}{\sqrt{\varrho}} \left(\sqrt{TH^3} \sqrt{T^{\frac{d}{2\alpha+d}} \log^6 T} + H^2 \sqrt{T} \sqrt{\log(2/\delta) \log T}\right) + H\sqrt{T} \\ &\lesssim \left(\frac{\epsilon}{A}\right)^{-\frac{K}{2}} \frac{1}{\sqrt{\varrho}} \left(H^{\frac{3}{2}} T^{\frac{\alpha+d}{2\alpha+d}} \log^3 T + H^2 \sqrt{T} \sqrt{\log(2/\delta)} \log T\right) + H\sqrt{T} . \end{aligned} \tag{40}$$

Then taking $\delta/2$ in the statistical error $\texttt{Term}(\texttt{ii})$ in Lemma 1, if $\alpha > d(1/p-1/4)_+$, with probability at least $1-\delta$, we have

$$\mathrm{Regret}(T) \lesssim \left(\frac{\epsilon}{A}\right)^{-\frac{K}{2}} \frac{1}{\sqrt{\varrho}} \left(H^{\frac{3}{2}} T^{\frac{\alpha+d}{2\alpha+d}} \log^3 T + H^2 \sqrt{T} \sqrt{\log\left(\frac{2}{\delta}\right)} \log T \right) + \epsilon H T + \sqrt{T H^3 \log\left(\frac{4}{\delta}\right)} \,.$$

Then taking

$$\epsilon = \mathcal{O}(H^{\frac{2}{K+2}} K^{\frac{2}{K+2}} A^{\frac{K}{K+2}} T^{-\frac{2\alpha}{(2\alpha+d)(K+2)}}) \,,$$

which implies

$$\mathrm{Regret} \lesssim \widetilde{\mathcal{O}}(H^{\frac{H+4}{H+2}} K^{\frac{2}{K+2}} A^{\frac{K}{K+2}} T^{\frac{\alpha K + (\alpha+d)(K+2)}{(2\alpha+d)(K+2)}}) \,.$$

Finally we conclude the proof.

□

G Proofs of regret bounds via two-layer neural networks

In this section, we focus on generalization bounds under the independent but non-identically distributed data setting in the Barron space, and it is useful to present estimates of our regret bound.

Lemma 13. *For two-layer ReLU neural networks with bounded ℓ_1 path norm defined in Eq.* (3) *given the function class $\mathcal{F}_{\mathtt{SNN}}$ and n independent but non-identically distributed data points $X = \{\boldsymbol{x}_i\}_{i=1}^n \subseteq \mathcal{X}$, then we have*

$$\mathcal{R}_n(\mathcal{F}_{\mathtt{SNN}}) \leqslant 2B\sqrt{\frac{2\log(2d)}{n}} \,.$$

Proof. Here we directly focus on the ℓ_1 path norm, which is different from [31, Theorem 3]. Based on the definition of two-layer ReLU neural networks defined in Eq. (3), denote $\widetilde{\boldsymbol{w}}_k := (\boldsymbol{w}_k^\top, c_k)^\top$ and $\widetilde{\boldsymbol{x}} = (\boldsymbol{x}^\top, 1)^\top$ for simplicity, the empirical Rademacher complexity of $\mathcal{F}_{\mathtt{SNN}}$ under our setting can be upper bounded by

$$\begin{aligned}
\widehat{\mathcal{R}}_n(\mathcal{F}_{\mathtt{SNN}}, X) &= \frac{1}{n}\mathbb{E}_{\boldsymbol{\xi}}\left[\sup_{f\in\mathcal{F}_{\mathtt{SNN}}} \frac{1}{m}\sum_{k=1}^m b_k \sum_{i=1}^n \xi_i \sigma(\widetilde{\boldsymbol{w}}_k^\top \widetilde{\boldsymbol{x}})\right] \\
&\leqslant \mathbb{E}_{\boldsymbol{\xi}}\left[\sup_{f\in\mathcal{F}_{\mathtt{SNN}}} \frac{1}{m}\sum_{k=1}^m |b_k| \, \|\widetilde{\boldsymbol{w}}_k\|_1 \frac{1}{n}\left|\sum_{i=1}^n \xi_i \sigma\left(\frac{\widetilde{\boldsymbol{w}}_i}{\left\|\widetilde{\boldsymbol{w}}_i^\top\right\|_1}\widetilde{\boldsymbol{x}}_i\right)\right|\right] \\
&\le B\mathbb{E}_{\boldsymbol{\xi}}\left[\sup_{\|\widetilde{\boldsymbol{w}}\|_1\leqslant 1} \frac{1}{n}\left|\sum_{i=1}^n \xi_i\sigma(\widetilde{\boldsymbol{w}}^\top\widetilde{\boldsymbol{x}}_i)\right|\right] \\
&\leqslant 2B\mathbb{E}_{\boldsymbol{\xi}}\left[\sup_{\|\widetilde{\boldsymbol{w}}\|_1\leqslant 1}\frac{1}{n}\sum_{i=1}^n \xi_i \widetilde{\boldsymbol{w}}^\top \widetilde{\boldsymbol{x}}_i\right] \quad \text{[using symmetry of } \boldsymbol{\xi} \text{ and 1-Lipschitz of ReLU]} \\
&\leqslant 2B\mathbb{E}_{\boldsymbol{\xi}}\left\|\frac{1}{n}\sum_{i=1}^n \xi_i\widetilde{\boldsymbol{x}}_i\right\|_\infty \,, \quad \text{[using Hölder inequality]}
\end{aligned}$$

where the first inequality holds by the homogeneity of ReLU for any $\widetilde{\boldsymbol{w}} \in \mathbb{R}^d/\{\mathbf{0}\}$. Since the Massart's lemma is still valid under our independent but non-identically distributed data, $\mathcal{R}_n(\mathcal{F}_{\mathtt{SNN}})$ can be further expressed by

$$\widehat{\mathcal{R}}_n(\mathcal{F}_{\mathtt{SNN}}, X) \leqslant 2B\mathbb{E}_{\boldsymbol{\xi}}\left\|\frac{1}{n}\sum_{i=1}^n \xi_i\widetilde{\boldsymbol{x}}_i\right\|_\infty \leqslant 2B\sqrt{2\log(2d)/n} \,,$$

where the last inequality holds by the maximum of n sub-Gaussian random variables [72] since Rademacher random variables are sub-Gaussian, and finally we conclude the proof. □

Proof of Theorem 2. Denote $X := \{(s_h^{\tau_j}, a_h^{\tau_j}, s_{h+1}^{\tau_j})\}_{j=1}^{\tilde{t}}$ for simplicity and notice that the function $[f(s_h, a_h) - r_h(s_h, a_h) - V_{h+1}^t(s_{h+1})]^2$ is $4H$-Lipschitz. Then according to Lemma 7, for any $\delta \in (0,1)$, the following result holds with probability at least $1 - \delta/2$

$$\begin{aligned} \widehat{\mathcal{E}}_h^t(f) - \mathcal{E}_h^t(f) &\leqslant 2\widehat{\mathcal{R}}_{t-1}(\mathcal{F}_{\texttt{SNN}}, X) + 12H\sqrt{\frac{\log(4/\delta)}{2\tilde{t}}} \\ &\leqslant 8B\widetilde{R}H\sqrt{\frac{2\log(2d)}{\tilde{t}}} + 12H\sqrt{\frac{\log(4/\delta)}{2\tilde{t}}}\,, \end{aligned} \tag{41}$$

where we use the empirical Rademacher complexity in Lemma 13. Accordingly, by Lemma 5 and Eq. (41), then with probability at least $1 - \delta/2$, we have

$$\begin{aligned} \|\Gamma_h^t\|^2_{L^2(\mathrm{d}\mu_h^{\tilde{t}})} &\leqslant \left[\mathcal{E}_h(\widehat{Q}_h^t) - \min_{f \in \mathcal{F}_{\texttt{SNN}}} \mathcal{E}_h(f)\right] + \inf_{f \in \mathcal{F}_{\texttt{SNN}}} \|f - \mathbb{T}_h^\star Q_{h+1}^t\|^2_{L^2(\mathrm{d}\bar{\mu}_h^{\tilde{t}})} \\ &\leqslant \mathcal{E}_h(\widehat{Q}_h^t) - \min_{f \in \mathcal{F}_{\texttt{SNN}}} \mathcal{E}_h(f) + \frac{3\|\mathbb{T}_h^\star Q_{h+1}^t\|^2_{\mathcal{P}}}{m} \\ &\leqslant 2 \sup_{f \in \mathcal{F}_{\texttt{SNN}}} |\mathcal{E}_h^t(f) - \widehat{\mathcal{E}}_h^t(f)| + \frac{3\widetilde{R}^2}{m} \quad \text{[using Assumption 2]} \\ &\leqslant 16B\widetilde{R}H\sqrt{\frac{2\log(2d)}{\tilde{t}}} + 24H\sqrt{\frac{\log(4/\delta)}{2\tilde{t}}} + \frac{3\widetilde{R}^2}{m}\,, \quad \text{[using Eq. (41)]} \end{aligned} \tag{42}$$

where the second inequality uses the approximation result for two-layer ReLU neural networks and the Barron space in [31, Theorem 4]. Accordingly, by Lemma 4, for any $\delta \in (0,1)$, the $\texttt{Term(i)}$ in the regret decomposition can be upper bounded with probability at least $1 - \delta/2$

$$\begin{aligned} \texttt{Term(i)} &\lesssim \left(\frac{\epsilon}{A}\right)^{-\frac{K}{2}} H\sqrt{T}\sqrt{\sum_{t=1}^{T}\left(BH^2(\varrho t)^{-\frac{1}{2}}\sqrt{\log d} + H(\varrho t)^{-\frac{1}{2}}\sqrt{\log\frac{4}{\delta}} + \frac{H^2}{m}\right)} + H\sqrt{T} \\ &\lesssim \left(\frac{\epsilon}{A}\right)^{-\frac{K}{2}} H\sqrt{T}\sqrt{\frac{1}{\sqrt{\varrho}}\int_1^{T+1}\left(BH^2 t^{-\frac{1}{2}}\sqrt{\log d} + Ht^{-\frac{1}{2}}\sqrt{\log\frac{4}{\delta}}\right)\mathrm{d}t + \frac{H^2T}{m}} + H\sqrt{T} \\ &\lesssim \left(\frac{\epsilon}{A}\right)^{-\frac{K}{2}} \frac{1}{\sqrt{\varrho}}\left[T^{\frac{3}{4}}H^2B(\log d)^{\frac{1}{4}} + T^{\frac{3}{4}}H^{\frac{3}{2}}\log^{\frac{1}{4}}\left(\frac{4}{\delta}\right)\right] + \left(\frac{\epsilon}{A}\right)^{-\frac{K}{2}}\frac{H^2T}{\sqrt{m}} + H\sqrt{T}\,. \end{aligned} \tag{43}$$

where we use $\widetilde{R} \asymp H$ and $\int_1^T (t-1)^{-1/2}\mathrm{d}t = \mathcal{O}(\sqrt{T})$.

Accordingly, taking $\delta/2$ in the statistical error $\texttt{Term(ii)}$ in Lemma 1, then with the probability at least $1 - \delta$, the total regret can be upper bounded by

$$\text{Regret}(T) \lesssim \left(\frac{\epsilon}{A}\right)^{-\frac{K}{2}}\left(\frac{H^2T^{\frac{3}{4}}}{\sqrt{\varrho}}\left[B(\log d)^{\frac{1}{4}} + \log^{\frac{1}{4}}\left(\frac{4}{\delta}\right)\right] + \frac{H^2T}{\sqrt{m}}\right) + \epsilon HT + \sqrt{TH^3\log\left(\frac{4}{\delta}\right)}\,.$$

Taking $m = \Omega(\sqrt{T})$ and $\epsilon = \mathcal{O}\left(H^{\frac{2}{K+2}}T^{-\frac{1}{2(K+2)}}\right)$, the regret bound can be further represented as

$$\text{Regret}(T) \lesssim \widetilde{\mathcal{O}}(H^{\frac{K+4}{K+2}}T^{\frac{2K+3}{2K+4}})\,,$$

which concludes the proof. □