Title
Reliable Prediction and Decision-Making in Sequential Environments

Author
Rashidinejad, Paria

Publication Date
2022

Peer reviewed|Thesis/dissertation

Reliable Prediction and Decision-Making in Sequential Environments

by

Paria Rashidinejad

A dissertation submitted in partial satisfaction of the

requirements for the degree of

Doctor of Philosophy

in

Electrical Engineering and Computer Sciences

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Stuart Russell, Chair
Professor Jiantao Jiao
Professor Sergey Levine
Professor Paul Grigas

Spring 2022

Reliable Prediction and Decision-Making in Sequential Environments

Abstract

Reliable Prediction and Decision-Making in Sequential Environments

by

Paria Rashidinejad

Doctor of Philosophy in Electrical Engineering and Computer Sciences

University of California, Berkeley

Professor Stuart Russell, Chair

Building autonomous agents that learn to make predictions and take actions in sequential environments is a central problem in artificial intelligence, with applications as diverse as personalized medicine, self-driving cars, finance, and scientific discovery. Despite impressive success in certain areas such as natural language, games, and robotic demonstrations, sequential prediction and decision-making remains challenging in the absence of known models, accurate environment simulators, short-range dependencies, and large and diverse datasets.

In this thesis, we formulate problems to capture challenging yet prevalent settings encountered in the real-world. Given the formulations, we then design reliable and efficient learning algorithms, leveraging recent advances in statistics and optimization. In the first part of the thesis, we consider the problem of learning to make predictions in unknown and only partially observed linear dynamical systems. Contrary to prior predictive models which fail in the presence of long-range dependencies, we design an algorithm that provably returns near-optimal predictions regardless of the system's degree of stability and forecast memory.

In the second part, we shift our attention to reinforcement learning (RL), the problem of learning to make decisions in an unknown sequential environment. We start by focusing on the offline setting, where the agent is only provided with a previously-collected dataset of interactions and does not have further access to the environment. We propose a new framework to study offline learning problems given datasets of any composition, ranging from expert-only to uniform coverage, and thus unifying two main offline learning paradigms: imitation learning and vanilla offline RL. Equipped with this framework, we design an algorithm based on pessimism in the face of uncertainty and prove that it is nearly optimal for any, possibly unknown dataset composition.

We then turn to the online setting, where the agent learns while interacting with the environment. In this setting, the agent faces a dilemma in each step: whether it should exploit the current knowledge and select a seemingly optimal action or it should explore and visit differ-

ent regions of the environment. We propose a framework that unifies common exploration methods by adding an adaptive regularizer to the standard RL objective. We show that a particular regularizer design yields a simple optimistic exploration strategy that enjoys fast optimization and efficient exploration, achieving state-of-the-art performance in several locomotion and navigation tasks when combined with deep neural networks.

To Mom, Dad, Maryam, and Amin

Contents

List of Figures

List of Tables

Acknowledgments

I owe this thesis to many, whose support, guidance, and friendship have been crucial in its materialization.

I am forever indebted to my advisor Stuart Russell, who welcomed me into his research group Russell's Unusual Group of Students (RUGS) and set me on the path to a career in artificial intelligence. All throughout, he has been incredibly supportive, has given me freedom to explore, encouraged me to keep an eye on the big problems, and inspired me to invest in clear communication of ideas. I am grateful for his valuable insight and feedback and for making me a better researcher. He will always remain an inspiration to me.

I am equally grateful to my co-advisor Jiantao Jiao, with whom I started working later in my PhD journey. He has been incredibly generous with his time, giving valuable advice about anything and whenever I needed it. He has taught me a lot, from asking interesting questions and capturing the essence of a complex problem through simplification to writing and presenting my research. I am grateful for his confidence in what I could accomplish.

I have also had the pleasure of collaborating with Xiao Hu on projects related to physiological monitoring. Though the results of our collaboration are not included in this thesis, his guidance has enriched my perspective on challenges AI faces in applications like healthcare. He is a generous and caring mentor and I am grateful for his continued support.

In my time at Berkeley, I have had several other incredible mentors and collaborators. I thank Yusuf Bugra Erol, Karthika Mohan, and Dave Moore for giving valuable advice in earlier years of my PhD. I greatly enjoyed working with Cong Ma and Banghua Zhu on our "tale of pessimism" for offline RL and Tianjun Zhang on our online RL project. I was fortunate to have received mentorship from Yuandong Tian and I deeply appreciate his support of my career.

I would like to express my gratitude to EECS faculty and many colleagues, particularly in BAIR Lab, BLISS, and CHAI, for always being happy to help and for creating a collaborative and friendly environment. I am especially thankful to Sergey Levine, Paul Grigas, and Moritz Hardt for serving on my committees and Tom Courtade and Bora Nikolic for facilitating my arrival at Berkeley and providing guidance during my first year. I would also like to thank the EECS staff. In particular, I am extremely thankful to Shirley Salanio for always being so kind and helpful and for cheering me on all these years.

I am grateful for the time I spent as an intern at Systems Research Group at Oracle Labs. I would like to thank Vincent Lee for introducing me to the group, and my mentors Arun Raghavan, Navaneeth Jamadagni, and Suwen Yang for introducing me to questions around datacenter reliability. I am also thankful to Vincent Lee for his support during my internship and Shu Guanghua, Onur Kocberber, and Craig Schelp for helpful discussions.

Finally, I am incredibly thankful to my partner Amin for constant support, understanding, and all the words of encouragement. I am forever grateful to my parents Mitra and Hamidreza and my sister Maryam for their unwavering encouragement and unconditional love throughout my life. Thank you all for being there for me in smiles or tears, success or failure. This thesis is dedicated to you.

Chapter 1

Introduction

One of the central expressions of intelligence is the ability to complete tasks in the pursuit of certain goals in an environment, perceived through measurements. This thesis is concerned with designing artificially intelligent agents that reliably complete tasks in environments of sequential nature. This specification involves three components: a *sequential environment* or a dynamical system evolving over a dimension (often time), a *task* specifying the type of goal being pursued, and a *criteria* evaluating the goodness of an executed solution.

Consider for instance a sequential environment evolving over time. One example of the above specification is the task of making predictions about future observations given all available measurements with the criteria of achieving minimal *regret*, which compares agent's performance to that of an idealized benchmark. Another example is the task of learning a policy for manipulating the environment through a set of actions so as to maximize a notion of expected return.

When the mathematical laws governing the evolution of the environmental variables and their connections to the measurements are known, finding the best solution to complete the task becomes a purely computational problem. The optimal solution depends on all available measurements and yields the highest goodness, based on a scalar criteria. In the majority of cases, however, some aspects of environment evolution are unknown. For instance, unknown perturbations might be affecting the measurements and parameters describing the evolution or even the mathematical laws of evolution themselves might be unknown. This is where learning comes into the picture.

Learning refers to the process of building one or more mathematical models from available measurements. In sequential tasks, while model learning is most commonly referred to estimating the environment dynamics, learning in principle can be used to build other task-relevant models such as predictive models or policies.

Different modeling and learning algorithms may be designed to solve similar tasks. One central aspect that differentiate learning algorithms is their *sample complexity*, which refers to the number of samples required for a specific algorithm to reach a desired level of performance. Empirical evaluations on a set of tasks can be used to assess the sample complexity of algorithms. While such evaluations are flexible, comparing learning algorithms solely

based on the them and on limited sets of tasks have drawbacks. For example, recent studies suggest that the performance of many algorithms are not robust to the changes in their implementation, hyperparameters, or random seeds (Islam et al., 2017; Fortunato et al., 2018; Ilyas et al., 2019). On the flip side, theoretical finite-sample complexity alleviate such issues and deliver insights into the performance and reliability of learning algorithms, often by providing upper and lower bounds on the number of samples required to reach a performance guarantee.

Recent years have observed machine learning algorithms achieving impressive performance in certain domains. Examples include OpenAI's GPT-3, a language model that produces human-like text, and DeepMind's AlphaGo, a reinforcement learning agent that has beat the world's champion in the game of Go. These success stories, however, often rely on access to enormous and diverse datasets, expert demonstrations, or accurate simulators of the environment. Such requirements and lack of theoretical foundations for algorithms are among the reasons that preclude reliable integration of AI into many important applications.

Take AI for physiological monitoring as an example. The sequential environment of human physiology does not satisfy common statistical assumptions, exhibiting non-stationarity and long-term dependencies (Ghassemi et al., 2015; Alaa et al., 2017; Wiens et al., 2019; Rashidinejad et al., 2020a). Healthcare data are often small and sporadic due to the costs of expert labels, scarcity of certain physiological conditions, and irregular measurement times. These datasets are also relatively narrow, as data are collected by specific policies, and further data collection, for instance by exploring various treatment options, is heavily restricted. Furthermore, AI for physiological monitoring needs to be reliable, safe, explainable, and personalizable.

In the quest to build AI that can be integrated into the real world, in this thesis we formulate problems and design algorithms towards reliable prediction and decision-making in sequential environments.

Thesis roadmap

This thesis is based on three of the author's publications on prediction and decision-making in sequential environments. The rest of this chapter is dedicated to an overview of notations and a synopsis of technical contributions and results.

Section 1.1 of this introduction summarizes the problem formulation, techniques, and results of the work Rashidinejad et al. (2020c) on learning to make predictions in an unknown and partially-observed linear dynamical system (LDS), one of the most commonly used sequential models. The main contribution of this work is presenting the first provably sample-efficient prediction algorithm, whose regret is independent of both system's degree of stability and forecast memory. A detailed discussion of the algorithm and technical contributions are deferred to Chapter 2.

Section 1.2 outlines contributions on learning to make decisions in an unknown Markov decision process (MDP), a commonly used sequential model in reinforcement learning (RL). Policy learning is considered under two settings. The first setting is offline RL, in which

learning is based on access to only a previously-collected dataset of interactions. A new framework is proposed for studying offline learning problems which admit datasets with any *composition*, ranging from expert-collected to uniform coverage. Given this framework, we propose a new algorithm and conduct an extensive study of finite-sample properties of the algorithm as well as information-theoretic limits under a variety of settings. A more in-depth discussion of the offline setting is presented in Chapter 3, which is a lightly edited version of the publication Rashidinejad et al. (2021).

Online RL is the second setting considered, in which learning occurs while interacting with the environment. The online setting brings about a key question on the trade-off between exploring the environment and exploiting the already acquired knowledge. As detailed in Chapter 4, we frame this trade-off as adding a regularizer to the standard RL objective and proposing a new exploration algorithm that enjoys convergence guarantees and sample efficient exploration on a variety of navigation and locomotion tasks. This chapter is based on the publication Zhang et al. (2021b).

We conclude the thesis by reviewing open challenges and discussing promising avenues for future work in Chapter 5.

Notation

We denote the space of reals by $\mathbb{R}$, the space of n-dimensional real vectors by $\mathbb{R}^n$, and the space of $n \times m$-dimensional real matrices by $\mathbb{R}^{n\times m}$. We use calligraphy letters to denote sets and given a set $\mathcal{S}$, we write $|\mathcal{S}|$ to represent the cardinality of $\mathcal{S}$. The probability simplex over a set $\mathcal{S}$ is denoted by $\Delta(\mathcal{S})$.

Vectors are denoted by small letters and are assumed to be column vectors except for the probability and measure vectors. We denote by $\|.\|_2$, the Euclidean norm of vectors. For two n-dimensional vectors x and y, we use $x \cdot y = x^\top y = \langle x, y\rangle$ to denote their inner product and $x \le y$ to denote the element-wise inequality $x_i \le y_i$ for all $i \in \{1, \dots, n\}$. The vertical concatenation of $x_1, \dots, x_t \in \mathbb{R}^n$ is written as $x_{1:t} \in \mathbb{R}^{nt}$. We use $x_t(i)$ to refer to the i-th element of the vector $x_t = [x_t(1), \dots, x_t(n)]^\top$.

Throughout this text, matrices may be denoted by capital, small, or calligraphy letters. The spectral radius of a square matrix A is denoted by $\rho(A)$. We denote by $\|A\|_2$, the operator two norm of the matrix A. The eigenpairs of an $n \times n$ matrix are written as $\{(\sigma_j, \phi_j)\}_{j=1}^n$ where $\sigma_1 \ge \dots \ge \sigma_n$ are the eigenvalues and $\{\phi_j\}_{j=1}^k$ are the top k eigenvectors. The horizontal concatenation of matrices $A_1, \dots, A_n$ with appropriate dimensions, is denoted by $[A_i]_{i=1}^n = [A_1|\dots|A_n]$. The Kronecker product of matrices A and B is denoted by $A \otimes B$. Identity matrix of dimension n is represented by I_n.

We write $x \lesssim y$ when there exists a constant $c > 0$ such that $x \le cy$ and we write $x \lesssim_b y$ when c is a constant that only depends on b. We use the notation $x \asymp y$ if constants $c_1, c_2 > 0$ exist such that $c_1|x| \le |y| \le c_2|x|$ and write $x \asymp_b y$ if $c_1, c_2 > 0$ only depend on b. We write $x \vee y$ to denote the supremum of x and y. We write $f(x) = O(g(x))$ if there exists some positive real number M and some x_0 such that $|f(x)| \le Mg(x)$ for all $x \ge x_0$. We use

$\widetilde{O}(\cdot)$ to be the big-O notation ignoring logarithmic factors. We write $f(x) = \Omega(g(x))$ if there exists some positive real number M and some x_0 such that $|f(x)| \geq Mg(x)$ for all $x \geq x_0$.

$x \coloneqq y$ is used when quantity x is taken to be equal to quantity y by definition. Given a probability distribution p, we write $x_1, \ldots, x_n \sim p$ if random variables $x_1, \ldots, x_n$ are drawn independently and identically distributed from distribution p. $\mathbb{E}[X]$ and $\mathbb{V}[X]$ respectively denote the expected value and variance of random variable X.

1.1 Learning to make predictions

Chapter 2 considers partially-observed linear dynamical system (LDS), one of the most popular sequential environments, whose evolution is described as:

$$\begin{aligned} h_{t+1} &= Ah_t + Bx_t + \eta_t, \\ y_t &= Ch_t + Dx_t + \zeta_t. \end{aligned} \tag{1.1}$$

Here, h_t is the latent (hidden) state, x_t is the control input, y_t is the observation, and η_t and ζ_t respectively denote the process and measurement noise vectors with covariance matrices Q and R. Matrix A is the transition matrix and controls how fast the process mixes—i.e., how fast the marginal distribution of y_t becomes independent of y_1. Furthermore, matrix A determines whether the above LDS is stable $\rho(A) < 1$, marginally stable $\rho(A) = 1$, or unstable $\rho(A) > 1$, with $\rho(A)$ determining the system's degree of stability. At any step, our goal is to make reliable predictions about observation y_{t+1} given system input x_{t+1} and the causal history, i.e. all past observations $y_1, \ldots, y_t$ and inputs $x_1, \ldots x_t$.

If the system parameters, which are matrices A, B, C, D and noise covariance matrices Q and R, are known, the optimal linear[1] predictions are given by the celebrated Kalman filter. The recursive representation of the (stationary) Kalman predictive model also obeys a linear law

$$\begin{aligned} h_{t+1|t} &= Gh_{t|t-1} + Ky_t + (B - KD)x_t, \\ y_{t+1|t} &= Ch_{t+1|t} + Dx_{t+1}, \end{aligned} \tag{1.2}$$

where $h_{t+1|t}$ and $y_{t+1|t}$ denote the optimal state and observation predictions, respectively. Here, matrix K is called the (predictive) Kalman gain and matrix $G \coloneqq A - KC$ is called the closed-loop matrix, known to be marginally stable $\rho(G) \leq 1$. Notice that for the LDS that describes the Kalman predictive model, G determines the *forecast memory* of the system, i.e. how many past data points are ought to be considered to make a sound prediction.

If the system parameters are unknown, a common approach is to first estimate system parameters from data, which is referred to as *system identification*, and then make predictions using the Kalman formula. Direct identification of system (1.1), however, requires solving a

[1] For Gaussian process and measurement noise, Kalman predictive model returns optimal predictions, minimizing the conditional mean squared loss.

non-convex optimization problem via classical algorithms such as expectation maximization (EM), for which finite-sample guarantees are difficult.

Recently, several algorithms are developed that enjoy theoretical guarantees, yet these predictors either do not converge to the optimal predictions of a Kalman model in hindsight or their prediction error increases with system's degree of stability $\rho(A)$ and/or forecast memory $\rho(G)$. Indeed, the majority of popular prediction algorithms, such as the ones based on recurrent networks, effectively predict based on a finite lookback window only (Miller and Hardt, 2018), despite the ubiquity of long-range dependencies in many areas such as econometrics, linguistics, medicine, and climate sciences (Doukhan et al., 2002; Beran, 2017).

As laid out in detail in Chapter 2, we are interested in developing a reliable prediction algorithm even with the existence of long-range dependencies. More formally, we want the cumulative performance of the algorithm's predictions $\hat{y}_{t|t-1}$ compared to the Kalman predictions in hindsight $y_{t|t-1}$ defined as

$$\text{Regret}(T) := \sum_{t=1}^{T} \|y_t - \hat{m}_t\|_2^2 - \|y_t - m_t\|_2^2, \tag{1.3}$$

to be sublinear in T with high probability.

We introduce the prediction algorithm illustrated in Figure 1.1, named SLIP (spectral LDS improper predictor), which relies on two key observations described below.

Bypassing system identification. Since our goal is to design a predictor, we bypass the difficult system identification step and instead learn a computational process that directly approximates the Kalman predictor. To achieve this, we take a look at the expansion of the recursive formula for the Kalman predictor:

$$y_{t|t-1} = \sum_{i=1}^{t-1} CG^{t-i-1}Ky_i + \sum_{i=1}^{t-1} CG^{t-i-1}(B - KD)x_i + Dx_t. \tag{1.4}$$

From expansion above, it is apparent that first, the least-squares objective is non-convex in the predictor parameters C, G, K, B, D, and second, the optimal predictions are linear in current input x_t and causal history $y_{1:t-1}, x_{1:t-1}$.

Convex relaxation. *Proper learning* is learning from the original hypothesis class $\mathcal{H}$. In the case of linear dynamical systems, original hypothesis class refers to the system parameters $\mathcal{H} = \{A, B, C, D, Q, R\}$ and identifying these parameters is a non-convex problem as discussed earlier. On the other hand, in *improper learning* paradigm, one may reparameterize the hypothesized model class by an alternative class $\widetilde{H}$, which is often a relaxation that results in an easier optimization problem or improved computational efficiency. To meet our goal on theoretical guarantees, we resort to improper learning by slightly overparameterizing the Kalman predictive model and conducting a convex relaxation.

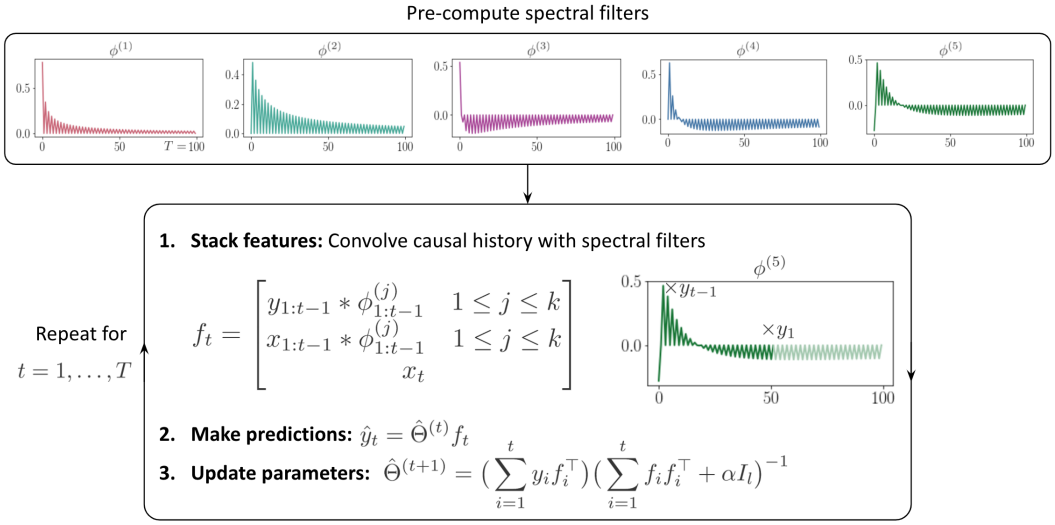

Figure 1.1: Block-diagram of the SLIP algorithm. First, given a horizon T, a set of k spectral filters are computed via the spectral decomposition of a particular Hankel matrix. Then, at every step, a feature vector is constructed by stacking quantities obtained by convolving previous observations and inputs with each spectral filter. The predictions are computed by multiplying the feature vector with a parameter matrix. Upon receiving the observations, the parameters are updated by minimizing a regularized least squares loss.

To ensure a good performance, the overparameterization for convex relaxation should first, be accurate and approximate the Kalman predictive model closely, and second, be tight and avoid introducing too many parameters. Accomplishing both of these goals at once may not always be possible and the two goals may even be at odds, resulting in a form of bias-variance tradeoff. We investigate the possibility of a tight convex relaxation by analyzing a generalized version of *Kolmogorov k-width* of the Kalman predictor coefficient set. The Kolmogorov k-width measures how well a set can be approximated by a low-dimensional linear subspace (Pinkus, 2012). Unlike other complexity notions, Kolmogorov width measures the approximation error in terms of the worst-case error, which provides a uniform upper bound on the convex relaxation error regardless of the ground truth values of the LDS parameters in a particular instance.

Our first result regarding the feasibility of conducting a tight convex relaxation is presented below.

Theorem 1.1 (Kalman predictor k-Width, Informal)**.** *The following statements hold for the stationary Kalman predictor in* (1.2) *with* $\rho(A) \le 1$*:*

1. *For a general closed-loop matrix G with $\rho(G) \le 1$, linear subspaces cannot be exploited to efficiently approximate the Kalman predictions.*

2. *Restricting G to be diagonalizable with real eignevalues, the Kalman predictor can be approximated efficiently via fixed known filters.*

The above theorem states that a tight convex relaxation of the Kalman predictive model, uniformly for any LDS with $\rho(A) \leq 1$ and $\rho(G) \leq 1$, is possible provided that the closed-loop matrix G is diagonalizable with real eigenvalues. Although a tight convex relaxation is necessary, it is not sufficient to ensure accurate learning and sublinear regret.

Analyzing the regret of the SLIP algorithm relies on understanding the finite sample properties of regularized least-squares in the presence of non-independent sequential data potentially generated by a system that is only marginally stable. Colloquially speaking, unlike the common assumptions of bounded feature norms, in marginally stable systems the feature norm can grow (polynomially) over time. This precludes using a *mixing time* argument, which is one of the most established techniques in statistics for handling non-independent sequential processes (Yu, 1994).

Informally, the mixing time of a sequence refers to the smallest window τ such that the elements that are at least as far apart as τ become "almost statistically independent". For the predictor in (1.4) with $\rho(G) < 1$, mixing time is related to the spectral norm of G according to $1/(1-\rho(G))$. This is due to the fact the definition of spectral norm ensures $\|G^i\| \approx \rho(G)^i$ and thus at any step t one can neglect the data received much earlier than $t - 1/(1-\rho(G))$. The mixing time technique argues that subsequences separated by τ are nearly statistically independent and thereby the estimation rates becomes analogous to those in i.i.d. systems, multiplied by a factor proportional to the mixing time. Hence, if a predictor has a uniform regret guarantee independent of the mixing time, the analysis must exploit arguments other than mixing time to establish the regret bound.

We obtain a uniform regret guarantee for the SLIP algorithm by decomposing the regret into several interpretable terms and bounding them separately. The key techniques that we use in our analysis are: (1) self-normalizing martingale properties and concentration results such as the ones given in Abbasi-Yadkori et al. (2011), and (2) block martingale small-ball condition which we establish for the feature process. The second technique is inspired by recent works of Mendelson (2014); Simchowitz et al. (2018), in which conditions other than concentration are shown that can also facilitate efficient learning. The following theorem gives an informal statement of the regret guarantee for the SLIP algorithm.

Theorem 1.2 (SLIP regret, Informal)**.** *Assume G is diagonalizable with real eignevalues and set the number of filters to $k \asymp \log^2 T$. Under mild assumptions,*

$$\text{Regret}(T) := \sum_{t=1}^{T} \|y_t - \hat{m}_t\|_2^2 - \|y_t - m_t\|_2^2 \lesssim \text{polylog}(T)$$

with high probability, uniformly for all $\rho(A) \leq$ and $\rho(G) \leq 1$.

In addition to theoretical results, we present empirical evaluations in Chapter 2 showing that SLIP significantly improves prediction accuracy compared to state-of-the-art methods.

1.2 Learning to make decisions

In this section, we consider decision-making in sequential environments. One of the primary frameworks to study sequential decision-making is reinforcement learning, whose goal is maximize a cumulative reward objective. RL is closely related to the field of optimal control, which is concerned with setting control inputs to satisfy a scalar objective. Furthermore, a large part of neuroscience is devoted to understanding how the brain makes decisions which is related to the neurotransmitter dopamine and its role resembles that of the reward in RL. The RL framework is very general and can capture many problems arises in practice; to the extent that a recent paper by Silver et al. (2021) hypothesizes that the reward maximization objective alone contains many or possibly even all the goals of intelligence.

What differentiates reinforcement learning from supervised learning is that the only supervision in RL is a *reward signal*, which provides a partial feedback and causes a credit assignment problem. The partial (or bandit) feedback refers to the fact that the agent only observes the reward signal on the actions it takes and not on other possible actions that it could have chosen. The credit assignment problem refers to the fact that earlier decisions may have delayed consequences, affecting the future states, available choices, and rewards.

These aspects alongside difficulties of data collection (exploration) and insufficient data coverage, makes the RL setting significantly more challenging than supervised learning. Indeed, the current successful RL-based programs are heavily restricted. For example, AlphaGo, the first program that has defeated the Go world champion, leverages known environment dynamics and finds a policy based on over 10 million self-plays. Other examples such as AlphaStar for Start Craft II or OpenAI Five for Dota 2 are trained over 10,000 years of the AI playing the game with itself. Despite the capacity that RL offers, obtaining such large amounts of data in many real applications is completely prohibitive, which motivates this thesis to investigate the challenges RL faces in practice.

We study RL in fully-observed Markov decision processes (MDPs), one of the most commonly-used sequential environments. In MDPs, upon taking an action a from an (observable) state s, the agent receives a (possibly random) reward r and transitions to the next state s'. Over time, this procedure generates a sequence $\{s_t, a_t, r_t\}_{t\geq 0}$. A policy π specifies which actions should be taken at any state. We write a deterministic policy as $a = \pi(s)$, which is a map from states to actions, and a stochastic policy as $a \sim \pi(\cdot \mid s)$, which defines a distribution over actions in each state. We consider the discounted infinite horizon setting, where a policy is ran for an infinite horizon and future rewards are discounted by a factor of $\gamma \in [0, 1)$ per time step. Any policy π induces a discounted occupancy or visitation distribution $d^\pi(s, a)$, which roughly characterizes the probability (density) (s, a) is occupied by π. We refer the reader to Section 3.1.1 for more formal definitions.

The RL objective (also called the return) is to find a policy that maximizes the expected cumulative discounted reward specified below:

$$J(\pi) \coloneqq \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r_t \middle| \pi\right] = \frac{1}{1-\gamma}\, \mathbb{E}_{s,a\sim d^\pi(\cdot,\cdot)}[r(s, a)]. \tag{1.5}$$

In the equation above, the second inequality gives an equivalent characterization of the RL objective—the reward averaged over the policy occupancy distribution and multiplied by the total amount of time spent in the environment, i.e. the effective horizon $1/(1-\gamma)$. We denote the optimal policy by $\pi^\star \in \arg\max J(\pi)$. Given any *target policy*[2] π, the expected sub-optimality of a learned policy $\hat{\pi}$ with respect to (or competing with) π is

$$\mathbb{E}[J(\pi) - J(\hat{\pi})], \tag{1.6}$$

where the expectation is taken with respect to all the randomness, namely in the data generation procedure or the algorithm.

In what follows, we describe challenges faced by RL in two learning modes of sequential decision-making: learning from a fixed dataset of prior interactions and learning while interacting with the environment.

1.2.1 Decision-making from a dataset

In this section, we study RL in the offline or batch setting, where the agent's goal is to achieve competence in a task using only a previously-collected dataset of interactions without further access to the environment. We denote this dataset by $\mathcal{D} = \{(s, a, r, s')\}$ which consists of N tuples of state, action, reward, and next state. As discussed earlier, RL in the current stage has a high data demand and thus, offline RL plays a critical role for exploiting previously-collected data. Furthermore, offline RL avoids interactive exploration which can be costly, dangerous, or even impossible in many real applications (Levine et al., 2020).

One the main challenges in learning from offline data is handling different types of datasets. There are two main categories of methods that are applied based on the composition of the offline dataset. The first one is *imitation learning* or behavioral cloning, which is suitable for data collected by an expert and is a supervised learning method. In theory, imitation learning achieves a sub-optimality that decays according to $1/N$, with N being the number of samples, and in practice, it is observed to succeed with relatively few samples (Vinyals et al., 2019; Salimans and Chen, 2018). The second category is vanilla offline RL which requires the dataset to cover states and actions uniformly, both from theoretical and practical standpoints. In practice, generic offline RL algorithms are observed to perform poorly for narrower datasets, such as the ones that include human demonstrations or are collected by handcrafted policies (Levine et al., 2020).

Yet, real datasets often deviate from the two extremes of expert-only and uniform coverage and the exact composition of the dataset can be unknown apriori. To bridge this gap, we present a new offline RL framework that smoothly interpolates between the two extremes of data composition, hence unifying imitation learning and vanilla offline RL. The new framework is centered around a weak version of the *concentrability coefficient* that measures the deviation of the behavior policy to the expert policy alone. Specifically, denoting by μ the

[2]Target policy can be viewed as an arbitrary policy against which the algorithm performance is measured.

distribution over state-action pairs in the data, we define the *single policy concentrability coefficient* C^π for policy π as

$$\max_{s,a} \frac{d^\pi(s,a)}{\mu(s,a)} \leq C^\pi. \tag{1.7}$$

For an expert policy π, $C^\pi = 1$ denotes an expert-only dataset whereas $C^\pi > 1$ indicates that the dataset includes spurious samples. For an optimal policy $\pi^\star$, finite $C^\star := C^{\pi^\star}$ is the weakest concentrability requirement, that has only appeared in characterizing the sample complexity of online RL algorithms (Scherrer, 2010). Prior to this, offline RL algorithms required "uniform" coverage such as by assuming a uniform concentrability coefficient $\max_\pi C^\pi \leq C$ (Chen and Jiang, 2019) or uniformly lower bounded data distribution (Sidford et al., 2018a; Agarwal et al., 2019a).

Under this new framework, we further investigate the question on algorithm design: can one develop an algorithm that achieves a minimax optimal rate and also adapts to unknown data composition? We make progress in answering this question by analyzing an offline learning algorithm based on pessimism in the face of uncertainty developed by constructing lower confidence bound (LCB) in the tabular setting. We study finite-sample properties of LCB as well as information-theoretic limits in multi-armed bandits (MAB), contextual bandits (CB), and Markov decision processes (MDPs), with an overview of results presented below.

Multi-armed bandits We start by studying offline learning in the MAB setting, in which we are provided with a dataset of arms and sampled random rewards $\{(a_i, r_i)\}$ and our goal is to find an arm $\hat{a}$ so as to minimize the suboptimality.

A natural option would be to compute the empirical average reward $\hat{r}(a)$ for each arm and then choose the arm with the maximum average reward: $\hat{a} \in \arg\max_a \hat{r}(a)$. Intuitively, however, the empirical average reward can be inaccurate for the arms with a small number of samples, which may result in suboptimal arms getting picked despite having a smaller true mean reward. We confirm this intuition by proving the following theorem.

Theorem 1.3 (Best empirical arm fails, Informal)**.** *There exist bandit problems where the best empirical arm fails to achieve a sub-optimality that decays with sample size N.*

Building on our intuition, we can take a different approach and instead be pessimistic about the rewards of the arms with fewer samples in the dataset. We apply pessimism by penalizing the empirical reward according to the confidence bound, i.e. constructing a lower confidence bound (LCB), and then choose the arm that maximizes the LCB. We pick the following specific LCB construction which is based on Hoeffding's inequality:

$$\hat{a} \in \arg\max \hat{r}(a) - \frac{L}{\sqrt{N(a) \vee 1}}, \tag{1.8}$$

where L is a constant affecting the degree of penalty. We show that, unlike the best empirical arm, the LCB approach of (1.8) is convergent for all data compositions.

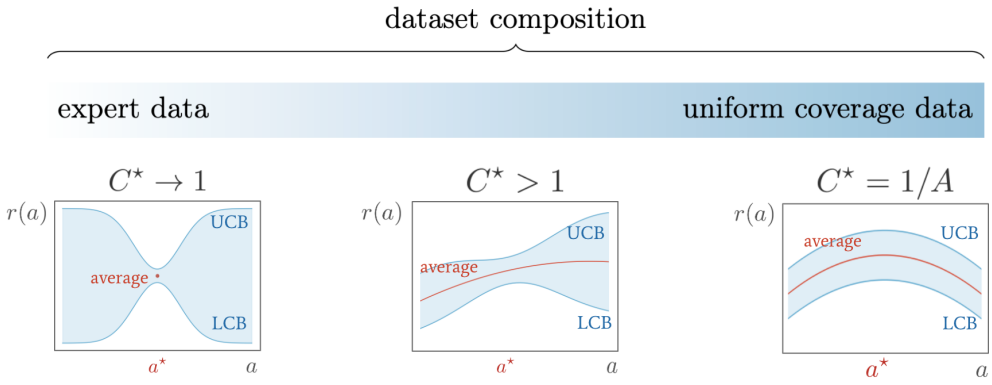

Figure 1.2: Left (expert data): Penalty in LCB eliminates sub-optimal arms with no samples and thus LCB acts similar to imitation learning. Right (uniform coverage data): Best empirical arm and LCB differ by a constant, therefore LCB acts similar to vanilla offline bandits. Middle (mixed data): LCB works well in the middle region while imitation learning and best empirical arm can fail.

Theorem 1.4 (LCB in MAB converges for any data composition, Informal)**.** *For all $C^\star \geq 1$, the suboptimality of the LCB algorithm is $\widetilde{O}\left(\sqrt{\frac{C^\star}{N}}\right)$.*

Figure 1.2 provides further intuition on how the LCB approach works across the dataset composition. We further analyze the information theoretic limits of offline learning in MAB, with an schematic representation of the results provided in Figure 1.3. In summary, we find that while the LCB algorithm works well across the data composition, it cannot achieve the information theoretic limit in the MAB setting, regardless of how one sets the parameter L.

Contextual bandits. Contextual bandits (CB) comprise of S separate MAB problems with an offline dataset of the form $\{(s_i, a_i, r_i)\}$. To extend the LCB algorithm to the CB setting, we simply select the arm with maximum LCB in each state (context):

$$\hat{\pi}(a) = \arg\max \hat{r}(s, a) - \frac{L}{\sqrt{N(s, a) \vee 1}}. \tag{1.9}$$

We analyze the LCB performance upper bound and the information-theoretic limit of offline learning in CB and prove the following theorem.

Theorem 1.5 (LCB is adaptively optimal in CB, Informal)**.** *For all $C^\star \geq 1$ and provided that $S > 1$, the suboptimality of the LCB algorithm for contextual bandits is $\widetilde{O}\left(\sqrt{\frac{S(C^\star - 1)}{N}} + \frac{S}{N}\right)$, which matches the information-theoretic lower bound up to logarithmic factors.*

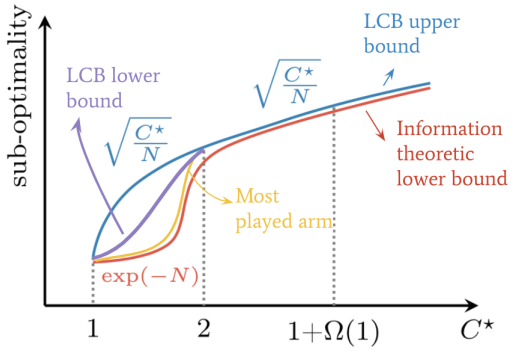

Figure 1.3: Summary of offline learning results in the MAB setting. While LCB achieves a decaying rate for all $C^\star$, the rate matches the information-theoretic lower bound for $C^\star \geq 2$. The case of $C^\star \in [1, 2)$ corresponds to $\mu(a^\star) > 1/2$, i.e. the optimal arm has more than 50% probability in data distribution. If such knowledge is provided, one can choose the most played arm, which nearly achieves the information-theoretic limit with the rate $\exp(-N)$. This is in contrast to the LCB that has a performance lower bound that is only polynomial in N.

The above theorem shows that going beyond the single state case of MAB, the LCB algorithm becomes adaptively optimal. This is due to the fact that the information-theoretic lower bound of offline learning in CB no longer has an exponential convergence rate in the sample size. Figure 1.4 gives a schematic representation of this result. In addition, the LCB performance rate in Theorem 1.6 reveals that as $C^\star$ increases from 1, the sub-optimality rate smoothly transitions from $1/N$, akin to rates in imitation learning, to $1/\sqrt{N}$, akin to rates in vanilla offline RL.

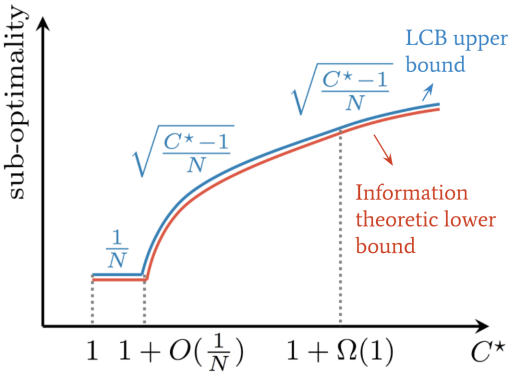

Figure 1.4: Summary of offline learning results in the CB setting. The LCB algorithm achieves adaptive optimal rates (up to logarithmic factors) in CB with at least two contexts.

Infinite-horizon MDPs. For the MDP setting, we incorporate LCB with the vanilla value iteration algorithm (Sutton and Barto, 2018). Let $V : \mathcal{S} \to V_{\max}$ and $Q : \mathcal{S} \times \mathcal{A} \to V_{\max}$ define scalar functions, called value function and Q-function, respectively, with $V_{\max} := 1/(1-\gamma)$ denoting the effective horizon. The vanilla value iteration algorithm repeats the following update until convergence:

$$\begin{aligned} Q(s,a) &\leftarrow r(s,a) + \gamma\, \mathbb{E}_{s' \sim P(\cdot|s,a)}[V(s')], \\ V(s) &\leftarrow \max_a Q(s,a). \end{aligned}$$

The induced greedy policy from the Q-function, i.e. $\pi(s) = \arg\max_a Q(s,a)$ is then returned by the algorithm. We modify the above update by first swapping the unknown rewards and expectation over next states with their empirical counterparts and and then subtracting a penalty from the Q-function to account for the statistical fluctuations of the empirical estimate:

$$\begin{aligned} Q(s,a) &\leftarrow \hat{r}(s,a) + \gamma \hat{\mathbb{E}}_{s' \sim P(\cdot|s,a)}[V(s')] - \frac{L V_{\max}}{\sqrt{m(s,a) \vee 1}}, \\ V(s) &\leftarrow \max_a Q(s,a). \end{aligned}$$

Here, $m(s,a)$ is the number of samples on state-action pair (s,a). We also add two more tricks to the algorithm, namely data splitting and monotonic update, which gives the full value iteration with lower confidence bound (VI-LCB). Similar to the offline bandits, we analyze both the LCB upper bound and information-theoretic lower bounds in the offline RL setting as stated below.

Theorem 1.6 (VI-LCB is almost adaptively optimal, Informal)**.** *For all $C^\star \geq 1$ the sub-optimality of any offline RL algorithm is*

$$\gtrsim \min\left(\frac{1}{1-\gamma}, \sqrt{\frac{S(C^\star - 1)}{(1-\gamma)^3 N}} + \frac{S}{(1-\gamma)N}\right),$$

and VI-LCB nearly achieves the above rate.

Figure 1.5 summarizes our findings for the offline RL setting. As the figure shows, VI-LCB algorithm achieves adaptive optimal rates for $C^\star \approx 1$ and $C^\star = 1 + \Omega(1)$. We have a gap in the middling region between the LCB upper bound and information-theoretic lower bound. However, we conjecture that this gap is due to our upper bound analysis and the VI-LCB algorithm enjoys adaptive optimality. We verify the conjecture in a simple example and discuss challenges faced in the analysis and potential methods to address the difficulty.

Figure 1.6 provides intuition on the type of analysis required for showing the fast rates and adaptive optimality of LCB. Consider the offline learning problem in the MAB setting with the true rewards plotted in blue in Figure 1.6. If one ensures that the empirical rewards closely estimate the true rewards, then the algorithm is likely to return the optimal arm, as

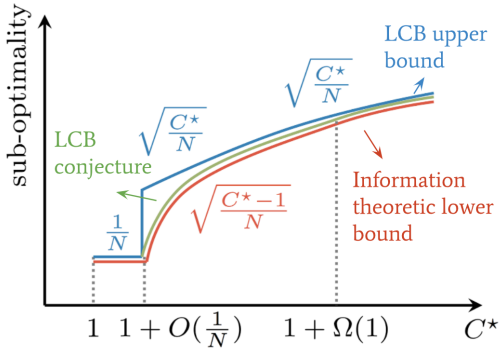

Figure 1.5: Summary of offline learning results in the MDP setting. The VI-LCB algorithm nearly achieves adaptive optimal rates.

is the case in Figure 1.6(a). However, this requirement is rather stringent and our analysis reveals that this requirement yields the slow rate of $1/\sqrt{N}$. It is possible for the pessimistic reward estimates to not be accurate but be accurate enough so that the algorithm picks the optimal arm. An example of this case is shown in Figure 1.6(b), which is the basis of our analysis in the CB setting for establishing the adaptive optimality of LCB (Section 3.8). Lastly, the case where the algorithm returns a slightly sub-optimal arm within the error tolerance is also acceptable. Accounting for this case yields an even tighter rate, which is the basis of the analysis of the example in which we verify the conjecture (Section 3.10.6).

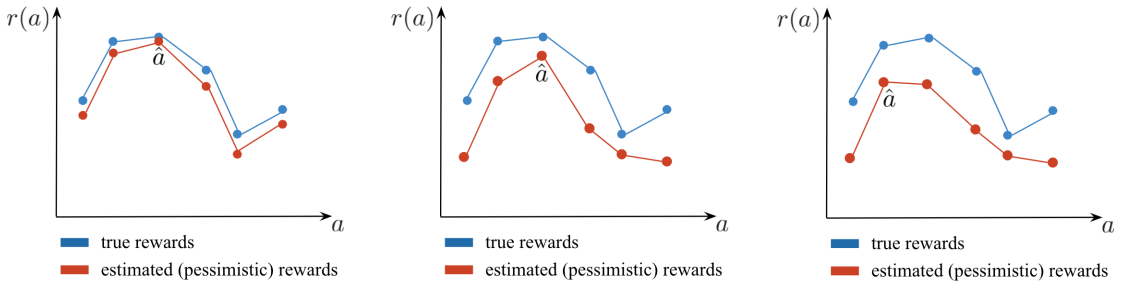

Figure 1.6: LCB analysis techniques. A reward or values guarantee can give the slow rate of $1/\sqrt{N}$ whereas a policy-based guarantee can give rates faster than $1/\sqrt{N}$.

1.2.2 Decision-making during interactions

The interactive mode, often called online RL, is considered the classical setting in reinforcement learning, where the agent aims at learning a policy with maximum cumulative reward

while interacting with the environment. During interactions, the agents needs to balance between exploring the environment further (to discover strategies with higher reward) and exploiting its current knowledge of the environment.[3] Effective exploration remains challenging, particularly in high dimensional environments that provide little feedback. In such cases, successful exploration methods often rely on manually designing dense rewards which are reliant on the problem-specific domain knowledge (Brockman et al., 2016).

A common strategy to conduct exploration is based on *intrinsic motivation*. Literature provide justifications for using intrinsic motivation from different angles, often related to novelty seeking. One angle is inspired by an analogy with neuroscience studies (Chentanez et al., 2004) showing that dopamine, a neuromodulator related to reward learning, plays a key role for intrinsic motivational behavior control associated with novelty and exploration (Dayan and Balleine, 2002; Kakade and Dayan, 2002). Another angle is the optimism in the face of uncertainty which is based on by statistical learning (Yang et al., 2020b; Azar et al., 2017). Other angles include curiosity (Pathak et al., 2017), gaining information (Russo and Van Roy, 2014; Nikolov et al., 2018), and empowerment (Klyubin et al., 2005a,b).

Intrinsic motivation is often done by adding an intrinsic reward (bonus) is added to the extrinsic reward from the environment to guide exploration. Provable methods of this category are usually based on constructing upper confidence bounds via concentration inequalities such as Hoeffding or Bernstein inequalities. However, it remains unclear how to construct confidence bounds for nonlinear function approximators such as neural network. Furthermore, Bernstein-based bonus achieves near-optimal performance in tabular setting by exploiting value function variance information and problem structure (Zanette and Brunskill, 2019) but, computing variance of the value function is difficult for nonlinear function. Exploration methods that can be combined with nonlinear function approximation such as random network distillation (Burda et al., 2018b), pseudo-counts (Bellemare et al., 2016), and curiosity (Pathak et al., 2017) are still sample inefficient and can suffer from issues such as "noisy TV", in which the agent is distracted by aleatoric uncertainty, forgetting, or getting stuck in locally optimal policies (Agarwal et al., 2020a).

Adaptive regularization to guide exploration. In the work (Zhang et al., 2021b), we pose the exploration and exploitation dilemma by adding an adaptive regularizer to the standard RL objective to guide exploration. In particular, viewing that the second objective in (1.5) as a function of d^π, we define the new regularized objective as

$$\max_{d^\pi \in \mathcal{K}} L_k(d^\pi) = \max_{d^\pi \in \mathcal{K}} \underbrace{\frac{1}{1-\gamma} \mathbb{E}_{s,a\sim d^\pi}[r(s,a)]}_{\text{exploitation}} + \tau_k \underbrace{R(d^\pi; \{d^{\pi_i}\}_{i=1}^k)}_{\text{exploration}}. \tag{1.10}$$

[3]Exploration-exploitation tradeoff appears when one is concerned with a cumulative performance. This is different from modalities of interactive RL such as explore-then-commit, in which the goal is to move quickly towards the best policy, and pure exploration, in which the goal is to fully explore the environment so as to later obtain good policies given any reward function without acquiring additional samples.

Here, d^π is the occupancy of the current policy, belonging to a set $\mathcal{K}$ of all valid policy occupancies in the environment, and $d^{\pi_1}, \ldots, d^{\pi_k}$ are occupancies are previous policies. The above framework admits many popular exploration methods such as the ones that depend on prior visitation counts (Bellemare et al., 2016; Zhang et al., 2020d) or intrinsic rewards that originate from entropy-based exploration (Zhang et al., 2021a).

We next propose a particular regularizer to conduct exploration. Define *policy cover* $\rho^k_{\text{cov}}(s,a)$ to be an average of previous policy occupancies, capturing the already explored regions. Then, we design our regularizer to *maximize deviation* (MADE) from policy cover:

$$\underbrace{R(d^\pi; \{d^{\pi_i}\}_{i=1}^k)}_{\text{exploration}} := \sum_{s,a} \sqrt{\frac{d^\pi(s,a)}{\rho^k_{\text{cov}}(s,a)}}. \tag{1.11}$$

The specific regularizer above has several favorable properties: (1) it is concave in d^π, (2) its optimum has the form $d^\pi(s,a) = \frac{1}{\rho^k_{\text{cov}}(s,a)}$, thus the next policy occupies the less explored regions, and (3) applying the Frank Wolfe algorithm to solve the constrained optimization problem (1.11) implies the following intrinsic reward:

$$\frac{\tau_k}{\sqrt{d^{\pi_{\text{mix}},k}(s,a)\rho^k_{\text{cov}}(s,a)}}, \tag{1.12}$$

where $d^{\pi_{\text{mix}},k}$ is roughly an average of prior policy occupancies with geometrically distributed weights and $\rho^k_{\text{cov}}(s,a)$ is prior policy occupancy average. For a practical implementation, we substitute $d^{\pi_{\text{mix}},k}$ and $\rho^k_{\text{cov}}(s,a)$ with their empirical estimates.

Taking a closer look in the special case of tabular parameterization, we have $\hat{\rho}^k_{\text{cov}}(s,a)$ proportional to the prior visitation count $N_k(s,a)$. Thus, the the intrinsic reward (1.12) applies a simple modification to the count-based methods:

$$\propto \underbrace{\frac{1}{\sqrt{d^{\pi_{\text{mix}},k}(s,a)}}}_{\text{correction term}} \underbrace{\frac{\tau_k}{\sqrt{N_k(s,a)}}}_{\text{count-based bonus}} \tag{1.13}$$

To combine the intrinsic reward (1.12) with deep neural network parameterization, one can use any method used for approximating the count-based bonus such as RND (Burda et al., 2018b) and multiply it by the correction term. In our experiments, we approximate $d^{\pi_{\text{mix}},k}$ using a variational autoencoder trained over a fixed-length dataset of prior interactions, often referred to as the replay buffer (Zhang and Sutton, 2017; Lillicrap et al., 2015; Andrychowicz et al., 2017).

Empirical findings. We conduct experiments both in tabular setting and online RL benchmark tasks to evaluate the performance of the MADE exploration method. A summary of our findings is presented below.

- MADE exploration improves over count-only exploration (Hoeffding bonus) in tabular setting, acting similar to the information-theoretically optimal Bernstein bonus without requiring value function variance estimation. We observe this performance regardless of the type of RL algorithm combined with MADE exploration.
- The regularization in MADE objective seems to offer improvements on optimization landscape, increasing the rate of convergence to the optimum. We observe this phenomenon in the tabular chain MDP of Agarwal et al. (2019b), where the optimization rate of policy gradient method is improved exponentially over the vanilla version without regularization.
- Implemented with deep neural networks, MADE exploration method beats state-of-the-art methods by a large margin in a variety of locomotion and navigation benchmark tasks, both when paired with model-free or model-based RL algorithms.

Chapter 2

Learning to Predict under Long-term Dependencies

In this chapter, we take a look at the prediction problem in a popular sequential environment called linear dynamical systems (LDS). Predictive models based on LDS have been successfully used in a wide range of applications with a history of more than half a century. Example applications in AI-related areas range from control systems and robotics (Durrant-Whyte and Bailey, 2006) to natural language processing (Belanger and Kakade, 2015), healthcare (Parker et al., 1999), and computer vision (Chen, 2011; Coskun et al., 2017). Other applications are found throughout the physical, biological, and social sciences in areas such as econometrics, ecology, and climate science.

Recall that the evolution of a discrete-time LDS is described by the following state-space model with $t \geq 1$:

$$\begin{aligned} h_{t+1} &= Ah_t + Bx_t + \eta_t, \\ y_t &= Ch_t + Dx_t + \zeta_t, \end{aligned}$$

where h_t are the latent states, x_t are the inputs, y_t are the observations, and η_t and ζ_t are process and measurement noise, respectively.

When the system parameters are known, the optimal linear predictor is the Kalman filter. When they are unknown, a common approach for prediction is to first estimate the parameters of a Kalman filter and then use them to predict system evolution. Direct parameter estimation usually involves solving a non-convex optimization problem, such as in the expectation maximization (EM) algorithm, whose theoretical guarantees may be difficult (Yu et al., 2018). Several recent works have studied finite-sample theoretical properties of LDS identification. For fully observed LDS, it has been shown that system identification is possible without a strict stability ($\rho(A) < 1$) assumption, where $\rho(A)$ is the spectral radius of A (Simchowitz et al., 2018; Sarkar and Rakhlin, 2018; Faradonbeh et al., 2018). For partially observed LDS, methods such as gradient descent (Hardt et al., 2018) and subspace identification (Tsiamis and Pappas, 2019) are developed, whose performances degrade polynomially when $\rho(A)$ is close to one.

We focus on constructing *predictors* of an LDS without identifying the parameters. In the case of a stochastic LDS, the recent work of Tsiamis and Pappas (2020) is most related to our question. Their method performs linear regression over a fixed-length lookback window to predict the next observation y_t given its causal history. Without using a mixing-time argument, Tsiamis and Pappas (2020) showed logarithmic regret with respect to the Kalman filter in hindsight even when the system is marginally stable ($\rho(A) \leq 1$). However, the prediction performance deteriorates if the true Kalman filter exhibits *long-term forecast memory*.

To illustrate the notion of forecast memory, we recall the recursive form of the (stationary) Kalman filter for $1 \leq t \leq T$, where T is the final horizon (Kailath et al., 2000, chap. 9):

$$\hat{h}_{t+1|t} = A\hat{h}_{t|t-1} + Bx_t + K(y_t - C\hat{h}_{t|t-1} - Dx_t) \tag{2.1}$$

$$= (A - KC)\hat{h}_{t|t-1} + Ky_t + (B - KD)x_t, \tag{2.2}$$

where $\hat{h}_{t|t-1}$ denotes the optimal linear predictor of h_t given all the observations $y_1, y_2, \ldots, y_{t-1}$ and inputs $x_1, x_2, \ldots, x_{t-1}$. The matrix K is called the (predictive) Kalman gain.[1] The Kalman predictor of y_t given $y_1, y_2, \ldots, y_{t-1}$ and $x_1, x_2, \ldots, x_t$, denoted by $\hat{y}_{t|t-1}$, is $C\hat{h}_{t|t-1} + Dx_t$. Assume that $\hat{h}_{1|0} = 0$. By expanding Equation (2.2), we obtain

$$m_t \triangleq \hat{y}_{t|t-1} = \sum_{i=1}^{t-1} CG^{t-i-1}Ky_i + \sum_{i=1}^{t-1} CG^{t-i-1}(B - KD)x_i + Dx_t, \tag{2.3}$$

where $G = A - KC$. In an LDS, the transition matrix A controls how fast the process mixes—i.e., how fast the marginal distribution of y_t becomes independent of y_1. However, it is G that controls how long the *forecast* memory is. Indeed, it was shown in Kailath et al. (2000, chap. 14) that if the spectral radius $\rho(G)$ is close to one, then the performance of a linear predictor that uses only y_{t-k} to y_{t-1} for fixed k in predicting y_t would be substantially worse than that of a predictor that uses all information y_1 up to y_{t-1} as $t \to \infty$. Conceivably, the sample size required by the algorithm of Tsiamis and Pappas (2020) explodes to infinity as $\rho(G) \to 1$, since the predictor uses a fixed-length lookback window to conduct linear regression.

The primary reason to focus on long-term forecast memory is the ubiquity of long-term dependence in real applications, where it is often the case that not all state variables change according to a similar timescale[2] (Chatterjee and Russell, 2010). For example, in a temporal model of the cardiovascular system, arterial elasticity changes on a timescale of years, while the contraction state of the heart muscles changes on a timescale of milliseconds.

[1] One can interpret the Kalman filter Equation (2.1) as linear combinations of optimal predictor given existing data $A\hat{h}_{t|t-1}$, known drift Bx_t, and amplified innovation $K(y_t - C\hat{h}_{t|t-1} - Dx_t)$, where the term $y_t - C\hat{h}_{t|t-1} - Dx_t$, called the *innovation* of process y_t, measures how much additional information y_t brings compared to the known information of observations up to y_{t-1}.

[2] Indeed, a common practice is to set the timescale to be small enough to handle the fastest-changing variables.

Designing provably computationally and statistically efficient algorithms in the presence of long-term forecast memory is challenging, and in some cases, impossible. A related problem studied in the literature is the prediction of auto-regressive model with order infinity: AR(∞). Without imposing structural assumptions on the coefficients of an AR(∞) model, there is no hope to guarantee vanishing prediction error. One common approach to obtain a smaller representation is to make an exponential forgetting assumption to justify finite-memory truncation. This approach has been used in approximating AR(∞) with decaying coefficients (Goldenshluger and Zeevi, 2001), LDS identification (Hardt et al., 2018), and designing predictive models for LDS (Tsiamis and Pappas, 2020; Kozdoba et al., 2019). Inevitably, the performance of these methods degrade by either losing long-term dependence information or requiring very large sample complexity as $\rho(G)$ (and sometimes, $\rho(A)$) gets closer to one.

However, the Kalman predictor in (2.3) does seem to have a structure and in particular, the coefficients are geometric in G, which gives us hope to exploit it. Our main contributions are the following:

1. Generalized Kolmogorov width and spectral methods: We analyze the *generalized Kolmogorov width*, defined in Section 2.4.1, of the Kalman filter coefficient set. In Theorem 2.2, we show that when the matrix G is diagonalizable with *real* eigenvalues, the Kalman filter coefficients can be approximated by a linear combination of $\text{polylog}\,(T)$ *fixed known* filters with $1/\text{poly}\,(T)$ error. It then motivates the algorithm design of linear regression based on the *transformed* features, where we first transform the observations $y_{1:t}$ and inputs $x_{1:t}$ for $1 \leq t \leq T$ via these fixed filters. In some sense, we use the transformed features to achieve a good bias-variance trade-off: the small number of features guarantees small variance and the generalized Kolmogorov width bound guarantees small bias. We show that the fixed known filters can be computed efficiently via spectral methods. Hence, we choose spectral LDS improper predictor (SLIP) as the name for our algorithm.

2. Difficulty of going beyond real eigenvalues: We show in Theorem 2.2 that if the dimension of matrix G in (2.3) is at least 2, then without assuming real eigenvalues one has to use at least $\Omega(T)$ filters to approximate an arbitrary Kalman filter. In other words, the Kalman filter coefficient set is very difficult to approximate via linear subspaces in general. This suggests some inherent difficulty of constructing provable algorithms for prediction in an arbitrary LDS.

3. Logarithmic regret uniformly for $\boldsymbol{\rho(G) \leq 1, \rho(A) \leq 1}$: When $\rho(A)$ or $\rho(G)$ is equal to one the process does not mix and common assumptions regarding boundedness, concentration, or stationarity do not hold. Recently, Mendelson (2014) showed that such assumptions are not required and learning is possible under a milder assumption referred to as the *small-ball* condition. In Theorem 2.1, we leverage this idea as well as results on self-normalizing martingales and show a logarithmic regret bound for our algorithm uniformly for $\rho(G) \leq 1$ and $\rho(A) \leq 1$. A roadmap to our regret analysis method is provided in Section 2.5.

4. Experimental results: We demonstrate in simulations that our algorithm performs better than the state-of-the-art in LDS prediction algorithms. In Section 2.6, we compare the

performance of our algorithm to wave filtering (Hazan et al., 2017) and truncated filtering (Tsiamis and Pappas, 2020).

2.1 Related work

Adaptive filtering algorithms are classical methods for predicting observations without the intermediate step of system identification (Ljung, 1978; Fuller and Hasza, 1980, 1981; Wei, 1987; Lai and Ying, 1991; Lorentz et al., 1996). However, finite-sample performance and regret analysis with respect to optimal filters are typically not studied in the classical literature. From a machine learning perspective, finite-sample guarantees are critical for comparing the accuracy and sample efficiency of different algorithms. In designing algorithms and analyses for learning from sequential data, it is common to use mixing-time arguments (Yu, 1994). These arguments justify finite-memory truncation (Hardt et al., 2018; Goldenshluger and Zeevi, 2001) and support generalization bounds analogous to those in i.i.d. data (Mohri and Rostamizadeh, 2009; Kuznetsov and Mohri, 2017). An obvious drawback of mixing-time arguments is that the error bounds degrade with increasing mixing time. Several recent works established that identification is possible for systems that do not mix (Simchowitz et al., 2018; Faradonbeh et al., 2018; Simchowitz et al., 2019). For the problem of the linear quadratic regulator, where the state is fully observed, several results provided finite-sample regret bounds (Faradonbeh et al., 2017; Ouyang et al., 2017; Dean et al., 2018; Abeille and Lazaric, 2018; Mania et al., 2019; Simchowitz and Foster, 2020).

For prediction without LDS identification, Hazan et al. (2017, 2018) have proposed algorithms for the case of bounded adversarial noise. Similar to our work, they use spectral methods for deriving features. However, the spectral method is applied on a different set and connections with k-width and difficulty of approximation for the non-diagonalizable case are not studied. Moreover, the regret bounds are computed with respect to a certain fixed family of filters and competing with the Kalman filter is left as an open problem. Indeed, the predictor for general LDS proposed by Hazan et al. (2018) without the real eigenvalue assumption only uses a fixed lookback window. Furthermore, the feature norms are of order poly (T) in our formulation, which makes a naive application of online convex optimization theorems (Hazan, 2019) fail to achieve a sublinear regret.

We focus on a more challenging problem of learning to predict in the presence of unbounded stochastic noise and long-term memory, where the observation norm grows over time. The most related to our work are the recent works of Tsiamis and Pappas (2020) and Ghai et al. (2020), where the performance of an algorithm based on a finite lookback window is shown to achieve logarithmic regret with respect to the Kalman filter. However, the performance of this algorithm degrades as the forecast memory increases. In fact, this algorithm can be viewed as a special case of our algorithm where the fixed filters are chosen to be standard basis vectors.

We investigate the possibility of conducting tight convex relaxation of the Kalman predictive model by defining a notion that generalizes Kolmogorov width. The Kolmogorov

width is a notion from approximation theory that measures how well a set can be approximated by a low-dimensional linear subspace (Pinkus, 2012). Kolmogorov width has been used in a variety of problems such as minimax risk bounds for truncated series estimators (Donoho et al., 1990; Javanmard and Zhang, 2012), minimax rates for matrix estimation (Ma and Wu, 2015), density estimation (Hasminskii et al., 1990), hypothesis testing (Wei and Wainwright, 2020; Wei et al., 2020), and compressed sensing (Donoho, 2006). In Section 2.4, we present a generalization of Kolmogorov width, which facilitates measuring the convex relaxation approximation error.

2.2 Preliminaries and problem formulation

2.2.1 Problem statement

We consider the problem of predicting observations generated by the following linear dynamical system with inputs $x_t \in \mathbb{R}^n$, observations $y_t \in \mathbb{R}^m$, and latent states $h_t \in \mathbb{R}^d$:

$$\begin{aligned} h_{t+1} &= Ah_t + Bx_t + \eta_t, \\ y_t &= Ch_t + Dx_t + \zeta_t, \end{aligned} \tag{2.4}$$

where A, B, C, and D are matrices of appropriate dimensions. The sequences $\eta_t \in \mathbb{R}^d$ (process noise) and $\zeta_t \in \mathbb{R}^m$ (measurement noise) are assumed to be zero-mean, i.i.d. random vectors with covariance matrices Q and R, respectively. For presentation simplicity, we assume that η_t and ζ_t are Gaussian; extension of our regret analysis to sub-Gaussian and hypercontractive noise is straightforward. We assume that the discrete Riccati equation of the Kalman filter for the state covariance has a solution P and the initial state starts at this stationary covariance. This assumption ensures the existence of the stationary Kalman filter with stationary gain K; see Kailath et al. (2000) for details.

Define the observation matrix $\mathcal{O}_t$ and the control matrix $\mathcal{C}_t$ of a stationary Kalman filter as

$$\begin{aligned} \mathcal{O}_t &= \begin{bmatrix} CG^{t-1}K & CG^{t-3}K & \dots & CK \end{bmatrix}, \\ \mathcal{C}_t &= \begin{bmatrix} CG^{t-1}(B-KD) & CG^{t-3}(B-KD) & \dots & C(B-KD) \end{bmatrix}. \end{aligned} \tag{2.5}$$

where $G = A - KC$ is called the closed-loop matrix. The Kalman predictor (2.3) can be written as

$$m_{t+1} = \mathcal{O}_t y_{1:t} + \mathcal{C}_t x_{1:t} + Dx_{t+1}, \tag{2.6}$$

The prediction error $e_t = y_t - m_t$, also called the *innovation*, is zero-mean with a stationary covariance V. Our goal is to design an algorithm $\hat{m}_t(y_{1:t-1}, x_{1:t})$ such that the following regret

$$\text{Regret}(T) \triangleq \sum_{t=1}^{T} \|y_t - \hat{m}_t\|_2^2 - \|y_t - m_t\|_2^2 \tag{2.7}$$

is bounded by polylog(T) with high probability.

2.2.2 Improper learning

Most existing algorithms for LDS prediction include a preliminary system identification step, in which system parameters are first estimated from data, followed by the Kalman filter. However, the loss function (such as squared loss) over system parameters is non-convex, for which methods based on heuristics such as EM and subspace identification are commonly used. Instead, we aspire to an algorithm that optimizes a convex loss function for which theoretical guarantees of convergence and sample complexity analysis are possible. This motivates developing an algorithm based on *improper learning*.

Instead of directly learning the model parameters in a hypothesis class $\mathcal{H}$, improper learning methods reparameterize and learn over a different class $\widetilde{\mathcal{H}}$. For example in system (2.4), proper learning hypothesis class $\mathcal{H}$ contains possible values for parameters A, B, C, D, Q and R. Improper learning is used for statistical or computational considerations when the original hypothesis class is difficult to learn. The class $\widetilde{\mathcal{H}}$ is often a *relaxation*: it is chosen in a way that is easier to optimize and more computationally efficient while being close to the original hypothesis class. Improper learning has been used to circumvent the proper learning lower bounds (Foster et al., 2018).

In this paper, we use improper learning to conduct a tight *convex relaxation*, i.e. we slightly overparameterize the LDS predictive model in such a way that the resulting loss function is convex. Designing an overparameterized improper learning class requires care as using a small number of parameters may result in a large bias whereas using too many parameters may result in high variance. Section 2.4.3 presents our overparameterization approach based on spectral methods that enjoys a small approximation error with relatively few parameters.

2.2.3 Additional Notation

We define $M = (R_\Theta, m, \gamma, \kappa, \beta, \gamma, \delta)$ to be a shorthand for the PAC bound parameters (defined in Theorem 2.1). Given a function $f : \mathbb{N} \to R$, we write $x \lesssim_M f(T), x \asymp_M f(T)$ to specify the dependency only on the horizon T.

2.2.4 Systems with long forecast memory

As discussed before, system (2.4) exhibits long forecast memory when $\rho(G)$ is close to one. The closed-loop matrix G itself is related to parameters A, C, Q, and R. In the following example, we discuss when long forecast memory is instantiated in a scalar dynamical system.

Example 1. *Consider system (2.4) with $d = m = 1$. The following holds for a stationary Kalman filter*

$$KC = \frac{AC^2P^+}{C^2P^+ + R} \Rightarrow 0 \leq KC \leq A \qquad \text{for } d = m = 1,$$

where P^+ is the variance of state predictions $\hat{h}_{t|t-1}$ (Kailath et al., 2000). The above constraint yields $G = A - KC \leq A$, which implies that the forecast memory can only be long in systems that mix slowly. We write

$$G = A\big(1 - \frac{C^2 P^+}{C^2 P^+ + R}\big), \qquad \text{for } d = m = 1.$$

The above equation suggests if $R \gg C^2 P^+$, then G is close to A. In words, linear dynamical systems with small observed signal to noise ratio $C/\sqrt{R}$ have long forecast memory, provided that they mix slowly.

Another parameter that affects the forecast memory of a system is the process noise variance Q. When Q is small and A is close to one, latent state h_t is almost constant. In this setting, the observations in the distant past are informative on h_t and therefore should be considered when making predictions.

In multi-dimensional systems, the chance of encountering a system with long forecast memory is much higher as it suffices for only one variable or direction to exhibit long forecast memory. Systems represented in the discrete-time form of Equation (2.4) are often obtained by discretizing differential equations and continuous dynamical systems, for which choosing a small time step results in a better approximation. However, reducing the time step directly increases the forecast memory. These types of issues has motivated a large body of research on alternative methods such as continuous models (Nodelman et al., 2002) and adaptive time steps (Aleks et al., 2009). It is therefore desirable to have algorithms whose performance is not affected by the choice of time step, which is one of our goals in this paper.

2.3 SLIP: Spectral LDS improper predictor

In this section, we present the SLIP algorithm and the main regret theorem. The derivation of the algorithm and the sketch for regret analysis are respectively provided in Section 2.4 and Section 2.5.

Algorithm 1 presents a pseudocode for the SLIP algorithm. Our algorithm is based on an online regularized least squares and a linear predictor $\hat{m}_t = \hat{\Theta}^{(t)} f_t$, where f_t is an l-dimensional vector of features and $\hat{\Theta}^{(t)} \in \mathbb{R}^{m \times l}$ is a parameter matrix. The features are constructed from past observations and inputs using eigenvectors of a particular $T \times T$ Hankel matrix with entries

$$H_{ij} = \frac{1 + (-1)^{i+j}}{2(i + j - 1)}, \quad 1 \leq i, j \leq T. \tag{2.8}$$

Let $\phi_1, \ldots, \phi_k$ for $k \leq T$ be the top k eigenvectors of matrix H, to which we refer as *spectral filters*. At every time step, we obtain our feature vector by concatenating the current input x_t to k *output features* based on $y_{1:t-1}$ and k *input features* based on $x_{1:t-1}$. More specifically, we have

$$\begin{aligned}\widetilde{y}_{t-1}(j) &\triangleq (\phi_j^\top(t-1:1)\otimes I_m)y_{1:t-1} = \phi_j(1)y_{t-1}+\cdots+\phi_j(t-1)y_1 \quad \text{(output features)},\\ \widetilde{x}_{t-1}(j) &\triangleq (\phi_j^\top(t-1:1)\otimes I_n)x_{1:t-1} = \phi_j(1)x_{t-1}+\cdots+\phi_j(t-1)x_1 \quad \text{(input features)},\end{aligned} \tag{2.9}$$

for $j \in \{1,\dots,k\}$, resulting in a feature vector f_t with dimension $l = mk + nk + n$. Upon receiving a new observation, the parameter matrix is updated by minimizing the regularized loss

$$\sum_{i=1}^{t} \|\hat{\Theta} f_t - y_t\|^2 + \alpha \|\hat{\Theta}\|_2^2,$$

for $\alpha > 0$, which yields the following update rule

$$\hat{\Theta}^{(t+1)} = \Big(\sum_{i=1}^{t} y_i f_i^\top\Big)\Big(\sum_{i=1}^{t} f_i f_i^\top + \alpha I_l\Big)^{-1}. \tag{2.10}$$

Algorithm 1 SLIP: **S**pectral **L**DS **I**mproper **P**redictor

Inputs: Time horizon T, number of filters k, regularization parameter α, input dimension n,
observation dimension m.
Output: One-step-ahead predictions $\hat{m}_t(x_{1:t}, y_{1:t-1})$.

Compute the top k eigenvectors $\{\phi_j\}_{j=1}^k$ of matrix H with elements

$$H_{ij} = \frac{(-1)^{i+j}+1}{2(i+j-1)}, \quad 1 \le i,j \le T.$$

Set vectors $\psi_i = [\phi_1(i),\dots,\phi_k(i)]^\top$ for $i \in \{1,\dots,T\}$, where $\phi_j(i)$ is the i-th element of ϕ_j.
Initialize $\hat{\Theta}^{(1)} \in \mathbb{R}^{m\times l}$ with $l = (n+m)k + n$.
for $t = 1,\dots,T$ **do**
Set $\Psi_{t-1} = [\psi_{t-1},\dots,\psi_1]$, where $\Psi_0 = 0_k$.
Set $x_{1:t-1} = [x_1^\top,\dots,x_{t-1}^\top]^\top, y_{1:t-1} = [y_1^\top,\dots,y_{t-1}^\top]^\top$, $x_{1:0} = 0_n, y_{1:0} \triangleq 0_m$.
Compute l-dimensional feature vector f_t:

$$f_t = \begin{bmatrix}\widetilde{y}_{t-1}\\ \widetilde{x}_{t-1}\\ x_t\end{bmatrix} = \begin{bmatrix}(\Psi_{t-1}\otimes I_m)y_{1:t-1}\\ (\Psi_{t-1}\otimes I_n)x_{1:t-1}\\ x_t\end{bmatrix}.$$

Predict $\hat{m}_t = \hat{\Theta}^{(t)} f_t$.
Observe y_t and update parameters $\hat{\Theta}^{(t+1)} = \big(\sum_{i=1}^t y_i f_i^\top\big)\big(\sum_{i=1}^t f_i f_i^\top + \alpha I_l\big)^{-1}$.

Importantly, Algorithm 1 requires no knowledge of the system parameters, noise covariance, or state dimension and the predictive model is learned online only through sequences of inputs and observations. Note that the spectral filters are computed by conducting a single eigendecomposition and are fixed throughout the algorithm; matrix Ψ_t merely selects certain elements of spectral filters used for constructing features. Computing eigenvectors when T is large is possible by solving the corresponding second-order Sturm-Liouville equation, which allows using efficient ordinary differential equation solvers; see Hazan et al. (2017) for details.

The next theorem analyzes the regret achieved by the SLIP algorithm. A proof sketch of the theorem is provided in Section 2.5 and a complete proof is deferred to Section 2.8.8.

Theorem 2.1. (Regret of the SLIP algorithm) *Consider system (2.4) without inputs with initial state covariance equal to the stationary covariance P. Let m_t be the predictions made by the best linear predictor (Kalman filter) and $\hat{m}_t$ be the predictions made by Algorithm 1. Fix the failure probability $\delta > 0$ and make the following assumptions:*

(i) There exists a finite R_Θ that $\|C\|_2, \|P\|_2, \|Q\|_2, \|R\|_2, \|V\|_2 \leq R_\Theta$ and $\|\mathcal{O}_t\|_2 \leq R_\Theta t^\beta$ for a bounded constant $\beta \geq 0$. Let κ be the maximum condition number of R and Q.

(ii) The system is marginally stable with $\rho(A) \leq 1$ and $\|A^t\|_2 \leq \gamma t^{\log(\gamma)}$ for a bounded constant $\gamma \geq 1$. Furthermore, the closed-loop matrix G is diagonalizable with real eigenvalues.

(iii) The regularization parameter α and the number of filters k satisfy the following

$$k \asymp \log^2(T)\,\mathrm{polylog}(m, \gamma, R_\Theta, \frac{1}{\delta}), \qquad \alpha \asymp \frac{1}{R_\Theta k T^\beta}.$$

(iv) There exists $s \lesssim_{R_\Theta, m, \gamma, \beta, \delta} t/(k \log k)$ and t_0 such that for all $t \geq t_0$

$$t\Omega_{s/2}(A; \psi) - \Omega_{t+1}(A; \psi) \succeq 0. \tag{2.11}$$

$\Omega_t(A; \psi)$ is called the filter quadratic function *of ψ with respect to A defined as*

$$\begin{aligned}\Omega_t(A; \psi) = (\psi_1^{(d)})(\psi_1^{(d)})^\top + (\psi_2^{(d)} + \psi_1^{(d)} A)(\psi_2^{(d)} + \psi_1^{(d)} A)^\top + \dots \\ + (\psi_{t-1}^{(d)} + \dots + \psi_1^{(d)} A^{t-2})(\psi_{t-1}^{(d)} + \dots + \psi_1^{(d)} A^{t-2})^\top\end{aligned}$$

where $\psi_i^{(d)} = [\phi_1(i), \dots, \phi_k(i)]^\top \otimes I_d$.

Then, for all $T \geq \max\{10, t_0\}$, the following holds with probability at least $1 - \delta$,

$$\mathrm{Regret}(T) \leq \mathrm{polylog}(T, \gamma, \frac{1}{\delta})\kappa\,\mathrm{poly}(R_\Theta, \beta, m).$$

Theorem 2.1 states that if G is diagonalizable with real eigenvalues, provided that the number of filters $k \asymp_M \log^2(T)$, the regret is $\mathrm{polylog}(T)$ with high probability and the regret bound is independent of both transition matrix spectral radius $\rho(A)$ (related to mixing rate) and closed-loop matrix spectral radius $\rho(G)$ (related to forecast memory).

Remark 2.1. Note that for any matrix A, there exists a constant $\gamma \geq 1$ such that $\|A^t\|_2 \leq \gamma t^{\log(\gamma)}$ (Kozyakin, 2009). We justify our assumption on diagonalizable G with real eigenvalues in the following section. The filter quadratic condition is easily verified for $s > 2(k+1)$ and $t_0 \gtrsim_{R_\Theta,m,\gamma,\beta,\delta} k^2 \log(k)$ for all A with $\rho(A) \leq 1$ for the filters corresponding to truncated observations (a.k.a. basis vectors) such as in Tsiamis and Pappas (2020). When A is symmetric, this condition can be further simplified to $t\Omega_{s/2}(D;\psi) - \Omega_{t+1}(D;\psi) \succeq 0$ for all diagonal matrices D with $|D_{ii}| \leq 1$.

2.4 Approximation error: Generalized Kolmogorov width

2.4.1 Width of a subset

The SLIP algorithm is based on approximating the Kalman predictive model. In this section, we start by introducing a generalization of *Kolmogorov k-width of a subset*, which is a criterion to assess the quality of a function approximation method. We then present our approximation technique which gives the SLIP algorithm.

Definition 2.1. (Generalized Kolmogorov k-width) *Let W be a subset in a normed linear space with norm $\|.\|$ whose elements are $d \times n$ matrices. Given $d \times n$ matrices $u_1, \dots, u_k$ for $k \geq 1$, let*

$$U(u_1, \dots, u_k) \triangleq \left\{ y \;\middle|\; y = \sum_{i=1}^{k} a_i u_i,\ \forall a_i \in \mathbb{R}^{d \times d} \right\}$$

be the subset constructed by linear combinations of $u_1, \dots, u_k$ with coefficient matrices $a_1, \dots, a_k$. For a fixed $k \geq 1$, denote by $\mathcal{U}_k$ the set of $U(u_1, \dots, u_k)$ for all possible choices of $u_1, \dots, u_k$:

$$\mathcal{U}_k \triangleq \left\{ U(u_1, \dots, u_k) \;\middle|\; \forall u_i \in \mathbb{R}^{d \times n} \right\}.$$

The generalized k-width of W is defined as

$$d_k(W) \triangleq \inf_{U \in \mathcal{U}_k} \sup_{x \in W} dist(x; U) = \inf_{U \in \mathcal{U}_k} \sup_{x \in W} \inf_{y \in U} \|x - y\|,$$

where $dist(x;U)$ is the distance of x to subset U and the first infimum is taken over all subsets $U \in \mathcal{U}_k$.

Here, we are interested in approximating W with the "best" subset in the set $\mathcal{U}_k$: the subset that would minimize the worst case projection error of $x \in W$ among all subsets in $\mathcal{U}_k$. This minimal error is given by the generalized k-width of W. Figure 2.1 illustrates an example in which W is an ellipsoid in $\mathbb{R}^3$ and we are interested in approximating it with a

2-dimensional plane ($k = 2$). In this example, $\mathcal{U}_2$ is the set of all planes and plane U offers the smallest worst-case projection error $d_2(W)$ for approximating W.

Definition 2.1 generalizes the original Kolmogorov k-width definition in two ways. First, in our definition W is allowed to be a subset of matrices whereas in the original Kolmgorov width, W is a subset of vectors. This generalization is necessary as we wish to approximate the coefficient set of the Kalman predictive model whose elements $\mathcal{O}_t$ and $\mathcal{C}_t$ are matrices. Second, we allow the coefficients a_i to be matrices, generalizing over the scalar coefficients used in the original definition of Kolmogorov width. When constructing a reparameterization, a linear predictive model yields a convex objective regardless of whether the coefficients are matrices or scalars. Allowing coefficients to be matrices as opposed to restricting them to be scalars gives flexibility to find a reparameterization with small approximation error, as demonstrated in Theorem 2.2.

2.4.2 From a small width to an efficient convex relaxation

Before stating our approximation technique, we briefly describe how a small generalized k-width allows for an efficient convex relaxation. The ideas presented in this section will be made more concrete in subsequent sections.

To understand the main idea, consider system (2.4) with no inputs whose predictive model can be written as $m_{t+1} = \mathcal{O}_t y_{1:t}$. Matrix $\mathcal{O}_t$ belongs to a subset in $\mathbb{R}^{m \times mt}$ restricted by the constraints on system parameters. A naive approach for a convex relaxation is learning $\mathcal{O}_t$ in the linear predictive model $\mathcal{O}_t y_{1:t}$ directly. However in this approach, the total number of parameters is $m^2 t$, which hinders achieving sub-linear regret.

Now suppose that there exists $k \ll t$ for which the generalized k-width is small, i.e. there exist fixed known matrices $u_1, \ldots, u_k \in \mathbb{R}^{m \times mt}$ that approximate any $\mathcal{O}_t$ with a small error $\mathcal{O}_t \approx \sum_{i=1}^{k} a_i u_i$, where $a_1, \ldots, a_k \in \mathbb{R}^{m \times m}$ are coefficient matrices. The predictive model can be approximated by

$$m_{t+1} \approx \sum_{i=1}^{k} a_i u_i y_{1:t},$$

provided that norm of $y_{1:t}$ (compared to the approximation error of $\mathcal{O}_t$) is controlled with high probability. Since u_i and $y_{1:t}$ are known, we only need to learn coefficients $a_1, \ldots, a_k$ resulting in a total of $m^2 k$ parameters which is much smaller than the naive approach with $m^2 t$ parameters.

2.4.3 Filter approximation

Consider the matrix

$$\mu(G) \triangleq [I, G, G^2, \ldots, G^{T-1}],$$

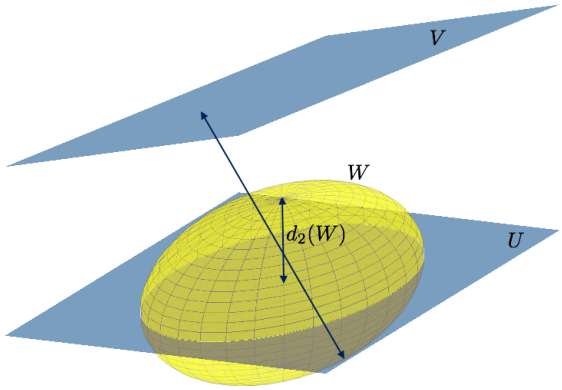

Figure 2.1: Approximating W, a 3D ellipsoid, by a 2D plane $U(u_1, u_2)$ among $\mathcal{U}_2$, the set of all planes. In this example, U has the smallest worst-case projection error that is equal to the 2-width of W denoted by $d_2(W)$.

where $G \in \mathbb{R}^{d\times d}$ is a real square matrix with spectral radius $\rho(G) \leq 1$. We seek to approximate $\mu(G) \approx \widetilde{\mu}(G) = \sum_{i=1}^{k} a_i u_i$ by a linear combination of k matrices $u_1, \ldots, u_k \in \mathbb{R}^{d\times Td}$ and coefficient matrices $\{a_1, \ldots, a_k\} \in \mathbb{R}^{d\times d}$. We evaluate the quality of approximation in operator 2-norm $\|\mu(G) - \widetilde{\mu}(G)\|_2$ by studying the generalized k-width of $\mu(G)$.

We demonstrate a sharp phase transition. Precisely, we show that when G is diagonalizable with real eigenvalues, the width $d_k(W)$ decays exponentially fast with k, but for a general G with $d \geq 2$ it decays only polynomially fast. In other words, when $d \geq 2$ the inherent structure of the set W is not easily exploited by linear subspaces.

Theorem 2.2. (Kalman filter **k**-width) *Let*

$$W \triangleq \left\{ \mu(G) = [I, G, G^2, \ldots, G^{T-1}] \;\middle|\; G \in \mathbb{R}^{d\times d}, \rho(G) \leq 1 \right\}$$

and endow the space of W with the 2-norm. The following bounds hold on the generalized k-width of the set W.

1. *If $d \geq 2$, then for $1 \leq k \leq T$,*

$$d_k(W) \geq \sqrt{T-k}.$$

2. *Restrict G to be diagonalizable with real eigenvalues. If $T \geq 10$, then for any $d \geq 1$*

$$d_k(W) \leq C_0 d\sqrt{T}(\log T)^{1/4} c^{-k/\log T},$$

where $c = \exp(\pi^2/16)$ and $C_0 = \sqrt{43}$. Moreover, there exists an efficient spectral method to compute a k-dimensional subspace that satisfies this upper bound.

Proof. Here, we only provide a proof sketch; see Section 2.8.4 for a complete proof.

Let $\lambda_1, \dots \lambda_d \in [-1, 1]$ be the eigenvalues of G. Let v_i be the right eigenvectors of G and $w_i^\top$ be the left eigenvectors of G and write

$$\mu(G) = \sum_{i=1}^{d} v_i w_i^\top ([1, \lambda_i, \dots, \lambda_i^{T-1}] \otimes I_d) = \sum_{i=1}^{d} v_i w_i^\top (\mu(\lambda_i) \otimes I_d).$$

We approximate the row vector $\mu(\lambda)$ for any $\lambda \in [-1, 1]$ using principal component analysis (PCA). The covariance matrix of $\mu(\lambda)$ with respect to a uniform measure is given by

$$H = \int_{\lambda=-1}^{1} \frac{1}{2} \mu(\lambda)^\top \mu(\lambda) d\lambda \quad \Rightarrow \quad H_{ij} = \int_{-1}^{1} \frac{1}{2} \lambda^{i-1} \lambda^{j-1} d\lambda = \frac{(-1)^{i+j} + 1}{2(i+j-1)}.$$

Let $\{\phi_j\}_{j=1}^k$ be the top k eigenvectors of H. We approximate $\mu(\lambda)$ by $\widetilde{\mu}(\lambda) = \sum_{j=1}^k \langle \mu^\top(\lambda), \phi_j \rangle \phi_j^\top$ and thus obtain

$$\mu(G) \approx \widetilde{\mu}(G) = \sum_{j=1}^{k} \Big[\sum_{i=1}^{d} \langle \mu^\top(\lambda_i), \phi_j \rangle v_i w_i^\top \Big] (\phi_j^\top \otimes I_d) = \sum_{j=1}^{k} a_j u_j.$$

We show a uniform bound on $\|\mu(G) - \widetilde{\mu}(G)\|$ by first analyzing the PCA approximation error which depends on the spectrum of matrix H. Matrix H is a positive semi-definite Hankel matrix, a square matrix whose ij-th entry only depends on the sum $i + j$. We leverage a recent result by Beckermann and Townsend (2017) who proved that the spectrum of positive semi-definite Hankel matrices decays exponentially fast.

This result, however, only guarantees a small *average* error but we need to prove that the *maximum* error is small to ensure a uniform bound on regret. Observe that the PCA error $r(\lambda) = \mu(\lambda) - \widetilde{\mu}(\lambda)$ is defined over a finite interval $[-1, 1]$ with a small average. Thus, by computing the Lipschitz constant of $r(\lambda)$, we show that the maximum approximation error is small, resulting in an upper bound on $d_k(W)$.

For the first claim, we lower bound the generalized k-width of W by relaxing the sup-norm by a weighted average, resulting in a *weighted* version of generalized k-width. We observe that the weighted k-width can be computed using PCA. We compute the approximation error of PCA showing that this error is large. □

The approximation technique used in the above theorem can readily be applied to approximate the coefficients of the Kalman predictive model by

$$\widetilde{\mathcal{O}}_t = \sum_{j=1}^{k} \Big[\sum_{i=1}^{d} \langle \mu(\lambda_i)^\top, \phi_j \rangle C v_i w_i^\top K \Big] (\phi_j^\top (t:1) \otimes I_m),$$

$$\widetilde{\mathcal{C}}_t = \sum_{j=1}^{k} \Big[\sum_{i=1}^{d} \langle \mu(\lambda_i)^\top, \phi_j \rangle C v_i w_i^\top (B - KD) \Big] (\phi_j^\top (t:1) \otimes I_n),$$

where we used the fact that $[\lambda_i^{t-1}, \ldots, \lambda_i, 1]$ can be approximated by truncated eigenvectors $\{\phi_j(t:1)\}_{j=1}^k$. The relaxed model $\widetilde{m}_t \triangleq \widetilde{\mathcal{O}}_t y_{1:t-1} + \widetilde{\mathcal{C}}_t x_{1:t-1} + Dx_t$ can be written in the form $\widetilde{m}_t = \widetilde{\Theta} f_t$. The feature vector f_t is defined in (2.9) and the parameter matrix $\widetilde{\Theta}$ is obtained by concatenating the corresponding coefficient matrices as described below

$$\widetilde{\Theta} = \Bigg[\underbrace{\Big[\sum_{i=1}^{d} \langle \mu(\lambda_i)^\top, \phi_j \rangle C v_i w_i^\top K\Big]_{j=1}^{k}}_{\substack{\in \mathbb{R}^{m \times mk} \\ \text{for output features}}} \Bigg| \underbrace{\Big[\sum_{i=1}^{d} \langle \mu(\lambda_i)^\top, \phi_j \rangle C v_i w_i^\top (B - KD)\Big]_{j=1}^{k}}_{\substack{\in \mathbb{R}^{m \times nk} \\ \text{for input features}}} \Bigg| \underbrace{D}_{\substack{\in \mathbb{R}^{m \times n} \\ \text{for } x_t}} \Bigg]_{m \times l} \tag{2.12}$$

A complete derivation of convex relaxation along with an approximation error analysis is provided in Section 2.8.6.

2.5 Regret analysis sketch

In this section we present a proof sketch for Theorem 2.1; the complete proof is deferred to Section 2.8.7 and Section 2.8.8. Let $e_t = y_t - m_t$ denote the innovation process and $b_t = \widetilde{m}_t - m_t$ denote the bias due to convex relaxation. Define

$$\mathcal{L}(T) \triangleq \sum_{t=1}^{T} \|\hat{m}_t - m_t\|_2^2. \tag{2.13}$$

$\mathcal{L}(T)$ measures the difference between Algorithm 1 predictions and the Kalman predictions in hindsight. Regret defined in (2.7) can be written as

$$\text{Regret}(T) = \sum_{t=1}^{T} \|\hat{m}_t - m_t\|_2^2 - \sum_{t=1}^{T} 2e_t^\top (\hat{m}_t - m_t) = \mathcal{L}(T) - \sum_{t=1}^{T} 2e_t^\top (\hat{m}_t - m_t). \tag{2.14}$$

Using an argument based on self-normalizing martingales, the second term is shown to be of order $\sqrt{\mathcal{L}(T)}$ and thus, it suffices to establish a bound on $\mathcal{L}(T)$. Define

$$Z_t \triangleq \alpha I + \sum_{i=1}^{t} f_i f_i^\top, \qquad E_t \triangleq \sum_{i=1}^{t} e_i f_i^\top, \qquad B_t \triangleq \sum_{i=1}^{t} b_i f_i^\top. \tag{2.15}$$

A straighforward decomposition of loss gives

$$\mathcal{L}(T) \leq \underbrace{3 \sum_{i=1}^{T} \|E_{t-1} Z_{t-1}^{-1} f_t\|_2^2}_{\text{least squares error}} + \underbrace{3 \sum_{i=1}^{T} \|B_{t-1} Z_{t-1}^{-1} f_t + b_t\|_2^2}_{\text{improper learning bias}} + \underbrace{3 \sum_{i=1}^{T} \|\alpha \tilde{\Theta} Z_{t-1}^{-1} f_t\|_2^2}_{\text{regularization error}}. \tag{2.16}$$

2.5.1 Least squares error

Among all, it is most difficult to establish a bound on the least squares error. Consider the following upper bound

$$\sum_{t=1}^{T}\|E_{t-1}Z_{t-1}^{-1}f_t\|_2 \leq \max_{1\leq t\leq T}\|E_{t-1}Z_{t-1}^{-1/2}\|_2 \sum_{t=1}^{T}\|Z_{t-1}^{-1/2}f_t\|_2.$$

We show the first term is bounded by polylog(T) for any $\delta \geq 0$. In particular,

$$\max_{1\leq t\leq T}\|E_{t-1}Z_{t-1}^{-1/2}\|_2 \lesssim_{R_\Theta,m,\gamma,\beta,\delta} \max_{1\leq t\leq T}\log\Big(\frac{\det(Z_t)\det(\alpha I)^{-1}}{\delta}\Big) \lesssim_{R_\Theta,m,\gamma,\beta,\delta} k\log(T).$$

Our argument is based on vector self-normalizing martingales, a similar technique used by Abbasi-Yadkori et al. (2011); Sarkar and Rakhlin (2018); Tsiamis and Pappas (2020). $\det(Z_t)$ is bounded by poly(T) for two reasons. First, the feature dimension, which is linear in the number of filters k, is polylog(T) on account of Theorem 2.2. Second, the marginal stability assumption ($\rho(A) \leq 1$) ensures that features and thus Z_t grow at most polynomially in t.

It remains to prove that the summation $\sum_{t=1}^{T}\|Z_{t-1}^{-1/2}f_t\|_2^2$ is bounded by polylog(T) with high probability. We use an argument inspired by Lemma 2 of Lai et al. (1982) and Schur complement lemma (Zhang, 2006) to conclude that

$$\sum_{t=1}^{T}\|Z_{t-1}^{-1/2}f_t\|_2^2 \asymp_M \text{polylog}(T) \quad \Leftrightarrow \quad Z_{t-1} - \frac{1}{c_T}f_tf_t^\top \succeq 0 \quad \text{for} \quad c_T \asymp_M \text{polylog}(T).$$

Therefore, it suffices to prove the right-hand side. We show a high probability Löwner upper bound on $f_tf_t^\top$ based on the feature covariance $\text{cov}(f_t)$ using sub-Gaussian quadratic tail bounds (Vershynin, 2018). To capture the excitation behavior of features, we establish a Löwner lower bound on Z_t by proving that the process $\{f_t\}_{t\geq 1}$ satisfies a *martingale small-ball condition* (Mendelson, 2014; Simchowitz et al., 2018). We leverage the small-ball condition lower tail bounds and prove the following lemma.

Lemma 2.1. (Martingale small-ball condition) *Let $\phi_1,\dots,\phi_k \in \mathbb{R}^T$ be orthonormal and fix $\delta > 0$. Given system (2.4), let $\mathcal{F}_t = \sigma\{\eta_0,\dots,\eta_{t-1},\zeta_1,\dots,\zeta_t\}$ be a filteration and for all $t \geq 1$ define*

$$f_t = \psi_1 \otimes y_{t-1} + \cdots + \psi_{t-1}\otimes y_1, \quad \textit{where} \quad \psi_i = [\phi_1(i),\dots,\phi_k(i)]^\top.$$

Let $\Gamma_i = cov(f_{t+i}|\mathcal{F}_t)$.

1. *For any $1 \leq s \leq T$, the process $\{f_t\}_{t\geq 1}$ satisfies a $(s,\Gamma_{s/2},p=3/20)$-block martingale small-ball (BMSB) condition, i.e. for any $t \geq 0$ and any fixed ω in unit sphere $\mathcal{S}^{l-1}$*

$$\frac{1}{s}\sum_{i=1}^{s}\mathbb{P}\Big(|\omega^\top f_{t+i}| \geq \sqrt{\omega^\top\Gamma_{s/2}\omega} \mid \mathcal{F}_t\Big) \geq p.$$

2. Under the assumptions of Theorem 2.1, the following holds with probability at least $1-\delta$

$$\sum_{t=1}^{T}\|Z_{t-1}^{-1/2}f_t\|_2^2 \leq \kappa k^2 \log(T)\,\mathrm{poly}(R_\Theta, \beta, m, \log(\gamma), \log\left(\frac{1}{\delta}\right)).$$

Provided that the number of filters is polylog(T), the above lemma ensures that $\sum_{t=1}^{T}\|Z_{t-1}^{-1/2}f_t\|_2^2$ is also polylog(T), which is the desired result.

2.5.2 Improper learning bias

We characterize the improper learning bias term in (2.16) by first showing a uniform high probability bound on the convex relaxation error stated in the theorem below. The proof can be found in Section 2.8.6.

Theorem 2.3. (Convex relaxation error bound, informal) *Consider system (2.4) with bounded inputs* $\|x_t\|_2 \leq R_x$ *and assume conditions (i)-(ii) of Theorem 2.1 holds. Then for any* $\epsilon, \gamma \geq 0$*, if the number of filters* k *satisfies* $k \gtrsim_M \log(T)\log(T/\epsilon)$*, then the following holds for* $\widetilde{\Theta}$ *as defined in (2.12)*

$$\mathbb{P}\left[\|\widetilde{\Theta}f_t - m_t\|_2^2 \geq \epsilon\right] \leq \delta.$$

In Section 2.8.8, the result of the above theorem is followed by an application of a vector self-normalizing martingale theorem to prove a polylog(T) bound on the improper learning bias.

Remark 2.2. While the algorithm derivation, convex relaxation approximation error, and most of the regret analysis consider a system with control inputs, the excitation result of Lemma 2.1 is given without inputs. We believe that extending our analysis for LDS with inputs is possible by characterizing input features and in light of the experiments. However, such an extension requires some care. For instance, one needs to characterize the covariance between features constructed from observations and features constructed from inputs to demonstrate a small-ball condition.

2.5.3 Regularization error

Lastly, we demonstrate an upper bound on the regularization error in (2.16). We write the following bound

$$\sum_{t=1}^{T}\|\alpha\tilde{\Theta}Z_{t-1}^{-1}f_t\|_2^2 \leq \alpha^2\frac{1}{\alpha}\|\tilde{\Theta}\|_2^2\sum_{t=1}^{T}\|Z_{t-1}^{-1/2}f_t\|_2^2 \leq \sum_{t=1}^{T}\|Z_{t-1}^{-1/2}f_t\|_2^2.$$

The first inequality is based on $Z_t \succeq \alpha I$ and the submultiplicative property of norm. The second inequality uses the fact that $\|\widetilde{\Theta}\|_2^2 \leq 1/\alpha$ for $\alpha \asymp_M (R_\Theta k T^\beta)^{-1}$ as shown in Section 2.8.8. The last term is bounded as result of Lemma 2.1.

2.6 Experiments

We carry out experiments to evaluate the empirical performance of our provable method in three dynamical systems with long-term memory. We compare our results against those yielded by the wave filtering algorithm (Hazan et al., 2017) implemented with follow the regularized leader and the truncated filtering algorithm (Tsiamis and Pappas, 2020). We consider $\|\hat{m}_t - m_t\|^2$, the squared error between algorithms predictions and predictions by a Kalman filtering algorithm that knows system parameters, as a performance measure. For all algorithms, we use $k = 20$ filters and run each experiment independently 100 times and present the average error with 99% confidence intervals.

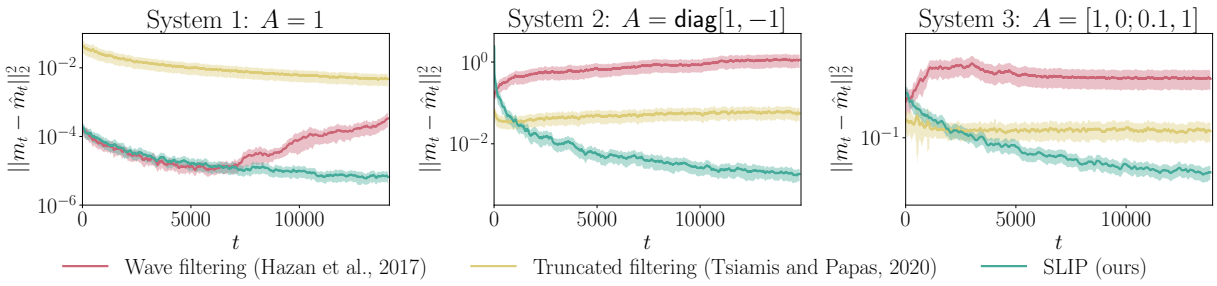

Figure 2.2: Performance of our algorithm compared with wave filtering and truncated filtering. System 1 is an scalar LDS with $A = B = D = 1$, $C = Q = R = 0.001$, and $x_t \sim \mathcal{N}(0, 2)$. System 2 is a multi-dimensional LDS with no inputs and $A = \mathsf{diag}[-1, 1]$, $C = [0.1, 0.5]$, $R = 0.5$, and $Q = [4, 6; 6, 10] \times 10^{-3}$. System 3 is another multi-dimensional LDS with non-symmetric $A = [1, 0; 0.1, 1]$, $x_i \sim \mathcal{U}(-0.01, 0.01)$, $Q = 10^{-3}I$, $R = I$, $C = [0, 0.1; 0.1, 1]$, and B, D are matrices of all ones.

In the first example (Figure 2.2, left), we consider a scalar marginally stable system with $A = 1$ and Gaussian inputs. This system exhibits long forecast memory with $G \approx 0.999$. Observe that the truncated filter suffers from a large error which is due to ignoring long-term dependencies. The wave filter predictions also deviates from optimal predictions as it only considers $y_{t-1}, x_{1:t}$ for predicting y_t. The middle plot in Figure 2.2 presents the results for a multi-dimensional system with $A = \mathsf{diag}[-1, 1]$ and no inputs. This system also has a long forecast memory (G has eigenvalues $\approx \{0.991, -0.932\}$), resulting in poor performance of the truncated filter. The wave filter also performs poorly in this system as it is only driven by stochastic noise. For the last example, we consider another multi-dimensional system where A is a lower triangular matrix (Figure 2.2, right). This is a difficult example where $\rho(A) = 1$ but $\|A\|_2 > 1$, resulting in a polynomial growth of the observations over time. The results show that our algorithm outperforms both the wave filter, which requires a symmetric A, and the truncated filter in the case of fast-growing observations.

Comparison with the EM algorithm. We conduct an experiment in a scalar LDS to compare the performance of our algorithm with the EM algorithm that estimates system parameters (Figure 2.3, left). The parameters estimated by the EM algorithm are later used by the Kalman filter for predictions. In this experiment, we set the horizon $T = 200$ due to the large computation time required by the EM algorithm. The number of filters k is set to 5 for all other three algorithms. The experiment was simulated 100 independent times and the average error together with the 99% confidence intervals are presented.

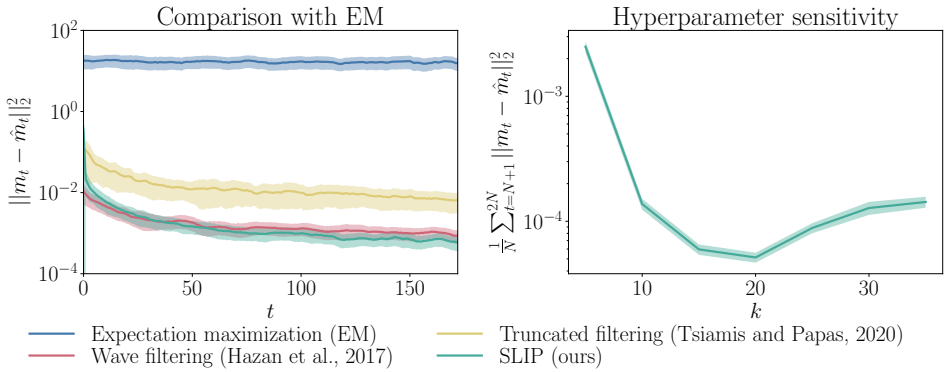

Figure 2.3: Left: Performance of our algorithm compared with wave filtering, truncated filtering, and expectation maximization in a scalar system with parameters $A = B = C = D = 1$, noise covariance matrices $Q = R = 0.001$, inputs $x_t \sim \mathcal{N}(0, 2)$, and horizon $T = 200$. Right: Hyperparameter sensitivity of our algorithm in the same systems with inputs $x_t \sim \mathcal{N}(0, 0.5)$ and horizon $T = 10000$.

For the system considered in this experiment, EM performs poorly. System-identification-based methods such as EM, besides being significantly slower, do not have regret guarantees and they can fail in some examples; a similar observation was made by Hazan et al. (2017).

Hyperparameter sensitivity. The SLIP algorithm has two hyperparameters: the number of filters k and the regularization parameter α. In the experiments, we set $\alpha > 0$ only when the empirical feature covariance matrix is singular, which we observe only happens in the first two time steps. For the number of filters k, Theorem 2.1 provides a guideline of choosing k of order $\log^2(T)$. The right plot in Figure 2.3 demonstrates the sensitivity of the SLIP algorithm with respect to the number of filters k. The system considered for this experiment is scalar with Gaussian inputs and the horizon is set to 10000. As before, the experiment was simulated 100 independent times. We vary k from 5 to 35 and measure the average prediction error from 5000 to 10000 ($N = 5000$ in the plot). We observe that the SLIP algorithm is robust with respect to parameter k.

2.7 Discussion

We presented the SLIP algorithm, an efficient algorithm for learning a predictive model of an unknown LDS. Our algorithm provably and empirically converges to the optimal predictions of the Kalman filter given the true system parameters, even in the presence of long forecast memory. We analyzed the generalized k-width of the Kalman filter coefficient set with closed-loop matrix G and obtained a low-dimensional linear approximation of the Kalman filter when G is diagonalizable with real eigenvalues. We proved that without assuming real eigenvalues, the Kalman filter coefficient set is difficult to approximate by linear subspaces. Our approach of studying k-width as a measure for the possibility of an efficient convex relaxation may be of independent interest. Important future directions are designing efficient algorithms that handle arbitrary G, providing theoretically guaranteed uncertainty estimation for prediction, extending the ideas for prediction in the presence of long-term memory to non-linear systems and predictors, and investigating the efficacy of such predictors in applications such as device failure prediction for the analysis of datacenter reliability (Rashidinejad et al., 2020b).

2.8 Proofs

2.8.1 Organization

The rest of this chapter includes the omitted proofs in the prior sections. We start by giving a matrix representation of system (2.4) in Section 2.8.2 that describes aggregated observations $y_{1:t}$ in terms of past inputs and noise. We also restate our matrix representation of the Kalman predictive model. In Section 2.8.3, we provide upper bounds on the matrix coefficients used in the aggregated system representation as well as a high probability upper bound on the norm of observations $\|y_{1:t}\|_2$. We also discuss our assumption on the 2-norm of the Kalman coefficient matrices (control matrix $\mathcal{C}_t$ and observation matrix $\mathcal{O}_t$) and present two examples providing bounds on the 2-norm of these coefficients.

In Section 2.8.4, we first analyze the error of approximating $\mu(\lambda)$ by spectral methods, considering the spectrum of the Hankel covariance matrix and give a proof for Theorem 2.2. We then analyze convex relaxation approximation error in Section 2.8.6 and show that the convex relaxation bias is small with high probability, provided that the number of filters $k \gtrsim_M \log^2(T)$.

We begin our analysis of regret in Section 2.8.7, in which we write a bound on regret decomposed into least squares error, improper learning bias, regularization error, and innovation error. We further extract the term $\|Z_{t-1}^{-1/2} f_t\|_2^2$ making the bound ready for analysis in subsequent sections. We then complete our regret analysis in Section 2.8.8. We start by giving a high probability bound on $\det(Z_t)$ that appears multiple times throughout our analysis. We derive a result on self-normalizing vector martingales that assists bounding several terms. and provide a bound on $\|Z_{t-1}^{-1/2} f_t\|_2^2$ using sub-Gaussian tail properties, a

block-martingale small-ball condition, and a filter quadratic function condition. The proof of Lemma 2.1, analysis of the regularization term, and innovation error are subsequently provided. The section is concluded by the proof of the main regret bound. A few technical lemmas are presented in Section 2.8.9.

.

2.8.2 Aggregated representations

We start by introducing an aggregated notation for representing linear dynamical systems and the Kalman predictive model.

Linear dynamical systems

For the linear dynamical system of (2.4), define the following matrices

$$\mathcal{T}_t = \begin{bmatrix} C & 0 & 0 & \dots & 0 \\ CA & C & 0 & \dots & 0 \\ CA^2 & CA & C & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ CA^{t-1} & CA^{t-2} & CA^{t-3} & \dots & C \end{bmatrix} \begin{bmatrix} AP^{1/2} & 0 & 0 & \dots & 0 \\ 0 & Q^{1/2} & 0 & \dots & 0 \\ 0 & 0 & Q^{1/2} & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & Q^{1/2} \end{bmatrix},$$

$$\mathcal{I}_t = \begin{bmatrix} D & 0 & 0 & \dots & 0 \\ CB & D & 0 & \dots & 0 \\ CAB & CB & D & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ CA^{t-2}B & CA^{t-3}B & CA^{t-4}B & \dots & D \end{bmatrix}, \tag{2.17}$$

$$\mathcal{R}_t = \begin{bmatrix} R^{1/2} & 0 & 0 & \dots & 0 \\ 0 & R^{1/2} & 0 & \dots & 0 \\ 0 & 0 & R^{1/2} & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & R^{1/2} \end{bmatrix}.$$

Let $\mathcal{K}_t\mathcal{K}_t^\top = \mathcal{T}_t\mathcal{T}_t^\top + \mathcal{R}_t\mathcal{R}_t^\top$, where $\mathcal{K}_t$ is the unique solution to Cholesky decomposition. The system observations $y_{1:t}$ can be written as

$$y_{1:t} = \mathcal{K}_t\xi_{1:t} + \mathcal{I}_t x_{1:t}, \tag{2.18}$$

where $\xi_i \in \mathbb{R}^m$ is a Gaussian random vector with covariance I_m.

Kalman filter

For convenience, we restate our notation of the Kalman predictive model from Section 2.2.1. Define the following matrices

$$\begin{aligned}\mathcal{O}_t &= \begin{bmatrix} CG^{t-1}K & CG^{t-3}K & \dots & CK \end{bmatrix}, \\ \mathcal{C}_t &= \begin{bmatrix} CG^{t-1}(B-KD) & CG^{t-2}(B-KD) & \dots & C(B-KD) \end{bmatrix}.\end{aligned} \tag{2.19}$$

We refer to $\mathcal{O}_t$ and $\mathcal{C}_t$ as *observation matrix* and *control matrix*, respectively. Using the above notation, the Kalman prediction m_{t+1} is given by

$$m_{t+1} = \mathcal{O}_t y_{1:t} + \mathcal{C}_t x_{1:t} + D x_{t+1}.$$

2.8.3 Norm bounds

As a preliminary step, we compute a few bounds that will be used later in the regret analysis of the SLIP algorithm. In particular, we compute upper bounds on the norms of parameter matrices defined in (2.17) and discuss upper bounds on the norms of observation and control matrix of the Kalman predictive model. Further, we derive a high probability upper bound on the observation norm.

Bounds on parameters

The following lemma provides upper bounds on the norm of matrices that describe a linear dynamic system.

Lemma 2.8.1. (LDS parameter bounds) *Consider system (4). Let* $R_P = \max\{\|B\|_2, \|C\|_2, \|D\|_2\}$ *and* $R_C = \max\{\|P\|_2, \|Q\|_2, \|R\|_2\}$. *Suppose that* $\|A^t\|_2 \leq \gamma t^{\log(\gamma)}$ *for a bounded constant* $\gamma \geq 1$. *For* $\mathcal{T}_t$, $\mathcal{I}_t$, *and* $\mathcal{K}_t$ *defined in* (2.17), *the following operator norm bounds hold:*

(i) $\|\mathcal{T}_t\|_2 \leq R_C^{1/2} R_P \gamma(1+\gamma) t^{\log(\gamma)+1}$,

(ii) $\|\mathcal{I}_t\|_2 \leq R_P[1 + t\gamma t^{\log(\gamma)}]$,

(iii) $\|\mathcal{K}_t\|_2 \leq \sqrt{R_C + R_C R_P^2 (1+\gamma)^4 t^{2\log(\gamma)+2}}$.

Proof. By Lemma 2.8.19,

$$\|\mathcal{T}_t\|_2 \leq (\|A\|_2+1) R_C^{1/2} \|C\|_2 \sum_{i=1}^{t} \|A^i\|_2 \leq R_C^{1/2} R_P \gamma(1+\gamma) t^{\log(\gamma)+1}.$$

Similarly,

$$\|\mathcal{I}_t\|_2 \leq \|D\|_2 + \|C\|_2 \|B\|_2 \sum_{i=1}^{t} \|A^i\|_2 \leq R_P + R_P^2 \gamma t^{\log(\gamma)+1}.$$

It follows by the sub-additive property of matrix operator norm that

$$\|\mathcal{K}_t\mathcal{K}_t^\top\|_2 = \|\mathcal{K}_t\|_2^2 \leq \|\mathcal{T}_t\|_2^2 + \|\mathcal{R}_t\|_2^2 \quad \Rightarrow \quad \|\mathcal{K}_t\|_2 \leq \sqrt{R_C + R_C R_P^2 (1+\gamma)^4 t^{2\log(\gamma)+2}}.$$

□

In the regret analysis, we assume that $\|\mathcal{O}_t\|_2 \leq R_{\mathcal{O}} t^\beta$ for a finite $\beta \geq 0$. We justify this assumption in the examples below. The following example shows that $\beta = 0$ when the system is single-input single-output (SISO).

Example 2.8.1. (Observation matrix norm bound in SISO systems) *For a SISO linear dynamical system, the following equation holds*

$$KC = \frac{A\Sigma^+ C^2}{\Sigma^+ C^2 + R} \Rightarrow 0 \leq KC \leq A.$$

We have $G = A - KC$. Applying the above constraint gives

$$G \leq A$$

The squared norm of vector $\mathcal{O}_t$ is given by

$$\|\mathcal{O}_t\|_2^2 = \sum_{i=0}^{t-1} (KCG^i)^2 = \sum_{i=0}^{t-1} (A-G)^2 G^{2i}.$$

Under the constraint $G \leq A \leq 1$, the maximum of $\|\mathcal{O}_t\|_2^2$ is 1 obtained when $G = 0$ and $A = 1$.

In the following example, we compute a loose upper bound on $\|\mathcal{O}_t\|_2$.

Example 2.8.2. (Observation matrix norm bound in MIMO systems with $\mathbf{d} = \mathbf{m}$) *We begin by computing an upper bound on the norm of the Kalman gain. Let $K = AK'$. By the recursive updates of a stationary Kalman gain, we write*

$$CK' = C\Sigma^+ C^\top [C\Sigma^+ C^\top + Q]^{-1} \preceq I \Rightarrow \|CK'\|_2 \leq 1.$$

Lower bounding $\|CK\|_2$ yields

$$\|K'\|_2 \sigma_{\min}(C) \leq \|CK'\|_2 \leq 1 \Rightarrow \|K'\|_2 \leq \frac{1}{\sigma_{\min}(C)}.$$

Let $\kappa_C = \sigma_{\max}(C)/\sigma_{\min}(C)$ to be the condition number of C. Assume $\|G^t\|_2 \leq \gamma_g t^{\log(\gamma_g)}$. We have

$$\|\mathcal{O}_t\|_2 \leq \sum_{i=1}^{t} \|C\|_2 \|G^i\|_2 \|K'\|_2 \leq \kappa_C \gamma_g t^{\log(\gamma_g)+1}.$$

Bound on observation norm

One of the quantities that appear in the regret analysis of our algorithm is the squared norm of $y_{1:t}$. The following lemma provides a high probability upper bound for $\|y_{1:t}\|_2^2$.

Lemma 2.8.2. (Observation norm bound) *Consider system (4). Let* $R_P = \max\{\|B\|_2, \|C\|_2, \|D\|_2\}$, $R_C = \max\{\|P\|_2, \|Q\|_2, \|R\|_2\}$, *and* $\|x_t\|_2 \leq R_x$. *Suppose that* $\|A^t\|_2 \leq \gamma t^{\log(\gamma)}$ *for a bounded constant* $\gamma \geq 1$. *For any* $\delta > 0$ *and all* $t \geq 0$,

$$\mathbb{P}\left[\|y_{1:t}\|_2^2 \geq 6(R_P^2+1)(R_x^2+R_C)(1+\gamma)^4(mt+\delta)t^{2+2\log(\gamma)}\right] \leq e^{-\delta}.$$

Proof. From (2.18), we see that

$$\|y_{1:t}\|_2^2 \leq 2\|\mathcal{I}_t\|_2^2\|x_{1:t}\|_2^2 + 2\|\mathcal{K}_t\|_2^2\|\xi_{1:t}\|_2^2$$

Using Gaussian upper tail bounds (Hsu et al., 2012), we have

$$\mathbb{P}\left[\|\xi_{1:t}\|_2^2 > 2mt+3\delta\right] \leq \mathbb{P}\left[\|\xi_{1:t}\|_2^2 > mt+2\sqrt{mt\delta}+2\delta\right] \leq e^{-\delta}.$$

Using the bounds computed in Lemma 2.8.1, the following holds with probability at least $1-e^{-\delta}$

$$\begin{aligned}\|y_{1:t}\|_2^2 &\leq 2\|\mathcal{I}_t\|_2^2\|x_{1:t}\|_2^2 + 2\|\mathcal{K}_t\|_2^2\|\xi_{1:t}\|_2^2 \\ &\leq 6(R_P^2+1)(R_x^2+R_C)(1+\gamma)^4(mt+\delta)t^{2+2\log(\gamma)}.\end{aligned} \tag{2.20}$$

□

2.8.4 Filter approximation and width analysis

In this section we first provide a series of lemmas characterizing the reconstruction error of applying PCA to approximate the vector function $\mu(\lambda) = [1, \lambda, \ldots, \lambda^{T-1}]$. These lemmas are later used to prove Theorem 2.2.

Bounds on PCA approximation error

The goal of this section is to establish a uniform bound on the norm of the reconstruction error of approximating $\mu(\lambda)$ with $\widetilde{\mu}(\lambda)$. The following lemma states a standard result on the average PCA reconstruction error, presented here for completeness.

Lemma 2.8.3. (Average reconstruction error bound) *Let* $\mu(\lambda) \in \mathbb{R}^T$ *be a vector function parameterized by* $\lambda \in \mathcal{A}$. *Define the following matrix with respect to probability measure* p

$$Z = \int_{\mathcal{A}} \mu(\lambda)\mu^\top(\lambda)p(d\lambda).$$

Let $\{(\sigma_j, \phi_j)\}_{j=1}^T$ be the eigenpairs of Z. Let $\widetilde{\mu}(\lambda)$ be the projection of $\mu(\lambda)$ to the linear subspace spanned by $\{\phi_1, \ldots, \phi_k\}$. Then,

$$\int_{\mathcal{A}} \|\mu(\lambda) - \widetilde{\mu}(\lambda)\|_2^2 p(d\lambda) = \sum_{j=k+1}^{T} \sigma_j.$$

Proof. Define U_k to be a $T \times k$ matrix with columns $\phi_1, \ldots \phi_k$, the eigenvectors of matrix Z. The reconstruction error can be written as

$$r(\lambda) = \mu(\lambda) - U_k U_k^\top \mu(\lambda) = (I - U_k U_k^\top)\mu(\lambda) = \Pi_k \mu(\lambda).$$

The average squared norm of reconstruction error is given by

$$\begin{aligned}\int_{\mathcal{A}} \|r(\lambda)\|_2^2 p(d\lambda) &= \int_{\mathcal{A}} \mathrm{tr}[r(\lambda) r(\lambda)^\top] p(d\lambda) = \int_{\mathcal{A}} \mathrm{tr}[\Pi_k \mu(\lambda)\mu(\lambda)^\top \Pi_k^\top] p(d\lambda) \\ &= \mathrm{tr}[\Pi_k \int_{\mathcal{A}} \mu(\lambda)\mu(\lambda)^\top p(d\lambda) \Pi_k^\top] = \mathrm{tr}[\Pi_k Z \Pi_k^\top] = \sum_{j=k+1}^{T} \sigma_j.\end{aligned}$$

□

We then use Lipschitz continuity of $\mu(\lambda)$ over the interval $[-1, 1]$ to establish a uniform bound on the reconstruction error.

Lemma 2.8.4. *Let $\mu(\lambda) = [1, \lambda, \lambda^2, \ldots, \lambda^{T-1}]^\top$ for $\lambda \in [-1, 1]$ and define*

$$H = \int_{-1}^{1} \frac{1}{2} \mu(\lambda)\mu(\lambda)^\top d\lambda.$$

Let $\{(\sigma_j, \phi^j)\}_{j=1}^T$ be the eigenpairs of H, where σ_j are in decreasing order. Let $\widetilde{\mu}(\lambda)$ be the projection of $\mu(\lambda)$ to the linear subspace spanned by $\{\phi_1, \ldots, \phi_k\}$. Then, for any $\lambda \in [-1, 1]$ and $T \geq 1$,

$$\|\mu(\lambda) - \widetilde{\mu}(\lambda)\|_2^2 \leq T \sqrt{2 \sum_{j=k+1}^{T} \sigma_j}.$$

Proof. Let us first compute an upper bound on the Lipschitz constant of $\mu(\lambda)$ over $\lambda \in [-1, 1]$. The Lipschitz constant of $\mu(\lambda)$ is bounded by the norm of Jacobian $J(\mu(\lambda)) = [0, 1, 2\lambda, \ldots, (T-1)\lambda^{T-2}]$. Thus,

$$\frac{\|\mu(\lambda_2) - \mu(\lambda_1)\|_2}{|\lambda_2 - \lambda_1|} \leq \|J(\mu(\lambda))\|_2 \leq \sqrt{\sum_{t=1}^{T-1} t^2} \leq \sqrt{T^3/3}.$$

Define U_k to be a matrix with columns $\phi_1, \dots \phi_k$. The reconstruction error can be written as $r(\lambda) = (I - U_k U_k^\top)\mu(\lambda) = \Pi_k \mu(\lambda)$. A Lipschitz constant for reconstruction error norm is given by

$$\begin{aligned}
\|r(\lambda_2)\|_2 - \|r(\lambda_1)\|_2 &\leq \|r(\lambda_2) - r(\lambda_1)\|_2 && \text{(inverse triangle inequality)} \\
&= \|\Pi_k(\mu(\lambda_2) - \mu(\lambda_1))\|_2 \\
&\leq \|\Pi_k\|_2 \|(\mu(\lambda_2) - \mu(\lambda_1))\|_2 && \text{(multiplicative property of norm)} \\
&\leq \|(\mu(\lambda_2) - \mu(\lambda_1))\|_2 && (\Pi_k \text{ is contractive}) \\
&\leq \sqrt{T^3/3}|\lambda_2 - \lambda_1| && \text{(Lipschitz continuity of } \mu(\lambda))
\end{aligned}$$

Thus, an upper bound on the Lipschitz constant of $\|r(\lambda)\|_2^2$ can be computed

$$\begin{aligned}
\|r(\lambda_2)\|_2^2 - \|r(\lambda_1)\|_2^2 &= (\|r(\lambda_2)\|_2 - \|r(\lambda_1)\|_2)(\|r(\lambda_2)\|_2 + \|r(\lambda_1)\|_2) \\
&\leq \left(\sqrt{T^3/3}|\lambda_2 - \lambda_1|\right)\left(2 \max_\lambda \|r(\lambda)\|_2\right) \\
&\leq 2\sqrt{T^3/3}\|\Pi_k\|_2 \max_\lambda \|\mu(\lambda)\|_2 |\lambda_2 - \lambda_1| \\
&\leq 2T^2|\lambda_2 - \lambda_1|.
\end{aligned}$$

Let $R_r = \max_\lambda \|r(\lambda)\|_2^2$. On the account of Lemma 2.8.3, $\|r(\lambda)\|_2^2$ has a bounded average over the interval $[-1, 1]$. A bounded and $(2T^2)$-Lipschitz function that achieves the maximum R_r has a triangular shape. It follows that

$$\frac{R_r^2}{2T^2} \geq \sum_{j=k+1}^{T} \sigma_j \quad \Rightarrow \quad \|r(\lambda)\|_2^2 \leq R_r \leq T\sqrt{2\sum_{j=k+1}^{T} \sigma_j}.$$

□

In the following lemma, we prove that the PCA reconstruction error is small due to the exponential decay of the spectrum of the Hankel covariance matrix H.

Lemma 2.8.5. (Uniform bound on reconstruction error) *Under the assumptions of Lemma 2.8.4 and for any $T \geq 10$*

$$\|\mu(\lambda) - \widetilde{\mu}(\lambda)\|_2^2 \leq C_0 T\sqrt{\log T} c^{-k/\log T},$$

where $c = \exp(\pi^2/8)$ and $C_0 = 43$.

Proof. We appeal to the following, which appears as Corollary 5.4 in Beckermann and Townsend (2017).

Lemma 2.8.6. *Let $H_n \in \mathbb{R}^{n \times n}$ be a positive semi-definite Hankel matrix. Then,*

$$\sigma_{j+2k} \leq 16\left[\exp\left(\frac{\pi^2}{4\log(8\lfloor n/2 \rfloor/\pi)}\right)\right]^{-2k+2} \sigma_j(H_n), \quad \text{for} \quad 1 \leq j + 2k \leq n. \tag{2.21}$$

Setting $j = 1$ in (2.21) with the assumption $T \geq 10$ yields

$$\sigma_{2+2k} \leq \sigma_{1+2k} \leq 16\sigma_1 \exp\left(\frac{\pi^2}{4\log T}\right)^{-2k+2} \leq 1168\sigma_1 \exp\left(\frac{\pi^2}{4\log T}\right)^{-2k}.$$

Let $c = \exp(\pi^2/8)$. It follows that

$$\sigma_j \leq 1168\sigma_1 c^{\frac{-2(j-2)}{\log T}} \leq 10512\sigma_1 c^{\frac{-2j}{\log T}}.$$

The largest singular value of Hankel matrix H is bounded by

$$\sigma_1 \leq \mathrm{tr}(H) \leq \sum_{k=1}^{T} \frac{1}{2k+1} \leq \sum_{k=1}^{T} \frac{1}{k} - 1 \leq \log T,$$

where the last inequality is due to a classic bound on the T-th harmonic number. We conclude from Lemma 2.8.4 that

$$\begin{aligned}\|\mu(\lambda) - \widetilde{\mu}(\lambda)\|_2^2 &\leq T\sqrt{21024\sigma_1 \sum_{j=k+1}^{T} c^{-2j/\log T}} \\ &\leq T\sqrt{21024\log T \frac{c^{-2k/\log T}}{c^2-1}} \leq 43T\sqrt{\log T}c^{-k/\log T}.\end{aligned}$$

□

2.8.5 Generalized Kolmogorov width analysis: Proof of Theorem 2.2

Proof of Theorem 2.2. We first prove the second claim. Let $\lambda_1, \ldots \lambda_d \in [-1, 1]$ denote the eigenvalues of G. Let v_i be the right eigenvectors of G and $w_i^\top$ be the left eigenvectors of G. Eigendecomposition of G^t implies $G^t = \sum_{i=1}^{d} v_i w_i^\top \lambda_i$. Therefore, matrix $\mu(G) = [I, G, \ldots, G^{T-1}]$ can be written as

$$\mu(G) = \sum_{i=1}^{d} v_i w_i^\top ([1, \lambda_i, \ldots, \lambda_i^{T-1}] \otimes I_d) = \sum_{i=1}^{d} v_i w_i^\top (\mu(\lambda_i) \otimes I_d),$$

where $\mu(\lambda_i) = [1, \lambda_i, \ldots, \lambda_i^{T-1}]$ is a row vector. We approximate $\mu(\lambda)$ for any $\lambda \in [-1, 1]$ using principal component analysis (PCA). The covariance matrix of $\mu(\lambda)$ with respect to a uniform measure is given by

$$H = \int_{\lambda=-1}^{1} \frac{1}{2}\mu(\lambda)^\top \mu(\lambda) d\lambda \quad \Rightarrow \quad H_{ij} = \int_{-1}^{1} \frac{1}{2}\lambda^{i-1}\lambda^{j-1} d\lambda = \frac{(-1)^{i+j}+1}{2(i+j-1)}.$$

Let $\{\phi_j\}_{j=1}^k$ be the top k eigenvectors of H. We approximate $\mu(\lambda)$ by $\widetilde{\mu}(\lambda) = \sum_{j=1}^k \langle \mu^\top(\lambda), \phi_j \rangle \phi_j^\top$:

$$\begin{aligned}\mu(G) \approx \widetilde{\mu}(G) &= \sum_{i=1}^d v_i w_i^\top \Big(\sum_{j=1}^k \langle \mu^\top(\lambda), \phi_j \rangle \phi_j^\top \otimes I_d\Big) \\ &= \sum_{j=1}^k \Big[\sum_{i=1}^d \langle \mu^\top(\lambda_i), \phi_j \rangle v_i w_i^\top\Big](\phi_j^\top \otimes I_d) = \sum_{j=1}^k a_j u_j.\end{aligned}$$

Check that $a_1, \ldots, a_k \in \mathbb{R}^{d \times d}$ and $u_1, \ldots, u_k \in \mathbb{R}^{d \times dT}$. We have

$$\begin{aligned}d_k(W) = \|\mu(G) - \widetilde{\mu}(G)\|_2 &= \|\sum_{i=1}^d v_i w_i^\top (\mu(\lambda_i) - \widetilde{\mu}(\lambda_i)) \otimes I_d\|_2 \\ &\le \sum_{i=1}^d \|\mu(\lambda_i) - \widetilde{\mu}(\lambda_i)\|_2 \\ &\le d \sup_\lambda \|\mu(\lambda) - \widetilde{\mu}(\lambda)\|_2.\end{aligned}$$

The first inequality uses subadditive and submultiplicative properties of norm and that $\|v_i w_i^\top\|_2 \le 1, \|I_d\|_2 = 1$. By Lemma 2.8.5,

$$d_k(W) \le d\sqrt{43T}(\log T)^{1/4}\Big(\exp(\pi^2/16)\Big)^{-k/\log T}.$$

Now we prove the first claim by showing that the lower bound is realized for a particular set W. Since the case of $d = 2$ can be embedded as a subset for general $d \ge 2$ as the left top block, it suffices to show it for $d = 2$. We further constrain the set W and only consider those G with representation

$$G = \begin{bmatrix} a & b \\ -b & a \end{bmatrix},$$

where $a, b \in \mathbb{R}$. The eigenvalues of this matrix are complex numbers $a - jb$ and $a + jb$, which satisfy $\rho(G) \le 1$ if $a^2 + b^2 \le 1$, where $\rho(G)$ is the spectral radius of G. The nice property of this type of matrices is that there exists an explicit expression of G^i for integer $i \ge 2$. Define complex number $z = a + jb$, then for integer $i \ge 0$:

$$G^i = \begin{bmatrix} \Re(z^i) & \Im(z^i) \\ -\Im(z^i) & \Re(z^i) \end{bmatrix},$$

where $\Re(z)$ represents the real part of complex number z, and $\Im(z)$ represents the imaginary part of z.

We want to approximate $\mu(G) \in \mathbb{R}^{2\times 2T}$ by $\sum_{i=1}^{k} a_i u_i$, where $a_i \in \mathbb{R}^{2\times 2}$ and $u_i \in \mathbb{R}^{2\times 2T}$. Let W_1 be the subset of row vectors realized by the first row of $\mu(G)$ for all $G \in \mathbb{R}^{2\times 2}$ with $\rho(G) \leq 1$. We use the following property: the 2-norm of a matrix is lower bounded by the 2-norm of one of its rows. Based on this property, the 2-norm of error in approximating $\mu(G)$ is lower bounded by the 2-norm of error in approximating only one row of $\mu(G)$. Therefore, the generalized k-width of approximating $\mu(G)$ in 2-norm is lower bounded by the error of approximating the first row of $\mu(G)$ by a linear combination of $2k$ row vectors with dimension $2T$. In other words,

$$d_{2k}(W_1) \leq d_k(W). \tag{2.22}$$

To see this, denote by $u_i(1), u_i(2) \in \mathbb{R}^{2T}$ the first and second row of matrix u_i, respectively. The first row of $\mu(G)$ can be written as $\sum_{i=1}^{k} a_i(1,1)u_i(1)+a_i(1,2)u_i(2)$, a linear comibination of $2k$ row vectors, where $a_i(1,1), a_i(1,2)$ are the elements of the first row of matrix a_i.

To lower bound the generalized Kolmogorov width of the constrained set W_1, we consider a relaxed *weighted* version of the width. Precisely, let p be a probability measure on the set W_1, then the *weighted squared deviation* of W_1 from U under weight p is defined as

$$d_{2k}^2(W_1; p) \triangleq \inf_{U\in\mathcal{U}_{2k}} \mathbb{E}_{x\sim p} \inf_{y\in U} \|x-y\|^2 \leq \inf_{U\in\mathcal{U}_{2k}} \sup_{x\in W_1} \inf_{y\in U} \|x-y\|^2 = d_{2k}^2(W_1). \tag{2.23}$$

We observe that $d_{2k}^2(W_1; p)$ in general can be computed using spectral methods. Indeed, for the subset U, the y that achieves $\inf_{y\in U}\|x-y\|^2$ can be computed via a projection matrix $\hat{y} = U_{2k}U_{2k}^\top x$, where U_{2k} consists of $2k$ columns of orthonormal vectors. We now have

$$\begin{aligned}
\mathbb{E}_{x\sim p} \inf_{y\in Q} \|x-y\|^2 &= \mathbb{E}_{x\sim p}\|x - U_{2k}U_{2k}^\top x\|^2 \\
&= \mathbb{E}_{x\sim p}[x^\top x - x^\top U_{2k}U_{2k}^\top x] \\
&= \mathrm{tr}((I - U_{2k}U_{2k}^\top)\mathbb{E}_{x\sim p}[xx^\top]).
\end{aligned}$$

The minimizer U_{2k} of $\mathrm{tr}((I - U_{2k}U_{2k}^\top)\mathbb{E}_{x\sim p}[xx^\top])$ is the same as the maximizer of $\mathrm{tr}(U_{2k}U_{2k}^\top\mathbb{E}_{x\sim p}[xx^\top])$, which is given by the first $2k$ eigenvectors of $\mathbb{E}_{x\sim p}[xx^\top]$, and the value of the weighted squared generalized k-width is given by the sum of all eigenvalues of $\mathbb{E}_{x\sim p}[xx^\top]$ except for the first largest $2k$ eigenvalues (Lemma 2.8.3).

We compute the weighted squared generalized k-width of the constrained set W_1, and it would serve as a lower bound of the squared generalized k-width. We choose the probability measure of $(a,b)^\top \in \mathbb{R}^2$ as the uniform measure on the unit circle. We compute the matrix $\mathbb{E}_{x\sim p}[xx^\top]$, which is $\mathbb{E}[\mu_1(G)^\top\mu_1(G)]$, where $\mu_1(G)$ is the first row of $\mu(G)$. Concretely, we write $\mu_1(G) = [\nu_0; \nu_1; \ldots; \nu_{T-1}]$ for $\nu_l \in \mathbb{R}^2$ and equal to

$$\nu_l = [\Re(z^l), \Im(z^l)],$$

where $z = a + jb$ and for all $l \in \{0, 1, \ldots, T-1\}$.

We claim that $\mathbb{E}[\nu_l \nu_m^\top] = 0$ whenever $l \neq m$. Indeed, when $l \neq m$, each of the 4 entries of matrix $\mathbb{E}[\nu_l \nu_m^\top]$ are of the form either $\Re(z^l)\Re(z^m)$, $\Im(z^l)\Im(z^m)$, or $\Re(z^l)\Im(z^m)$ for some $l \neq m$. For the complex number $z = re^{j\theta}$, we know $z^l = r^l e^{jl\theta}$ which implies that $\Re(z^l) = r^l \cos(l\theta)$ and $\Im(z^l) = r^l \sin(l\theta)$. We now compute $\mathbb{E}[r^l \cos(l\theta) r^m \sin(m\theta)]$ for $l \neq m, l \geq 0, m \geq 0$ and other cases can be computed analogously. Since we are considering a uniform distribution on the unit circle, $r \equiv 1$. We have

$$\begin{aligned}
&\int_{\theta \in [0,2\pi]} \cos(k\theta)\sin(m\theta)\frac{1}{2\pi} d\theta \\
&= \frac{1}{2\pi}\int_0^{2\pi} \frac{1}{2}(\sin((k+m)\theta) + \sin((k-m)\theta)) d\theta \\
&= 0.
\end{aligned}$$

Hence, it suffices to only compute $\mathbb{E}[\nu_l \nu_l^\top]$

$$\mathbb{E}[\nu_l \nu_l^\top] = \frac{1}{2}\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}. \tag{2.24}$$

Therefore, $\mathbb{E}[\mu_1(G)^\top \mu_1(G)] = 0.5 I_{2T}$. Using Lemma 2.8.3 $d_{2k}^2(W_1; p = \mathcal{U})$ is equal to the sum of bottom $2T - 2k$ eigenvalues: $d_{2k}^2(W_1; p = \mathcal{U}) = (2T - 2k)/2 = T - k$. By (2.22) and (2.23)

$$d_k(W) \geq d_{2k}(W_1) \geq d_{2k}(W_1; p = \mathcal{U}) = \sqrt{T - k}.$$

□

2.8.6 Convex relaxation analysis: Proof of Theorem 2.3

Recall matrix $\widetilde{\Theta}$ defined in (2.12). The following theorem is a formal restatement of Theorem 2.3 that analyzes the approximation error due to convex relaxation.

Theorem 2.3. (Convex relaxation error bound) *Denote by m_t, the one-step-ahead predictions made by the best linear predictor (Kalman filter) for system (4). Let $R_P = \max\{\|B\|_2, \|C\|_2, \|D\|_2\}$, $R_C = \max\{\|P\|_2, \|Q\|_2, \|R\|_2, \|K\|_2\}$, and $\|x_t\|_2 \leq R_x$. Suppose that $\|A^t\|_2 \leq \gamma t^{\log(\gamma)}$ for a bounded constant $\gamma \geq 1$. Let $C_0 = 43, C_1 = 520$. For any $\epsilon, \delta > 0$, if the number of filters k satisfies*

$$k \geq \frac{\pi^2}{8}\log(T)\log\left(\frac{12C_0 d^2(1+R_P^2)^3(2R_x^2 + R_C)(1+R_C^2)(1+\gamma)^4(mT + \log(1/\delta))T^{3+2\log(\gamma)}}{\epsilon}\right),$$

then the following holds for $\widetilde{\Theta}$

$$\mathbb{P}\left[\|\widetilde{\Theta} f_t - m_t\|_2^2 \geq \epsilon\right] \leq \delta. \tag{2.25}$$

Proof. Denote by $G = U\Lambda U^{-1}$ the eigendecomposition of matrix G, where $\Lambda = \mathsf{diag}(\lambda_1, \dots \lambda_d)$ are eigenvalues of G. Let v_l be the columns of U and $w_l^\top$ be rows of U^{-1}. Write

$$\begin{aligned}
m_t =& \sum_{i=1}^{t-1} CG^{t-i-1}Ky_i + \sum_{i=1}^{t-1} CG^{t-i-1}(B-KD)x_i + Dx_t \\
=& \sum_{i=1}^{t-1} CU\Lambda^{t-i-1}U^{-1}Ky_i + \sum_{i=1}^{t-1} CU\Lambda^{t-i-1}U^{-1}(B-KD)x_i + Dx_t \\
=& \sum_{i=1}^{t-1} CU\Big[\sum_{l=1}^{d}(\lambda_l^{t-i-1})e_l \otimes e_l\Big]U^{-1}Ky_i + \sum_{i=1}^{t-1} CU\Big[\sum_{l=1}^{d}(\lambda_l^{t-i-1})e_l \otimes e_l\Big]U^{-1}(B-KD)x_i + Dx_t \\
=& \sum_{l=1}^{d} CUe_l \otimes e_l U^{-1}K \sum_{i=1}^{t-1}\lambda_l^{t-i-1}y_i + \sum_{l=1}^{d} CUe_l \otimes e_l U^{-1}(B-KD)\sum_{i=1}^{t-1}\lambda_l^{t-i-1}x_i + Dx_t \\
=& \sum_{l=1}^{d} Cv_l w_l^\top K \sum_{i=1}^{t-1}\lambda_l^{t-i-1}y_i + \sum_{l=1}^{d} Cv_l w_l^\top (B-KD)\sum_{i=1}^{t-1}\lambda_l^{t-i-1}x_i + Dx_t.
\end{aligned}$$

Let $Y_t = [y_1, \dots, y_t] \in \mathbb{R}^{m\times t}$ and $X_t = [x_1, \dots, x_t] \in \mathbb{R}^{n\times t}$. We can write m_t and $\widetilde{m}_t$ as

$$\begin{aligned}
m_t =& \sum_{l=1}^{d} Cv_l w_l^\top K Y_{t-1}\mu_{t-1:1}(\lambda_l) + \sum_{l=1}^{d} Cv_l w_l^\top (B-KD)X_{t-1}\mu_{t-1:1}(\lambda_l) + Dx_t, \\
\widetilde{m}_t =& \sum_{l=1}^{d} Cv_l w_l^\top K Y_{t-1}\widetilde{\mu}_{t-1:1}(\lambda_l) + \sum_{l=1}^{d} Cv_l w_l^\top (B-KD)X_{t-1}\widetilde{\mu}_{t-1:1}(\lambda_l) + Dx_t.
\end{aligned}$$

We write $b_t = m_t - \widetilde{m}_t$ using the PCA reconstruction error $r_t = \mu_t - \widetilde{\mu}_t$

$$b_t = m_t - \widetilde{m}_t = \sum_{i=1}^{d} Cv_i w_i^\top K Y_{t-1} r_{t-1:1}(\lambda_i) + Cv_i w_i^\top (B-KD)X_{t-1}r_{t-1:1}(\lambda_i).$$

The Euclidean norm of bias is bounded by

$$\begin{aligned}
\|b_t\|_2 &\le \Big(\sum_{i=1}^{d}\|C\|_2\|v_i w_i^\top\|_2\|K\|_2\|Y_{t-1}\|_2 + \|C\|_2\|v_i w_i^\top\|_2(\|B\|_2 + \|K\|_2\|D\|_2)\|X_{t-1}\|_2\Big)\sup_\lambda\|r(\lambda)\|_2 \\
&\le \Big(dR_P R_C\|Y_{t-1}\|_2 + dR_P^2(1+R_C)\|X_{t-1}\|_2\Big)\sup_\lambda\|r(\lambda)\|_2 \\
&\le \Big(dR_P R_C\|Y_{t-1}\|_2 + dR_P^2(1+R_C)\sqrt{t}R_x\Big)\Big(C_0 T\sqrt{\log T}c^{-k/\log T}\Big)^{1/2}.
\end{aligned}$$

The first inequality uses simple properties such as sub-multiplicative and sub-additive properties of norm. The second inequality uses the upper bound assumptions on parameters.

The third inequality is due to Lemma 2.8.5 where $c = \exp(\pi^2/8)$ and $C_0 = 43$. The squared approximation error is given by

$$\|b_t\|_2^2 \leq 2d^2(1+R_C^2)(1+R_P^2)^2\Big(\|Y_{t-1}\|_2^2 + tR_x^2\Big)C_0 T\sqrt{\log T}c^{-k/\log T}.$$

Observe that $\|Y_{1:t}\|_2^2 \leq \|Y_{1:t}\|_F^2 = \|y_{1:t}\|_2^2$. By (2.20), the following holds with probability greater than $1-\delta$

$$\|b_t\|_2^2 \leq 12d^2(1+R_P^2)^3(2R_x^2+R_C)(1+R_C^2)(1+\gamma)^4(mT+\log(1/\delta))T^{3+2\log(\gamma)}c^{-k/\log T}.$$

We finish the proof by setting the number of filters k such that the error is smaller than ϵ, i.e.

$$k \geq \frac{\log T}{\log c}\log\Big(\frac{12C_0d^2(1+R_P^2)^3(2R_x^2+R_C)(1+R_C^2)(1+\gamma)^4(mT+\log(1/\delta))T^{3+2\log(\gamma)}}{\epsilon}\Big).$$

□

Informally, the above theorem states that choosing $k \asymp_M \log(T)\log(T/\epsilon)$ is sufficient to ensure an approximation error smaller than ϵ.

2.8.7 Regret decomposition

Recall the definitions of innovation e_t and model bias b_t

$$e_t = y_t - \mathbb{E}[y_t|y_{1:t-1}, x_{1:t}] = y_t - m_t \qquad \text{and} \qquad b_t = \widetilde{\Theta}f_t - m_t = \widetilde{m}_t - m_t, \tag{2.26}$$

where m_t is the predictions made by the Kalman filter in hindsight and $\widetilde{\Theta}$ is defined in (2.12). Let $\hat{m}_t$ be the predictions made by the algorithm. Regret can be written as

$$\begin{aligned}
\text{Regret}(T) &= \sum_{t=1}^{T}\|y_t - \hat{m}_t\|_2^2 - \|y_t - m_t\|_2^2 \\
&= \sum_{t=1}^{T}\|m_t + e_t - \hat{m}_t\|_2^2 - \|e_t\|_2^2 \\
&= \sum_{t=1}^{T}\|\hat{m}_t - m_t\|_2^2 - \sum_{t=1}^{T}2e_t^\top(\hat{m}_t - m_t) \\
&= \mathcal{L}(T) - \sum_{t=1}^{T}2e_t^\top(\hat{m}_t - m_t),
\end{aligned}$$

where $\mathcal{L}(T)$ is the squared error between the Kalman filter predictions and algorithm predictions defined in (2.13). Recall the following notation

$$Z_t \triangleq \alpha I + \sum_{i=1}^{t} f_i f_i^\top, \qquad E_t \triangleq \sum_{i=1}^{t} e_i f_i^\top, \qquad B_t \triangleq \sum_{i=1}^{t} b_i f_i^\top.$$

The error between the predictions made by our algorithm and Kalman filter can be written as

$$\hat{m}_t - m_t = \hat{\Theta}^{(t)} f_t - \widetilde{\Theta} f_t + b_t = \Big(\sum_{i=1}^{t-1} y_i f_i^\top \Big) Z_{t-1}^{-1} f_t - \widetilde{\Theta} f_t + b_t,$$

The second equation uses the update rule of $\hat{\Theta}^{(t)}$ given in (2.10). Simple algebraic manipulations give

$$\begin{aligned}
&\Big(\sum_{i=1}^{t-1} y_i f_i^\top \Big) Z_{t-1}^{-1} f_t - \widetilde{\Theta} f_t + b_t \\
&= \Big(\sum_{i=1}^{t-1} [\widetilde{\Theta} f_i + b_i + e_i] f_i^\top \Big) Z_{t-1}^{-1} f_t - \widetilde{\Theta} f_t + b_t \\
&= \Big(\sum_{i=1}^{t-1} [\widetilde{\Theta} f_i f_i^\top + b_i f_i^\top + e_i f_i^\top] \Big) Z_{t-1}^{-1} f_t - \widetilde{\Theta} f_t + b_t \\
&= \Big(\sum_{i=1}^{t-1} [\widetilde{\Theta} (f_i f_i^\top + \frac{\alpha}{t-1} I - \frac{\alpha}{t-1} I) + b_i f_i^\top + e_i f_i^\top] \Big) Z_{t-1}^{-1} f_t - \widetilde{\Theta} f_t + b_t \\
&= \widetilde{\Theta} \Big(\alpha I + \sum_{i=1}^{t-1} f_i f_i^\top \Big) Z_{t-1}^{-1} f_t - \alpha \widetilde{\Theta} Z_{t-1}^{-1} f_t + \Big(\sum_{i=1}^{t-1} b_i f_i^\top \Big) Z_{t-1}^{-1} f_t + \Big(\sum_{i=1}^{t-1} e_i f_i^\top \Big) Z_{t-1}^{-1} f_t - \widetilde{\Theta} f_t + b_t \\
&= \widetilde{\Theta} Z_{t-1} Z_{t-1}^{-1} f_t - \alpha \widetilde{\Theta} Z_{t-1}^{-1} f_t + B_{t-1} Z_{t-1}^{-1} f_t + E_{t-1} Z_{t-1}^{-1} f_t - \widetilde{\Theta} f_t + b_t \\
&= E_{t-1} Z_{t-1}^{-1} f_t + B_{t-1} Z_{t-1}^{-1} f_t + b_t - \alpha \widetilde{\Theta} Z_{t-1}^{-1} f_t.
\end{aligned}$$

We apply the RMS-AM inequality to obtain an upper bound on $\mathcal{L}(T)$

$$\begin{aligned}
\mathcal{L}(T) &= \sum_{t=1}^{T} \|\hat{m}_t - m_t\|_2^2 \\
&= \sum_{t=1}^{T} \|E_{t-1} Z_{t-1}^{-1} f_t + B_{t-1} Z_{t-1}^{-1} f_t + b_t - \alpha \widetilde{\Theta} Z_{t-1}^{-1} f_t\|_2^2 \\
&\leq \sum_{t=1}^{T} 3\|E_{t-1} Z_{t-1}^{-1} f_t\|_2^2 + 3\|B_{t-1} Z_{t-1}^{-1} f_t + b_t\|_2^2 + 3\|\alpha \widetilde{\Theta} Z_{t-1}^{-1} f_t\|_2^2.
\end{aligned}$$

Regret can thus be decomposed to the following terms

$$\begin{aligned}
\text{Regret}(T) \leq & \sum_{t=1}^{T} 3\|E_{t-1}Z_{t-1}^{-1}f_t\|_2^2 && \text{(least squares error)} \\
& + \sum_{t=1}^{T} 3\|B_{t-1}Z_{t-1}^{-1}f_t + b_t\|_2^2 && \text{(improper learning bias)} \\
& + \sum_{t=1}^{T} 3\|\alpha\widetilde{\Theta}Z_{t-1}^{-1}f_t\|_2^2 && \text{(regularization error)} \\
& - \sum_{t=1}^{T} 2e_t^\top(\hat{m}_t - m_t) && \text{(innovation error)}
\end{aligned}$$

We bound each of the first three terms by extracting a $\|Z_{t-1}^{-1/2}f_t\|_2^2$, i.e. we write

$$\begin{aligned}
\|E_{t-1}Z_{t-1}^{-1}f_t\|_2^2 &\leq \sup_{1\leq t\leq T}\|E_{t-1}Z_{t-1}^{-1/2}\|_2^2\sum_{t=1}^{T}\|Z_{t-1}^{-1/2}f_t\|_2^2, \\
\|B_{t-1}Z_{t-1}^{-1}f_t + b_t\|_2^2 &\leq \sup_{1\leq t\leq T}\|B_{t-1}Z_{t-1}^{-1/2}\|_2^2\sum_{t=1}^{T}\|Z_{t-1}^{-1/2}f_t\|_2^2 + \sum_{t=1}^{T}\|b_t\|_2^2, \\
\|\alpha\widetilde{\Theta}Z_{t-1}^{-1}f_t\|_2^2 &\leq \sup_{1\leq t\leq T}\|\alpha\widetilde{\Theta}Z_{t-1}^{-1/2}\|_2^2\sum_{t=1}^{T}\|Z_{t-1}^{-1/2}f_t\|_2^2.
\end{aligned}$$

In subsequent sections, we compute a high probability upper bound on $\sum_{t=1}^{T}\|Z_{t-1}^{-1/2}f_t\|_2^2$ as well as the specific terms in the above decomposition that affect least squares error, improper learning bias, regularization error, and innovation error, proving that regret is bounded by polylog(T).

2.8.8 Regret analysis

High probability bound on $\det(Z_t)$

We start by deriving an upper bound on $\log(\det(Z_t))$ as this quantity appears multiple times when analyzing regret. The following lemma provides a high probability bound on $\det(Z_t)$ for features defined in (2.9).

Lemma 2.8.7. (High probability upper bounds on $\det(\mathbf{Z_t})$) *Assume as in Lemma 2.8.2 and let* $Z_t = \alpha I + \sum_{i=1}^{t} f_t f_t^\top$. *Then, for any* $\delta \geq 0$

$$\mathbb{P}\left(\log(\det(Z_t)) \geq l\log\left[\alpha^2 + 8k(R_P^2+1)(R_x^2+R_C)(1+\gamma)^4(mt+\log\left(\frac{1}{\delta}\right))t^{3+2\log(\gamma)}\right]\right) \leq \delta.$$

Proof. Let l be the feature vector dimension. We have

$$Z_t = \alpha I + \sum_{i=1}^{t} f_t f_t^\top \preceq \alpha I + \sum_{i=1}^{t} (f_i^\top f_i) I \quad \Rightarrow \quad \det(Z_t) \leq \left(\alpha^2 + \sum_{i=1}^{t} \|f_i\|_2^2\right)^l.$$

Recall the definition $\Psi_t = [\psi_t, \ldots, \psi_1]$ from Algorithm 1 and the compact representation for input features $\widetilde{x}_t = (\Psi_t \otimes I_n) x_{1:t}$ and output features $\widetilde{y}_t = (\Psi_t \otimes I_n) y_{1:t}$. Observe that $\|\Psi_t\|_2 \leq 1$ since Ψ_t is a block of eigenvector matrix of hankel matrix H. Thus the feature norm is bounded by

$$\begin{aligned} \|f_t\|_2^2 &= \|\widetilde{y}_{t-1}\|_2^2 + \|\widetilde{x}_{t-1}\|_2^2 + \|x_t\|_2^2 \\ &\leq k\|y_{1:t-1}\|_2^2 + k\|x_{1:t-1}\|_2^2 + R_x^2 \\ &\leq k\|y_{1:t}\|_2^2 + 2ktR_x^2. \end{aligned}$$

From Lemma 2.8.2, with probability at least $1 - \delta$

$$\begin{aligned} \|f_t\|_2^2 &\leq 6k(R_P^2 + 1)(R_x^2 + R_C)(1+\gamma)^4 (mt + \log\left(\frac{1}{\delta}\right)) t^{2+2\log(\gamma)} + 2ktR_x^2 \\ &\leq 8k(R_P^2 + 1)(R_x^2 + R_C)(1+\gamma)^4 (mt + \log\left(\frac{1}{\delta}\right)) t^{2+2\log(\gamma)}. \end{aligned}$$

The above bound is increasing in t, therefore

$$\mathbb{P}\left(\det(Z_t) \geq \left[\alpha^2 + 8k(R_P^2 + 1)(R_x^2 + R_C)(1+\gamma)^4 (mt + \log\left(\frac{1}{\delta}\right)) t^{3+2\log(\gamma)}\right]^l\right) \leq \delta.$$

□

Given the PAC bound parameters M, if $k \asymp_M \text{polylog}(T)$ (and hence $l = (m+n)k + n \asymp_M \text{polylog}(T)$), then the above lemma states that $\log(\det(Z_t)) \lesssim_M \text{polylog}(T)$.

Self-normalizing vector martingales

We now prove a key result on vector self-normalizing martingales that is used multiple times throughout our regret analysis. The result is inspired by Theorem 1 of Abbasi-Yadkori et al. (2011), which provides a bound for self-normalizing martingales with scalar sub-Gaussian noise, and extend it to vector-valued sub-Gaussian noise with arbitrary covariance.

Theorem 2.8.1. (Bound on self-normalized vector martingale) *Let $\{\mathcal{F}_t\}_{t=0}^{\infty}$ be a filtration. Let $e_t \in \mathbb{R}^m$ be $\mathcal{F}_t$ measurable and $e_t|\mathcal{F}_{t-1}$ to be conditionally R_V-sub-Gaussian. In other words, for all $t \geq 0$ and $\omega \in \mathbb{R}^m$*

$$\mathbb{E}[\exp(\omega^\top e_t) \mid \mathcal{F}_{t-1}] \leq \exp(R_V^2 \|\omega\|_2^2 / 2).$$

Let $f_t \in \mathbb{R}^l$ be an $\mathcal{F}_{t-1}$-measurable stochastic process. Assume that Z is an $l \times l$ positive definite matrix. For any $t \geq 0$, define

$$Z_t = Z_0 + \sum_{i=1}^{t} f_t f_t^\top \qquad \text{and} \qquad E_t = \sum_{i=1}^{t} e_i f_i^\top.$$

Then, for any $\delta > 0$ and for all $t \geq 0$

$$\mathbb{P}\left[\|E_t Z_t^{-1/2}\|_2 \leq 8R_V^2 m + 4R_V^2 \log\Big(\frac{\det(Z_t)^{1/2}\det(Z_0)^{-1/2}}{\delta}\Big)\right] \geq 1-\delta.$$

Proof. We use an ϵ-net argument. First, we establish control over $\|\omega^\intercal E_t Z_t^{-1/2}\|_2$ for all vectors ω in unit sphere $\mathcal{S}^{m-1}$. We will discretize the sphere using a net and finish by taking a union bound over all ω in the net.

Let $\mathcal{N}$ be an ϵ-net of unit sphere $\mathcal{S}^{m-1}$ and set $\epsilon = 1/2$. Corollary 4.2.13 in Vershynin (2018) states that the covering number for unit sphere $\mathcal{S}^{m-1}$ is given by

$$|\mathcal{N}| \leq \Big(\frac{2}{\epsilon}+1\Big)^m = 5^m.$$

$\omega^\top e_i$ is R_V-sub-Gaussian for any $\omega \in \mathcal{N}$. Therefore, for any $\omega \in \mathcal{N}$ and any $u \geq 0$, Theorem 1 in Abbasi-Yadkori et al. (2011) yields

$$\mathbb{P}\left[\|\omega^\top E_t Z_t^{-1/2}\|_2 \geq u\right] \leq \det(Z_t)^{1/2}\det(Z_0)^{-1/2}\exp\big(-\frac{u}{2R_V^2}\big).$$

Using Lemma 4.4.1 in Vershynin (2018), we have

$$\|E_t Z_t^{-1/2}\|_2 \leq 2\sup_{\omega\in\mathcal{N}}\|\omega^\top E_t Z_t^{-1/2}\|_2.$$

Taking a union bound over $\mathcal{N}$, we conclude that

$$\begin{aligned}
\mathbb{P}\left[\|E_t Z_t^{-1/2}\|_2 \geq u\right] &\leq \mathbb{P}\left[\sup_{\omega\in\mathcal{N}}\|\omega^\top E_t Z_t^{-1/2}\|_2 \geq \frac{u}{2}\right] \\
&\leq \sum_{\omega\in\mathcal{N}} \mathbb{P}\left[\|\omega^\top E_t Z_t^{-1/2}\|_2 \geq \frac{u}{2}\right] \\
&\leq \det(Z_t)^{1/2}\det(Z_0)^{-1/2}\exp\big(2m - \frac{u}{4R_V^2}\big).
\end{aligned}$$

□

The above theorem combined with the result of Lemma 2.8.7 immediately implies that for $k \asymp_M \text{polylog}(T)$, we have $\|E_t Z_t^{-1/2}\|_2 \lesssim_M \text{polylog}(T)$ and $\|B_t Z_t^{-1/2}\|_2 \lesssim_M \text{polylog}(T)$ with high probability.

High probability bound on $\|Z_{t-1}^{-1/2} f_t\|_2^2$

In this section, we show that $\sum_{t=1}^{T} \|Z_{t-1}^{-1/2} f_t\|_2^2 \lesssim_M \text{polylog}(T)$. The proof steps are summarized below.

Step 1. We show a high probability Löwner upper bound on $f_t f_t^\top$ in terms of $\alpha_0 I + \mathbb{E}[f_t f_t^\top]$.

Step 2. We state the *block-martingale small-ball condition* and show that the process $\{f_t\}$ satisfies this condition. We prove a high probability lower bound on Z_t in terms of the conditional covariance $\text{cov}(f_{s+i} \mid \mathcal{F}_i)$ for large enough s.

Step 3. We define a *filter quadratic function condition* and prove that under this condition, there exists $c_T \asymp_M \text{polylog}(T)$ such that $Z_t - \frac{1}{c_T} f_{t+1} f_{t+1}^\top \succeq 0$. By Schur complement lemma, this is equivalent to $\|Z_t^{-1/2} f_{t+1}\|_2 \leq c_T \asymp_M \text{polylog}(T)$.

Step 1. The following lemma establishes a high probability upper bound on $f_t f_t^\top$ based on the covariance of feature vector f_t.

Lemma 2.8.8. (High probability upper bound on $\mathbf{f_t f_t^\top}$) *Let f_t be a zero-mean Gaussian random vector in $\mathbb{R}^l$ and let $\Sigma_t = \alpha_0 I + \mathbb{E}[f_t f_t^\top]$ for a real $\alpha_0 > 0$. Then, for any $\delta > 0$ and $\alpha_0 > 0$*

$$\mathbb{P}\left(f_t f_t^\top \preceq [2l + 4\log(1/\delta)]\Sigma_t\right) \geq 1 - \delta,$$

and if Σ_t is invertible, the results holds for $\alpha_0 = 0$.

Proof. Consider the random vector $\Sigma_t^{-1/2} f_t$. Jensen's inequality gives

$$\mathbb{E}\|\Sigma_t^{-1/2} f_t\|_2 \leq \sqrt{\mathbb{E}[f_t^\top \Sigma_t^{-1} f_t]} = \sqrt{\text{tr}(\Sigma_t^{-1} E[f_t f_t^\top])} \leq \sqrt{l}.$$

By standard bounds on tails of sub-gaussian random variables (for example, see Exercise 6.3.5 in Vershynin (2018)), for any $\delta > 0$

$$\mathbb{P}\left(\|\Sigma_t^{-1/2} f_t\|_2 > \sqrt{l} + \sqrt{2\log\frac{1}{\delta}}\right) \leq \delta$$

Let $c = 2l + 4\log\frac{1}{\delta}$. Then, the above bound implies

$$\mathbb{P}(f_t^\top \Sigma_t^{-1} f_t \leq c) \geq 1 - \delta.$$

Using Schur complement method, $c - f_t^\top \Sigma_t^{-1} f_t \geq 0$ if and only if the following matrix is positive semi-definite

$$\begin{bmatrix} \Sigma_t & f_t \\ f_t^\top & c \end{bmatrix} \succeq 0.$$

Using the other Schur complement, this is only true if and only if $\Sigma_t - \frac{1}{c} f_t f_t^\top \succeq 0$, which concludes the proof. □

Step 2. To capture the excitation behavior of features, we use the martingale small-ball condition (Mendelson, 2014; Simchowitz et al., 2018).

Definition 2.2. (Martingale small-ball) *Let $\{f_t\}_{t\geq 1}$ be an $\mathcal{F}_t$-adapted random processes taking values in $\mathbb{R}^l$. We say that $\{f_t\}_{t\geq 1}$ satisfies the (s, Γ_{sb}, p)-block martingale small-ball (BMSB) condition for $\Gamma_{sb} \succ 0$ if for any $t \geq 1$ and for any fixed ω in unit sphere $\mathcal{S}^{l-1}$*

$$\frac{1}{s}\sum_{i=1}^{s} \mathbb{P}(|w^\top f_{t+i}| \geq \sqrt{w^\top \Gamma_{\text{sb}} w} \mid \mathcal{F}_t) \geq p.$$

To show the process $\{f_t\}_{t\geq 1}$ satisfy a BMSB condition, we first show that the conditional covariance of features is increasing in the positive semi-definite cone.

Lemma 2.8.9. (Monotonicity of conditional covariance of features) *Let $\phi_1, \ldots, \phi_k$ for $k \leq T$ be a set of T-dimensional orthogonal vectors and let $\psi_i = [\phi_1(i), \ldots, \phi_k(i)]^\top$ be a k-dimensional vector. Consider system (4) and define the following for all $t \geq 2$*

$$f_t = \psi_1 \otimes y_{t-1} + \cdots + \psi_{t-1} \otimes y_1. \tag{2.27}$$

Let $\mathcal{F}_t = \sigma\{\eta_0, \ldots, \eta_{t-1}, \zeta_1, \ldots, \zeta_t\}$. Then, $cov(f_{t+i}|\mathcal{F}_t)$ is independent of t and increases with i in the positive semi-definite cone.

Proof. Expanding y_i in definition of f_t in (2.27) based on system (2.4), we have

$$\begin{aligned} f_{t+i} - \mathbb{E}[f_{t+i} \mid \mathcal{F}_t] = &(\psi_1 \otimes C)\eta_{t+i-2} \\ &+ (\psi_2 \otimes C + \psi_1 \otimes CA)\eta_{t+i-3} \\ &+ \ldots \\ &+ (\psi_{i-1} \otimes C + \cdots + \psi_1 \otimes CA^{i-2})\eta_t \\ &+ \psi_1 \otimes \zeta_{t+i-1} + \cdots + \psi_{i-1} \otimes \zeta_{t+1} \end{aligned}$$

Recall that $\mathbb{E}[\eta_t \eta_t^\top] = Q, \mathbb{E}[\zeta_t \zeta_t^\top] = R$ and that the process noise and the observation noise are i.i.d. Therefore,

$$\begin{aligned} \text{cov}(f_{t+i}|\mathcal{F}_t) = &(\psi_1 \otimes C)Q(\psi_1 \otimes C)^\top \\ &+ (\psi_2 \otimes C + \psi_1 \otimes CA)Q(\psi_2 \otimes C + \psi_1 \otimes CA)^\top \\ &+ \ldots \\ &+ (\psi_{i-1} \otimes C + \cdots + \psi_1 \otimes CA^{i-2})Q(\psi_{i-1} \otimes C + \cdots + \psi_1 \otimes CA^{i-2})^\top \\ &+ \psi_1 \otimes R\psi_1^\top \otimes I_m + \cdots + \psi_{i-1} \otimes R\psi_{i-1}^\top \otimes I_m. \end{aligned} \tag{2.28}$$

Observe that the conditional covariance is independent of t. Furthermore, all terms in the above sum are positive semi-definite; increasing i only adds two additional positive semi-definite terms. It follows that

$$\text{cov}(f_{t+i+1}|\mathcal{F}_t) \succeq \text{cov}(f_{t+i}|\mathcal{F}_t).$$

□

Equipped with the result of the above lemma, we now show that $\{f_t\}_{t\geq 1}$ satisfy a BMSB condition.

Lemma 2.8.10. (BMSB condition) *Consider the process $\{f_t\}_{t\geq 1}$ defined in Lemma 2.8.9 and let $\Gamma_i = cov(f_{t+i}|\mathcal{F}_t)$. For any $1 \leq s \leq T$, the process $\{f_t\}_{t\geq 1}$ satisfies the $(s, \Gamma_{s/2}, 3/20)$-BMSB condition.*

Proof. Note that $\omega^\top f_{t+i} \mid \mathcal{F}_t$ has a Gaussian distribution with variance $\sqrt{\omega^\top \Gamma_i \omega}$. By an application of Paley-Zygmund inequality, one has

$$\mathbb{P}(|w^\top f_{t+i}| \geq \sqrt{w^\top \Gamma_i w} \mid \mathcal{F}_t) \geq \mathbb{P}(|w^\top f_{t+i} - \mathbb{E}[w^\top f_{t+i} \mid \mathcal{F}_t]| \geq \sqrt{w^\top \Gamma_i w} \mid \mathcal{F}_t) \geq \frac{3}{10}$$

Let $1 \leq s' \leq s$. By Lemma 2.8.9, Γ_i is increasing in i. Therefore,

$$\begin{aligned}
\frac{1}{s}\sum_{i=1}^{s} \mathbb{P}(|w^\top f_{t+i}| \geq \sqrt{w^\top \Gamma_{s'} w}|\mathcal{F}_t) &\geq \frac{1}{s}\sum_{i=s'}^{s} \mathbb{P}(|w^\top f_{t+i}| \geq \sqrt{w^\top \Gamma_{s'} w}|\mathcal{F}_t) \\
&\geq \frac{1}{s}\sum_{i=s'}^{s} \mathbb{P}(|w^\top f_{t+i}| \geq \sqrt{w^\top \Gamma_{i} w}|\mathcal{F}_t) \quad (\Gamma_i \text{ increasing}) \\
&\geq \frac{3}{10}\frac{s-s'+1}{s}. \qquad \text{(Paley-Zygmund)}
\end{aligned}$$

Choosing $s' = s/2$ shows that f_t satisfies $(s, \Gamma_{s/2}, 3/20)$ small-ball condition. □

The small-ball condition can be used to establish high probability lower bound on $\sigma_{\min}(Z_t)$, as shown by the following lemma.

Lemma 2.8.11. (Lower bound on $\mathbf{Z_t}$) *Consider the process $\{f_t\}_{t\geq 1}$ defined in Lemma 2.8.9 and let $Z_t = \alpha I + \sum_{i=1}^{t} f_i f_i^\top$ for regularization parameter $\alpha > 0$. For $\delta, \alpha_0 > 0$ let*

$$\Gamma_i = \text{cov}(f_{t+i}|\mathcal{F}_i), \quad \Gamma_{\max} = t[2l + 4\log(2/\delta)][\alpha_0 I + \Gamma_t].$$

For any $\delta > 0$ if s satisfies the following

$$s \leq \frac{tp^2/10}{\log\det(\Gamma_{\max}) - l\log(\alpha) - \log(2/\delta)},$$

then

$$\mathbb{P}\left(Z_t \succeq \frac{\alpha}{2} I + \frac{s\lfloor t/s \rfloor p^2 \Gamma_{s/2}}{16}\right) \geq 1 - \delta.$$

Proof. According to Lemma 2.8.10, $\{f_t\}_{t\geq 1}$ satisfies the $(s, \Gamma_{s/2}, p = 3/20)$-BMSB condition. The following lemma from Simchowitz et al. (2018) gives tail probabilities for real-valued processes that satisfy a small-ball condition. Note that our notation for small ball condition in real-valued processes slightly differs from Simchowitz et al. (2018) which results in a slight difference in the statement of the lemma below.

Lemma 2.8.12. (Tail bounds for small-ball processes) *If a real-valued process* $\{z_t\}_{t\geq 1}$ *satisfies the* (s,σ,p)*-BMSB condition, then*

$$\mathbb{P}(\sum_{i=1}^{t} z_i^2 \leq \frac{p^2\sigma}{8} s\lfloor t/s \rfloor) \leq \exp\Big(-\frac{\lfloor t/s\rfloor p^2}{8}\Big).$$

For a fixed $\omega \in \mathcal{S}^{l-1}$, the process $\{\omega^\top f_t\}_{t\geq 1}$ satisfies $(s, \omega^\top \Gamma_{s/2}\omega, p)$. Using the above lemma, we have

$$\mathbb{P}\left(\omega^\top(\sum_{i=1}^{t} f_i f_i^\top)\omega \leq \frac{p^2\omega^\top\Gamma_{s/2}\omega}{8} s\lfloor t/s\rfloor\right) \leq \exp\Big(-\frac{\lfloor t/s\rfloor p^2}{8}\Big).$$

For large enough t, we can convert this high probability bound to obtain a uniform Löwner lower bound on Z_t by a discretization argument.

Given a regularization parameter $\alpha > 0$, define

$$\Gamma_{\min} = \alpha I + \frac{s\lfloor t/s\rfloor p^2 \Gamma_{s/2}}{8}$$

Define the following events

$$\mathcal{E}_1 = \Big\{Z_t \succeq \frac{\Gamma_{\min}}{2}\Big\} \quad \text{and} \quad \mathcal{E}_2 = \Big\{Z_t \preceq \Gamma_{\max}\Big\}.$$

We have $\mathbb{P}(\mathcal{E}_1^c) \leq \mathbb{P}(\mathcal{E}_1^c \cap \mathcal{E}_2) + \mathbb{P}(\mathcal{E}_2^c)$, where $\mathbb{P}(\mathcal{E}_2^c)$ is bounded by $\delta/2$ according to Lemma 2.8.8. Let $\mathcal{S}_{\Gamma_{\text{sb}}} = \{\omega : \omega^\top\Gamma_{\text{sb}}\omega = 1\}$ and let $\mathcal{T}$ be a 1/4-net of $\mathcal{S}_{\Gamma_{\text{sb}}}$ in the norm $\|\Gamma_{\max}^{1/2}(.)\|_2$. By Lemma 4.1 and Lemma D.1 in Simchowitz et al. (2018), we can write

$$\begin{aligned}
\mathbb{P}(\mathcal{E}_1^c \cap \mathcal{E}_2) &= \mathbb{P}\left(\Big\{Z_t \not\succeq \frac{\Gamma_{\min}}{2}\Big\} \cap \Big\{Z_t \preceq \Gamma_{\max}\Big\}\right) \\
&\leq \mathbb{P}\left(\Big\{\exists\omega \in \mathcal{T} : \|Z_t\omega\|^2 < \omega^\top\Gamma_{\min}\omega\Big\} \cap \Big\{Z_t \preceq \Gamma_{\max}\Big\}\right) \\
&\leq \exp\Big(-\frac{\lfloor t/s\rfloor p^2}{8} + \log\det(\Gamma_{\max}\Gamma_{\min}^{-1})\Big) \\
&\leq \exp\Big(-\frac{tp^2}{10s} + \log\frac{\det(\Gamma_{\max})}{\alpha^l}\Big)
\end{aligned}$$

Setting s such that the above probability is bounded by $\delta/2$

$$s \leq \frac{tp^2/10}{\log\det(\Gamma_{\max}) - l\log(\alpha) + \log(2/\delta)},$$

we conclude that $\mathbb{P}(\mathcal{E}_1^c) \leq \delta/2 + \delta/2 = \delta$. □

Step 3. So far we have computed a lower bound on Z_t and an upper bound on $f_t f_t^\top$ and our goal is to show that there exists $c_T \asymp_M \text{polylog}(T)$ such that $Z_t - \frac{1}{c_T} f_t f_t^\top \succeq 0$. This inequality, however, does not hold for any set of orthonormal filters $\phi_1, \dots, \phi_k$. We identify an assumption connecting filters with transition matrix A that ensures $Z_t - \frac{1}{c_T} f_t f_t^\top \succeq 0$. This assumption is based on a *filter quadratic function*, which we restate below.

Definition 2.3. (Filter quadratic function) Let $\phi_1, \dots, \phi_k$ for $k \leq T$ be a set of T-dimensional vectors, let $\psi_i = [\phi_1(i), \dots, \phi_k(i)]^\top$ be a k-dimensional vector, and let $\psi_i^{(d)} = \psi_i \otimes I_d$, for any $d \geq 1$. For any matrix $A \in \mathbb{R}^{d \times d}$, the following matrix is called the *filter quadratic function* of ψ with respect to A

$$\begin{aligned}\Omega_t(A; \psi) = (\psi_1^{(d)})(\psi_1^{(d)})^\top + (\psi_2^{(d)} + \psi_1^{(d)} A)(\psi_2^{(d)} + \psi_1^{(d)} A)^\top + \dots \\ + (\psi_{t-1}^{(d)} + \dots + \psi_1^{(d)} A^{t-2})(\psi_{t-1}^{(d)} + \dots + \psi_1^{(d)} A^{t-2})^\top.\end{aligned}$$

In the following lemma, we show that a condition on filter quadratic function implies $t\Gamma_{s/2} - \Gamma_{t+1}/c_0 \succeq 0$ for a constant c_0.

Lemma 2.8.13. (Filter quadratic condition) *Assume as in Lemma 2.8.9 and let κ be the maximum condition number of Q and R. For any A, if there exists $t_0 \geq 1$ for which there exists s such that*

$$t\Omega_{s/2}(A; \psi) - \Omega_{t+1}(A; \psi) \succeq 0, \qquad \forall t \geq t_0,$$

then $t\Gamma_{s/2} - \Gamma_{t+1}/c_0 \succeq 0$, where $c_0 \geq \kappa$.

Proof. Let $\psi_i^{(m)} = \psi_i \otimes I_m$. Recall the expression of the conditional covariance of f_t given in (2.28):

$$\begin{aligned}\Gamma_t &= (\psi_1^{(m)} C) Q (\psi_1^{(m)} C)^\top \\ &\quad + (\psi_2^{(m)} C + \psi_1^{(m)} CA) Q (\psi_2^{(m)} C + \psi_1^{(m)} CA)^\top \\ &\quad + \dots \\ &\quad + (\psi_{t-1}^{(m)} C + \dots + \psi_1^{(m)} CA^{t-2}) Q (\psi_{t-1}^{(m)} C + \dots + \psi_1^{(m)} CA^{t-2})^\top \\ &\quad + \psi_1^{(m)} R (\psi_1^{(m)})^\top + \dots + \psi_{t-1}^{(m)} R (\psi_{t-1}^{(m)})^\top\end{aligned}$$

Define the following terms

$$\Gamma_t^{(Q)} \triangleq (\psi_1^{(m)} C) Q (\psi_1^{(m)} C)^\top + \dots + (\psi_{t-1}^{(m)} C + \dots + \psi_1^{(m)} CA^{t-2}) Q (\psi_{t-1}^{(m)} C + \dots + \psi_1^{(m)} CA^{t-2})^\top,$$
$$\Gamma_t^{(R)} \triangleq \psi_1^{(m)} R (\psi_1^{(m)})^\top + \dots + \psi_{t-1}^{(m)} R (\psi_{t-1}^{(m)})^\top,$$

where $\Gamma_t = \Gamma_t^{(Q)} + \Gamma_t^{(R)}$. In order to show $t\Gamma_{s/2} - \Gamma_{t+1}/c_0 \succeq 0$, it is sufficient to show

$$t\Gamma_{s/2}^{(Q)} - \frac{1}{c_0}\Gamma_{t+1}^{(Q)} \succeq 0 \quad \text{and} \quad t\Gamma_{s/2}^{(R)} - \frac{1}{c_0}\Gamma_{t+1}^{(R)} \succeq 0.$$

Let $R_C = \max\{\|R\|_2, \|Q\|_2\}$ and $\sigma_r = \min\{\sigma_{\min}(Q), \sigma_{\min}(R)\}$. For $t\Gamma_{s/2}^{(R)} - \frac{1}{c_0}\Gamma_{t+1}^{(R)}$, we have

$$\begin{aligned}&\sigma_r[\psi_1^{(m)}(\psi_1^{(m)})^\top + \cdots + \psi_t^{(m)}(\psi_t^{(m)})^\top] \\ \preceq &\psi_1^{(m)} R(\psi_1^{(m)})^\top + \cdots + \psi_t^{(m)} R(\psi_t^{(m)})^\top \\ \preceq &R_C[\psi_1^{(m)}(\psi_1^{(m)})^\top + \cdots + \psi_t^{(m)}(\psi_t^{(m)})^\top].\end{aligned}$$

Setting $c_0 = R_C/\sigma_r$, gives

$$\begin{aligned}&t\Gamma_{s/2}^{(R)} - \frac{1}{c_0}\Gamma_{t+1}^{(R)} \\ \succeq &\sigma_r t[\psi_1^{(m)}(\psi_1^{(m)})^\top + \cdots + \psi_{s/2-1}^{(m)}(\psi_{s/2-1}^{(m)})^\top] - \sigma_r[\psi_1^{(m)}(\psi_1^{(m)})^\top + \cdots + \psi_t^{(m)}(\psi_t^{(m)})^\top] \succeq 0.\end{aligned}$$

The last matrix is positive semi-definite based on assumption (2.29) when $A = 0$. For $t\Gamma_{s/2}^{(Q)} - \frac{1}{c_0}\Gamma_{t+1}^{(Q)}$, write

$$\psi_i^{(m)} C = \begin{bmatrix}\phi_i^1 C \\ \phi_i^2 C \\ \vdots \\ \phi_i^k C\end{bmatrix}_{km\times d} = \begin{bmatrix}C & 0 & \dots & 0 \\ 0 & C & \dots & 0 \\ \vdots & & & \\ 0 & 0 & \dots & C\end{bmatrix}_{km\times kd} \begin{bmatrix}\phi_i^1 I_d \\ \phi_i^2 I_d \\ \vdots \\ \phi_i^k I_d\end{bmatrix}_{kd\times d} = \mathbf{C}\psi_i^{(d)}.$$

We have

$$\Gamma_{t+1}^{(Q)} = \mathbf{C}\Big[\psi_1^{(d)} Q(\psi_1^{(d)})^\top + \cdots + (\psi_t^{(d)} + \cdots + \psi_1^{(d)} A^{t-1}) Q(\psi_t^{(d)} + \cdots + \psi_1^{(d)} A^{t-1})^\top\Big]\mathbf{C}^\top$$

By a similar argument and given assumption (2.29), we have $t\Gamma_{s/2}^{(Q)} - \frac{1}{c_0}\Gamma_{t+1}^{(Q)} \succeq 0$. □

Remark 2.3. When A is symmetric ($A = UDU^\top$), the positive semi-definite condition filter quadratic function can be further simplified to $t\Omega_{s/2}(D;\psi) - \Omega_{t+1}(D;\psi) \succeq 0$ for all diagonal matrices D with $|D_{ii}| \leq 1$.

In the following lemma, we show a high probability upper bound on $\|Z_t^{-1/2} f_{t+1}\|_2$.

Lemma 2.8.14. ($\|\mathbf{Z_t^{-1/2}} \mathbf{f_{t+1}}\|_\mathbf{2}$ upper bound) *Assume as in Lemma 2.8.9 and let κ be the maximum condition number of Q and R. Define the following for all $t \geq 1$, regularization parameter $\alpha > 0$, $p = 3/20$, and fix $0 < \alpha_0 \leq 200\alpha$ and $\delta > 0$*

$$Z_t = \alpha I + \sum_{i=1}^{t} f_i f_i^\top, \quad \Gamma_{\max} = t[2km + 4\log(4/\delta)][\alpha_0 I + \Gamma_t], \quad \Gamma_{\min} = \alpha I + \frac{s\lfloor t/s\rfloor p^2 \Gamma_{s/2}}{8}$$

For any A, suppose that there exists $t_0 \geq 1$ for which there exists s such that

$$s \leq \frac{tp^2/10}{\log\det(\Gamma_{\max}) - l\log(\alpha) + \log(4/\delta)}, \quad t\Omega_{s/2}(A;\psi) - \Omega_{t+1}(A;\psi) \succeq 0. \tag{2.29}$$

Then, for all $t \geq t_0$ with probability at least $1 - \delta$

$$\|Z_{t-1}^{-1/2} f_t\|_2^2 \leq 10\kappa(2mk + 4\log(2/\delta))/p^2.$$

Proof. Let $c_T = 10\kappa(2mk + 4\log(2/\delta))/p^2$. With probability at least $1-\delta$, we lower bound $\sum_{i=1}^{t} f_i f_i^\top$ by Lemma 2.8.11 and upper bound $\frac{1}{c_T} f_{t+1} f_{t+1}^\top$ by Lemma 2.8.8

$$\begin{aligned} Z_t - \frac{1}{c_T} f_{t+1} f_{t+1}^\top &= \alpha I + \sum_{i=1}^{t} f_i f_i^\top - \frac{1}{c} f_{t+1} f_{t+1}^\top \\ &\succeq \frac{\alpha}{2} I + \frac{p^2}{10} t\Gamma_{s/2} - \frac{p^2}{10}\alpha_0 I - \frac{p^2}{10}\frac{1}{c_0}\Gamma_{t+1} \\ &\overset{(1)}{\succeq} + \frac{p^2}{10} t\Gamma_{s/2} - \frac{p^2}{10}\frac{1}{c_0}\Gamma_{t+1} \\ &\overset{(2)}{\succeq} 0 \end{aligned}$$

where inequality (1) is due to the assumption $\alpha_0 \leq 200\alpha$ and (2) uses the result of Lemma 2.8.13.

Using Schur complement lemma, $Z_t - \frac{1}{c_T} f_{t+1} f_{t+1}^\top$ is positive semi-definite if and only if the following matrix is positive semi-definite

$$\begin{bmatrix} Z_t & f_{t+1} \\ f_{t+1}^\top & c_T. \end{bmatrix}.$$

Using the other Schur complement, this is true if and only if $c_T - f_{t+1}^\top Z_t^{-1} f_t \geq 0$. Equivalently,

$$Z_t - \frac{1}{c_T} f_{t+1} f_{t+1}^\top \succeq 0 \quad \Leftrightarrow \quad \|Z_t^{-1/2} f_{t+1}\|_2 \leq c_T,$$

which concludes the proof. □

The above lemma states that if $k \asymp_M \text{polylog}(T)$ then $\|Z_{t-1}^{-1/2} f_t\|_2^2 \lesssim_M \text{polylog}(T)$ with high probability.

Proof of Lemma 2.1

We now prove that $\|Z_{t-1}^{-1/2} f_t\|_2^2 \lesssim_M \text{polylog}(T)$ implies $\sum_{t=1}^{T} \|Z_{t-1}^{-1/2} f_t\|_2^2 \lesssim_M \text{polylog}(T)$. We first present a lemma inspired by Lemma 2 of Lai et al. (1982).

Lemma 2.8.15. (Upper bound on $\sum_{\mathbf{i=1}}^{\mathbf{t}} \|\mathbf{Z_i^{-1/2} f_i}\|_{\mathbf{2}}^{\mathbf{2}}$) *Let $f_1, \ldots, f_t$ be l-dimensional vectors and Z_0 an $l \times l$ positive definite matrix. Define $Z_t = Z_0 + \sum_{i=1}^{t} f_i f_i^\top$. Then,*

$$\sum_{i=1}^{t} f_i^\top Z_i^{-1} f_i \leq \log\left(\frac{\det(Z_t)}{\det(Z_0)}\right).$$

Proof. First, note that Z_t is positive definite and has a positive determinant for all $t \geq 1$. Using matrix determinant lemma, we have

$$\det(Z_{t-1}) = \det(Z_t - f_t f_t^\top) = \det(Z_t)(1 - f_t^\top Z_t^{-1} f_t) \Rightarrow f_t^\top Z_t^{-1} f_t = \frac{\det(Z_t) - \det(Z_{t-1})}{\det(Z_t)}$$

Since $Z_i \succeq Z_{i-1}$, we have $\det(Z_i) \geq \det(Z_{i-1})$. We write

$$\sum_{i=1}^{t} f_i^\top Z_i^{-1} f_i = \sum_{i=1}^{t} 1 - \frac{\det(Z_{i-1})}{\det(Z_i)} \leq \sum_{i=1}^{t} \log\Big(\frac{\det(Z_i)}{\det(Z_{i-1})}\Big) = \log\Big(\frac{\det(Z_t)}{\det(Z_0)}\Big),$$

where we used the fact that $1 - x \leq \log(1/x)$ for $x \leq 1$. □

We are now ready to prove Lemma 2.1.

Proof of Lemma 2.1. The first claim is already proved in Lemma 2.8.10. We focus on proving the second claim. Recall the result of Lemma 2.8.15, which states that

$$\sum_{t=1}^{T} f_t^\top Z_t^{-1} f_t \leq \log\Big(\frac{\det(Z_T)}{\det(\alpha I)}\Big).$$

Using matrix determinant lemma, the above is equivalent to

$$\sum_{t=1}^{T} \frac{f_t^\top Z_{t-1}^{-1} f_t}{1 + f_t^\top Z_{t-1}^{-1} f_t} \leq \log\Big(\frac{\det(Z_T)}{\det(\alpha I)}\Big).$$

By Lemma 2.8.7, $\log\det(Z_t)$ is bounded by $\mathrm{polylog}(T)$ with high probability since $k \asymp_M \mathrm{polylog}(T)$. Furthermore, by Lemma 2.8.14, $\|Z_{t-1}^{-1/2} f_t\|_2^2 \lesssim_M \mathrm{polylog}(T)$ with high probability. Concretely,

$$\mathbb{P}(\|Z_{t-1}^{-1/2} f_t\|_2^2 \leq 10\kappa(2mk + 4\log(2/\delta))/p^2) \geq 1 - \delta,$$
$$\mathbb{P}\left(\log(\det(Z_t)) \leq mk \log\left[\alpha^2 + 8k(R_P^2 + 1)(R_x^2 + R_C)(1+\gamma)^4(mt - \log(\delta))t^{3+2\log(\gamma)}\right]\right) \geq 1 - \delta.$$

Therefore, we can apply Lemma 2.8.20 by combining the two bounds and taking a union bound

$$R_Z(T) \triangleq mk \log\left[\alpha^2 + 8k(R_P^2 + 1)(R_x^2 + R_C)(1+\gamma)^4(mT - \log(\delta))T^{3+2\log(\gamma)}\right],$$
$$\mathbb{P}\left\{\sum_{t=1}^{T} \|Z_{t-1}^{-1/2} f_t\|_2^2 \leq \Big(1 + \frac{10\kappa(2mk + 4\log(4/\delta)}{p^2}\Big)\big(R_Z(T) - mk\log(\alpha)\big)\right\} \geq 1 - \delta.$$

□

Regularization term

The following lemma computes an upper bound on the 2-norm of the relaxed model parameters $\widetilde{\Theta}$.

Lemma 2.8.16. (Model parameter bound) *Consider system* (2.4) *and let* k *be the number of spectral filters and* $\widetilde{\Theta}$ *be the parameters defined in* (2.12)*. If* $\|\mathcal{O}_t\|_2, \|\mathcal{C}_t\|_2 \leq R_K$ *and* $\|D\|_2 \leq R_P$ *then,*

$$\|\widetilde{\Theta}\|_2 \leq 2kR_K + R_P.$$

Proof. Parameter matrix $\widetilde{\Theta}$ is the concatenation of coefficients of features $\widetilde{y}_{t-1}, \widetilde{x}_{t-1}, x_t$. By matrix norm properties,

$$\|\widetilde{\Theta}\|_2 \leq \|D\|_2 + \sum_{j=1}^{k} \|\sum_{i=1}^{d} C v_i w_i^\top K \langle \mu(\lambda_i), \phi_j \rangle\|_2 + \|\sum_{i=1}^{d} C v_i w_i^\top (B - KD) \langle \mu(\lambda_i), \phi_j \rangle\|_2.$$

Recall that $\{\lambda_i\}_{i=1}^k$, $\{v_i\}_{i=1}^k$, and $\{w_i^\top\}_{i=1}^k$ are the top k eigenvalues, right eigenvectors, and left eigenvectors of G, respectively. Write

$$\|\sum_{i=1}^{d} C v_i w_i^\top K \langle \mu(\lambda_i), \phi_j \rangle\|_2 = \|\sum_{t=1}^{T} C G^{T-t} K \phi_j(t)\|_2 = \|\mathcal{O}_T \phi_j\|_2 \leq R_K,$$

and similarly,

$$\|\sum_{i=1}^{d} C v_i w_i^\top (B - KD) \langle \mu(\lambda_i), \phi_j \rangle\|_2 = \|\mathcal{C}_T \phi_j\|_2 \leq R_K.$$

Summing all terms gives the final bound. □

Lemma 2.8.17. (Regularization term bound) *Assume as in Lemma 2.8.16 and let* $Z_t = \alpha I + f_t f_t^\top$*. If* $\alpha \leq 1/\|\widetilde{\Theta}\|_2^2$*, then*

$$\|\alpha \widetilde{\Theta} Z_{t-1}^{-1/2}\|_2^2 \leq 1.$$

Proof. The regularization term implies $Z_t \succeq \alpha I$ and thus $\|Z_t^{-1/2}\|_2^2 \leq 1/\alpha$. By norm properties

$$\|\alpha \widetilde{\Theta} Z_{t-1}^{-1/2}\|_2^2 \leq \alpha^2 \|\widetilde{\Theta}\|_2^2 \|Z_{t-1}^{-1/2}\|_2^2 \leq 1.$$

□

Innovation error

The following lemma, based on the analysis given by Tsiamis and Pappas (2020), shows that the innovation error is bounded by $\sqrt{\mathcal{L}(T)}$ (defined in (2.13)).

Lemma 2.8.18. (Innovation error bound) *Let $\mathcal{L}(T) = \sum_{t=1}^{T} \|\hat{m}_t - m_t\|_2^2$ be the squared error between Kalman predictions in hindsight and predictions by Algorithm 1. Assume that the innovation covariance matrix has a bounded norm $\|V\|_2 \leq R_V$. For all $\delta > 0$, the following holds with probability greater than $1 - \delta$:*

$$\sum_{t=1}^{T} 2e_t^\top (\hat{m}_t - m_t) \leq 8R_V^2 \Big(\mathcal{L}(T) + 1\Big)^{1/2} \Big[2 + \log\Big(\frac{\mathcal{L}(T)+1}{\delta}\Big)\Big].$$

Proof. Write

$$\sum_{t=1}^{T} e_t^\top (\hat{m}_t - m_t) = \sum_{t=1}^{T} \sum_{i=1}^{m} e_{t,i}(\hat{m}_{t,i} - m_{t,i}).$$

Let $s = m\lfloor s/m \rfloor + r$ and define the following filtration

$$\mathcal{F}_s = \{e_{1,1}, \ldots, e_{\lfloor s/m \rfloor, r}\}.$$

A scalar version of Theorem 2.8.1 states that the following holds with probability at least $1 - \delta$

$$\Big(\sum_{t=1}^{T} \|\hat{m}_t - m_t\|_2^2 + 1\Big)^{-1/2} \sum_{t=1}^{T} e_t^\top (\hat{m}_t - m_t) \leq 4R_V^2 \Big[2 + \log\Big(\frac{1}{\delta}\Big) + \log\Big(\sum_{t=1}^{T} \|\hat{m}_t - m_t\|_2^2 + 1\Big)\Big].$$

Therefore, with probability at least $1 - \delta$

$$\sum_{t=1}^{T} 2e_t^\top (\hat{m}_t - m_t) \leq 8R_V^2 \Big(\mathcal{L}(T) + 1\Big)^{1/2} \Big[2 + \log\Big(\frac{\mathcal{L}(T)+1}{\delta}\Big)\Big].$$

□

Proof of Theorem 1

Proof of Theorem 2.1. Recall the regret decomposition given in Section 2.8.7:

$$\begin{aligned}\text{Regret}(T) \leq \sup_{1 \leq t \leq T} \Big(\|E_{t-1} Z_{t-1}^{-1/2}\|_2^2 + \|B_{t-1} Z_{t-1}^{-1/2}\|_2^2 + \|\alpha \widetilde{\Theta} Z_{t-1}^{-1/2}\|_2^2\Big) \Big(\sum_{t=1}^{T} \|Z_{t-1}^{-1/2} f_t\|_2^2\Big) \\ + T \sup_{1 \leq t \leq T} \|b_t\|_2^2 - \sum_{t=1}^{T} 2e_t^\top (\hat{m}_t - m_t).\end{aligned}$$

Let $\delta_1 = \delta/8$. We describe bounds on each term in the above regret bound. All lemmas and theorems used in this proof contain explicit dependencies on horizon T as well as PAC bound parameters. While one can combine these results to write a regret bound with explicit dependencies on all parameters, we refrain from writing in such detail here for a clear presentation.

Bounding $\|\mathbf{E_{t-1}Z_{t-1}^{-1/2}}\|_\mathbf{2}^\mathbf{2}$. According to Theorem 2.8.1, with probability at least $1-\delta_1$, the term $\|E_{t-1}Z_{t-1}^{-1/2}\|_2^2$ is bounded by

$$\|E_{t-1}Z_{t-1}^{-1/2}\|_2 \lesssim \text{poly}(R_\Theta, m)\Big[\log(1/\delta_1) + \log(\det(Z_t)) - l\log(\alpha)\Big],$$

$l = (m+n)k+n$ is the feature vector dimension. We substitute the regularization parameter α and the number of filters k according to Theorem 2.1 assumption (iii). Given the values for k, α and by Lemma 2.8.7, with probability at least $1-\delta_1$ we have

$$\log(\det(Z_t)) \lesssim \text{poly}(R_\Theta, m, \beta)\,\text{polylog}(\gamma, \frac{1}{\delta_1})\log^3(T).$$

Taking a union bound gives

$$\mathbb{P}\left[\|E_{t-1}Z_{t-1}^{-1/2}\|_2^2 \lesssim \text{poly}(R_\Theta, m, \beta)\,\text{polylog}(\gamma, \frac{1}{\delta_1})\log^6(T)\right] \geq 1-2\delta_1. \tag{2.30}$$

Bounding $\|\mathbf{B_{t-1}Z_{t-1}^{-1/2}}\|_\mathbf{2}^\mathbf{2}$. Recall the definitions $B_t = \sum_{i=1}^t b_i f_i^\top$ from (2.15) and $b_i = \widetilde{\Theta} f_t - m_t$ from (2.26). We choose the number of filters k to satisfy (2.25) with failure probability $\delta_1 > 0$ and $\epsilon = 1/T$,[3] which results in $k \gtrsim_M \log^2(T)$ satisfied by assumption (iii). Therefore, we can apply Theorem 2.3 which states that $\|b_t\|_2^2 \leq 1/T$ with probability at least $1-\delta_1$. Combining this result with the result of Theorem 2.8.1 with a union bound yields

$$\mathbb{P}\left[\|\Big(\sum_{i=1}^{t-1} b_i f_i^\top\Big)Z_{t-1}^{-1/2}\|_2 \leq \frac{4}{T}(2m + \log\Big(\frac{\det(Z_t)^{1/2}\det(\alpha I_l)^{-1/2}}{\delta_1}\Big))\right] \geq 1-2\delta_1.$$

With a similar argument used in bounding $\|E_{t-1}Z_{t-1}^{-1/2}\|_2^2$, we have

$$\mathbb{P}\left[\|B_{t-1}Z_{t-1}^{-1/2}\|_2^2 \lesssim \text{poly}(R_\Theta, m, \beta)\,\text{polylog}(\gamma, \frac{1}{\delta_1})\frac{\log^6(T)}{T}\right] \geq 1-3\delta_1. \tag{2.31}$$

Bounding $\|\mathbf{f}\widetilde{\mathbf{\Theta}}\mathbf{Z_{t-1}^{-1/2}}\|_\mathbf{2}^\mathbf{2}$. By assumption (iii) and as a result of Lemma 2.8.17, we have

$$\|\alpha\widetilde{\Theta}Z_{t-1}^{-1/2}\|_2^2 \lesssim 1.$$

[3] Setting $\epsilon = 1/T$ is later used for a uniform bound on $\|b_t\|_2^2$ and is not critical in this part of the proof.

Bounding $\sum_{\mathbf{t=1}}^{\mathbf{T}}\|\mathbf{Z}_{\mathbf{t-1}}^{\mathbf{-1/2}}\mathbf{f_t}\|_{\mathbf{2}}^{\mathbf{2}}$. Lemma 2.1 provides the following bound on the excitation term

$$\mathbb{P}\Big[\sum_{t=1}^{T}\|Z_{t-1}^{-1/2}f_t\|_2^2 \lesssim \kappa\,\mathrm{poly}(R_\Theta, m, \beta)\,\mathrm{polylog}(\gamma, \frac{1}{\delta_1})\log^5(T)\Big] \geq 1-\delta_1, \tag{2.32}$$

where the number filters k is substituted by assumption (iii).

Bounding $\mathbf{T}\sup_{\mathbf{1\leq t\leq T}}\|\mathbf{b_t}\|_{\mathbf{2}}^{\mathbf{2}}$. Applying Theorem 2.3 with parameters $\delta_1 > 0, \epsilon = 1/T$, we have

$$\mathbb{P}\left[T\sup_{1\leq t\leq T}\|b_t\|_2^2 \leq T\epsilon \leq 1\right] \geq 1-\delta_1. \tag{2.33}$$

Recall from Section 2.8.7 that $\mathcal{L}(T)$ is bounded by

$$\mathcal{L}(T) \leq \sup_{1\leq t\leq T}\Big(\|E_{t-1}Z_{t-1}^{-1/2}\|_2^2 + \|B_{t-1}Z_{t-1}^{-1/2}\|_2^2 + \|\alpha\widetilde{\Theta}Z_{t-1}^{-1/2}\|_2^2\Big)\Big(\sum_{t=1}^{T}\|Z_{t-1}^{-1/2}f_t\|_2^2\Big) + T\sup_{1\leq t\leq T}\|b_t\|_2^2.$$

Lemma 2.8.18 with δ_1 states that

$$\mathbb{P}\left[\sum_{t=1}^{T} e_t^\top(\hat{m}_t - m_t) \lesssim \mathrm{poly}(R_\Theta)\,\mathrm{polylog}\left(\frac{1}{\delta_1}\right)\sqrt{\mathcal{L}(T)+1}\right] \geq 1-\delta_1. \tag{2.34}$$

Combining the bounds given in (2.30), (2.31), (2.32), (2.33), (2.34), taking a union probability bound, and setting $\delta = 8\delta_1$ gives

$$\mathbb{P}\left[\mathrm{Regret}(T) \leq \kappa\log^{11}(T)\,\mathrm{poly}(R_\Theta, \beta, m)\,\mathrm{polylog}(\gamma, \frac{1}{\delta})\right] \geq 1-\delta.$$

□

2.8.9 Auxiliary lemmas

In this section, we present a few lemmas that we use throughout the theoretical analysis of our algorithm, presented here for completeness.

The following lemma provides an upper bound on the norm of block Toeplitz matrices (Tsiamis and Pappas, 2019).

Lemma 2.8.19. (Triangular Block Toeplitz Norm) *Let $\mathcal{T}_i \in \mathbb{R}^{m_1,m_2}$ for $i = 1, 2, \ldots, n$. Define the following triangular block Toeplitz matrix*

$$\mathcal{T} = \begin{bmatrix} \mathcal{T}_1 & \mathcal{T}_2 & \mathcal{T}_3 & \ldots & \mathcal{T}_{n-1} & \mathcal{T}_n \\ 0 & \mathcal{T}_1 & \mathcal{T}_2 & \ldots & \mathcal{T}_{n-2} & \mathcal{T}_{n-1} \\ \vdots & & & & & \\ 0 & 0 & 0 & \ldots & \mathcal{T}_1 & \mathcal{T}_2 \\ 0 & 0 & 0 & \ldots & 0 & \mathcal{T}_1 \end{bmatrix}.$$

Then,

$$\|\mathcal{T}\|_2 \leq \sum_{i=1}^{n} \|\mathcal{T}_i\|_2.$$

The following is a simple result for upper bounding a series.

Lemma 2.8.20. *Let $t \in \mathbb{N}$ and let z_t to be a non-negative sequence bounded by a non-decreasing poly-logarithmic function $g(t)$. Suppose that the following sum*

$$\sum_{t=1}^{T} \frac{z_t}{1 + z_t}$$

is bounded by $h(T)$, a non-decreasing poly-logarithmic function of T. Then, $\sum_{t=1}^{T} z_t$ is bounded by a non-decreasing function poly-logarithmic in T.

Proof. Let $z_m = \max_{t \in \{1,\ldots,T\}} z_i$. We have $z_m \leq g(m) \leq g(T)$. Therefore,

$$\sum_{t=1}^{T} z_t \leq \sum_{t=1}^{T} \frac{1 + z_m}{1 + z_t} z_t \leq (1 + g(T)) \sum_{t=1}^{T} \frac{z_t}{1 + z_t} \leq (1 + g(T))h(T), \tag{2.35}$$

which is the desired conclusion. □

Chapter 3

Learning to Make Decisions from a Dataset

In this chapter we shift our focus to the problem of decision-making in sequential environments. Reinforcement learning (RL) is a popular learning framework for sequential decision-making that have recently achieved tremendous empirical success including beating Go champions (Silver et al., 2016, 2017) and surpassing professionals in Atari games (Mnih et al., 2013, 2015), to name a few. Most success stories, however, are in the realm of online RL in which active data collection is necessary. This online paradigm falls short of leveraging previously-collected datasets and dealing with scenarios where online exploration is not possible (Fu et al., 2020). To tackle these issues, offline (or batch) reinforcement learning (Lange et al., 2012; Levine et al., 2020) arises in which the agent aims at achieving competence by exploiting a batch dataset without access to online exploration. This paradigm is useful in a diverse array of application domains such as healthcare (Wang et al., 2018; Gottesman et al., 2019; Nie et al., 2020), autonomous driving (Yurtsever et al., 2020; Bojarski et al., 2016; Pan et al., 2017), and recommendation systems (Strehl et al., 2010; Garcin et al., 2014; Thomas et al., 2017).

The key component of offline RL is a pre-collected dataset from an unknown stochastic environment. Broadly speaking, there exist two types of *data composition* for which offline RL algorithms have shown promising empirical and theoretical success; see Figure 3.1 for an illustration.

- **Expert data.** One end of the spectrum includes datasets collected by following an expert policy. For such datasets, imitation learning algorithms (e.g., behavior cloning (Ross and Bagnell, 2010)) are shown to be effective in achieving a small sub-optimality competing with the expert policy. In particular, it is recently shown in the work Rajaraman et al. (2020) that the behavior cloning algorithm achieves the minimal sub-optimality $1/N$ in episodic Markov decision processes, where N is the total number of samples in the expert dataset.

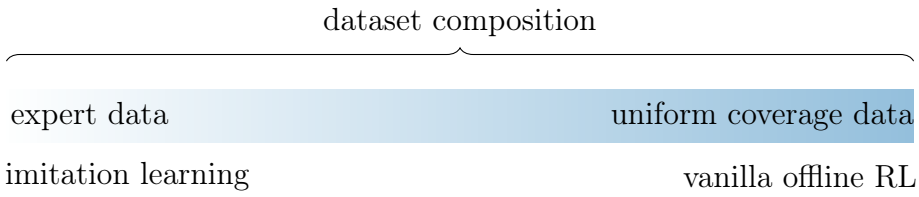

Figure 3.1: Dataset composition range for offline RL problems. On one end, we have expert data for which imitation learning algorithms are well-suited. On the other end, we have uniform exploratory data for which vanilla offline RL algorithms can be used.

- **Uniform coverage data.** On the other end of the spectrum lies the datasets with uniform coverage. More specifically, such datasets are collected with an aim to cover *all* states and actions, even the states never visited or actions never taken by satisfactory policies. Most vanilla offline RL algorithms are only suited in this region and are shown to diverge for *narrower* datasets (Fu et al., 2020; Koh et al., 2020), such as those collected via human demonstrations or hand-crafted policies, both empirically (Fujimoto et al., 2019b; Kumar et al., 2019) and theoretically (Agarwal et al., 2020d; Du et al., 2020). In this regime, a widely-adopted requirement is the *uniformly bounded concentrability coefficient* which assumes that the ratio of the state-action occupancy density induced by *any policy* and the data distribution is bounded uniformly over all states and actions (Munos, 2007; Farahmand et al., 2010; Chen and Jiang, 2019; Xie and Jiang, 2020). Another common assumption is uniformly lower bounded data distribution on all states and actions (Sidford et al., 2018a; Agarwal et al., 2020c), which ensures all states and actions are visited with sufficient probabilities. Algorithms developed for this regime are demonstrated to achieve a $1/\sqrt{N}$ sub-optimality competing with the optimal policy; see for example the papers Yin et al. (2020); Hao et al. (2020); Uehara et al. (2021a).

3.0.1 Motivating questions

Clearly, both of these two extremes impose strong assumptions on the dataset: at one extreme, we hope for a solely expert-driven dataset; at the other extreme, we require the dataset to cover every, even sub-optimal, actions. In practice, there are numerous scenarios where the dataset deviates from these two extremes, which has motivated the development of new offline RL benchmark datasets with different data compositions (Fu et al., 2020; Koh et al., 2020). With this need in mind, the first and foremost question is regarding offline RL formulations:

Question 1 (**Formulation**). *Can we propose an offline RL framework that accommodates the entire data composition range?*

We answer this question affirmatively by proposing a new formulation for offline RL that smoothly interpolates between two regimes: expert data and data with uniform coverage. More specifically, we characterize the data composition in terms of the ratio between the state-action occupancy density of an optimal policy[1] and that of the behavior distribution which we denote by $C^\star$; see Definition 3.1 for a precise formulation. In words, $C^\star$ can be viewed as a measure of the deviation between the behavior distribution and the distribution induced by the optimal policy. The case with $C^\star = 1$ recovers the setting with expert data since, by the definition of $C^\star$, the behavior policy is identical to the optimal policy. In contrast, when $C^\star > 1$, the dataset is no longer purely expert-driven: it could contain "spurious" samples—states and actions that are not visited by the optimal policy. As a further example, when the dataset has uniform coverage, say the behavior probability is lower bounded by $\mu_{\min}$ over all states and actions, it is straightforward to check that the new concentrability coefficient is also upper bounded by $\mu_{\min}^{-1}$.

Assuming a finite $C^\star$ is the weakest concentrability requirement (Scherrer, 2014; Geist et al., 2017; Xie and Jiang, 2020) that is currently enjoyed only by some online algorithms such as CPI (Kakade and Langford, 2002). $C^\star$ imposes a much weaker assumption in contrast to other concentrability requirements which involve taking a maximum over all policies; see Scherrer (2014) for a hierarchy of different concentrability definitions. We would like to immediately point out that existing works on offline RL either do not specify the dependency of sub-optimality on data coverage (Jin et al., 2020c; Yu et al., 2020), or do not have a batch data coverage assumption that accommodates the entire data spectrum including the expert datasets (Yin et al., 2021; Kidambi et al., 2020).

With this formulation in mind, a natural next step is designing offline RL algorithms that handle various data compositions, i.e., for all $C^\star \geq 1$. Recently, efforts have been made toward reducing the offline dataset requirements based on a shared intuition: the agent should act conservatively and avoid states and actions less covered in the offline dataset. Based on this intuition, a variety of offline RL algorithms are proposed that achieve promising empirical results. Examples include model-based methods that learn pessimistic MDPs (Yu et al., 2020; Kidambi et al., 2020; Yu et al., 2021), model-free methods that reduce the Q-functions on unseen state-action pairs (Liu et al., 2020; Kumar et al., 2020; Agarwal et al., 2020e), and policy-based methods that minimize the divergence between the learned policy and the behavior policy (Kumar et al., 2019; Nachum and Dai, 2020; Fujimoto et al., 2019b; Nadjahi et al., 2019; Laroche et al., 2019; Peng et al., 2019; Siegel et al., 2020; Ghasemipour et al., 2020).

However, it is observed empirically that existing policy-based methods perform better when the dataset is nearly expert-driven (toward the left of data spectrum in Figure 3.1) whereas existing model-based methods perform better when the dataset is randomly-collected (toward the right of data spectrum in Figure 3.1) (Yu et al., 2020; Buckman et al., 2020). It remains unclear whether a single algorithm exists that performs well regardless of

[1] In fact, our developments can accommodate arbitrary competing policies, however, we restrict ourselves to the optimal policy for ease of presentation.

data composition—an important challenge from a practical perspective (Kumar and Levine, 2020; Fu et al., 2020; Koh et al., 2020). More importantly, the knowledge of the dataset composition may not be available a priori to assist in selecting the right algorithm. This motivates the second question on the algorithm design:

Question 2 (**Adaptive algorithm design**). *Can we design algorithms that can achieve minimal sub-optimality when facing different dataset compositions (i.e., different $C^\star$)? Furthermore, can this be achieved in an adaptive manner, i.e., without knowing $C^\star$ beforehand?*

To answer the second question, we analyze a *pessimistic* variant of a value-based method in which we first form a lower confidence bound (LCB) for the value function of a policy using the batch data and then seek to find a policy that maximizes the LCB. A similar algorithm design has appeared in the recent work Jin et al. (2020c). It turns out that such a simple algorithm—fully agnostic to the data composition—is able to achieve *almost* optimal performance in multi-armed bandits and Markov decision processes, and optimally solve the offline learning problem in contextual bandits. See the section below for a summary of our theoretical results.

Table 3.1: A summary of our theoretical results with all the log factors ignored.

Multi-armed bandits	$C^\star \in [1, 2)$	$C^\star \in [2, \infty)$
Algorithm 2 (MAB-LCB) sub-optimality (Theorem 3.1)	$\sqrt{\frac{C^\star}{N}}$	$\sqrt{\frac{C^\star}{N}}$
Information-theoretic lower bound (Theorem 3.2)	$\exp\left(-(2-C^\star)\cdot\log\left(\frac{2}{C^\star-1}\right)\cdot N\right)$	$\sqrt{\frac{C^\star}{N}}$
Most played arm (Proposition 3.2)	$\exp\left(-N\cdot\mathsf{KL}\left(\mathrm{Bern}\left(\frac{1}{2}\right)\|\mathrm{Bern}\left(\frac{1}{C^\star}\right)\right)\right)$	N/A
Contextual bandits	$C^\star \in [1, \infty)$	
Algorithm 3 (CB-LCB) sub-optimality (Theorem 3.4)	$\sqrt{\frac{S(C^\star-1)}{N}}+\frac{S}{N}$	
Information-theoretic lower bound (Theorem 3.5)	$\sqrt{\frac{S(C^\star-1)}{N}}+\frac{S}{N}$	
Markov decision processes	$C^\star \in [1, 1+1/N)$	$C^\star \in [1+1/N, \infty)$
Algorithm 4 (VI-LCB) sub-optimality (Theorem 3.6)	$\frac{S}{(1-\gamma)^4 N}$	$\sqrt{\frac{SC^\star}{(1-\gamma)^5 N}}$
Information-theoretic lower bound (Theorem 3.7)	$\sqrt{\frac{S(C^\star-1)}{(1-\gamma)^3 N}}+\frac{S}{(1-\gamma)^2 N}$	$\sqrt{\frac{S(C^\star-1)}{(1-\gamma)^3 N}}+\frac{S}{(1-\gamma)^2 N}$

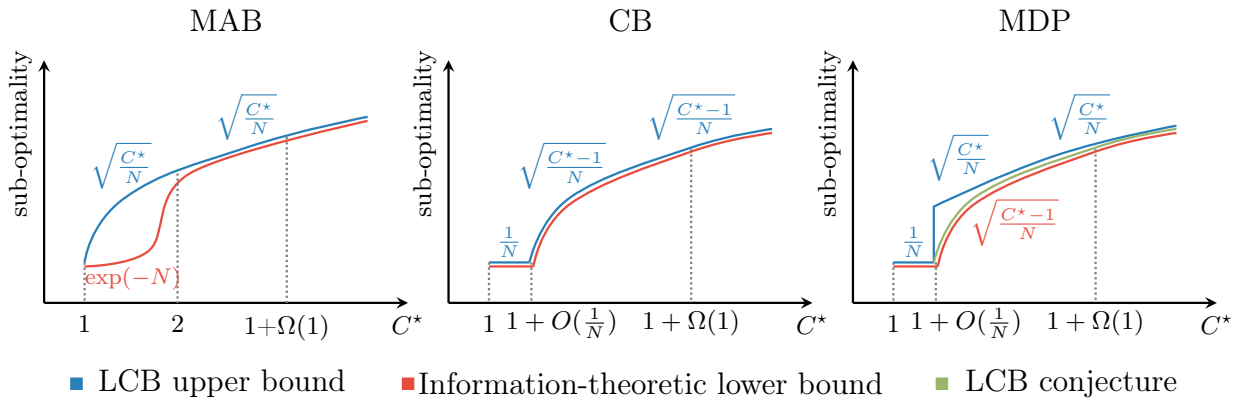

Figure 3.2: The sub-optimality upper bounds and information-theoretic lower bounds for the LCB-based algorithms in MAB, CB with at least two contexts, and MDP settings. In all setting, it is assumed that the knowledge of $C^\star$ is not available to the LCB algorithm.

3.0.2 Main results

In this subsection, we give a preview of our theoretical results; see Table 3.1 for a summary. Under the new framework defined via $C^\star$, we instantiate the LCB approach to three different decision-making problems with increasing complexity: (1) multi-armed bandits, (2) contextual bandits, and (3) infinite-horizon discounted Markov decision processes. We will divide our discussions on the main results accordingly. Throughout the discussion, N denotes the number of samples in the batch data, S denotes the number of states, and we ignore the log factors.

Multi-armed bandits. To address the offline learning problem in multi-armed bandits, LCB starts by forming a lower confidence bound—using the batch data—on the mean reward associated with each action and proceeds to select the one with the largest LCB. We show in Theorem 3.1 that LCB achieves a $\sqrt{C^\star/N}$ sub-optimality competing with the optimal action for all $C^\star \geq 1$. It turns out that LCB is adaptively optimal in the regime $C^\star \in [2, \infty)$ in the sense that it achieves the minimal sub-optimality $\sqrt{C^\star/N}$ without the knowledge of the $C^\star$; see Theorem 3.2. We then turn to the case with $C^\star \in [1, 2)$, in which the optimal action is pulled with more than probability $1/2$. In this regime, it is discovered that the optimal rate has an exponential dependence on N, i.e., e^{-N}, and is achieved by the naive algorithm of selecting the most played arm (cf. Theorem 3.2). To complete the picture, we also prove in Theorem 3.3 that LCB cannot be adaptively optimal for all ranges of $C^\star \geq 1$ if the knowledge of $C^\star$ range is not available.

At first glance, it may seem that LCB for offline RL mirrors upper confidence bound (UCB) for online RL by simply flipping the sign of the bonus. However, our results reveal that the story in the offline setting is much more subtle than that in the online case. Contrary

to UCB that achieves optimal regret in multi-armed bandits (Bubeck et al., 2011), LCB is provably *not* adaptively optimal for solving offline bandit problems under the $C^\star$ framework.

Contextual bandits. The LCB algorithm for contextual bandits shares a similar design to that for multi-armed bandits. However, the performance upper and lower bounds are more intricate and interesting when we consider contextual bandits with at least two states. With regards to the upper bound, we show in Theorem 3.4 that LCB exhibits two different behaviors depending on the data composition $C^\star$. When $C^\star \geq 1 + S/N$, LCB enjoys a $\sqrt{S(C^\star - 1)/N}$ sub-optimality, whereas when $C^\star \in [1, 1 + S/N)$, LCB achieves a sub-optimality with the rate S/N; see Figure 3.2(b) for an illustration. The latter regime ($C^\star \approx 1$) is akin to the imitation learning case where the batch data is close to the expert data. LCB matches the performance of behavior cloning for the extreme case $C^\star = 1$. In addition, in the former regime ($C^\star \geq 1 + S/N$), the performance upper bound depends on the data composition through $C^\star - 1$, instead of $C^\star$. This allows the rate of sub-optimality to smoothly transition from $1/N$ to $1/\sqrt{N}$ as $C^\star$ increases. More importantly, both rates are shown to be minimax optimal in Theorem 3.3, hence confirming the adaptive optimality of LCB for solving offline contextual bandits—in stark contrast to the bandit case. On the other hand, this showcases the advantage of the $C^\star$ framework as it provably interpolates the imitation learning regime and the (non-expert) offline RL regime.

On a technical front, to achieve a tight dependency on $C^\star - 1$, a careful decomposition of the sub-optimality is necessary. In Section 3.3.3, we present the four levels of decomposition of the sub-optimality of LCB that allow us to accomplish the goal. The key message is this: the sub-optimality is incurred by both the value difference and the probability of choosing a sub-optimal action. A purely value-based analysis falls short of capturing the probability of selecting the wrong arm and yields a $1/\sqrt{N}$ rate regardless of $C^\star$. In contrast, the decomposition laid out in Section 3.3.3 delineates the cases in which the value difference (or the probability of choosing wrong actions) plays a bigger role.

Markov decision processes. We combine the LCB approach with the traditional value iteration algorithm to solve the offline Markov decision processes. Ignore the dependence on the effective horizon $1/(1-\gamma)$ for a moment. Similar behaviors to contextual bandits emerge: when $C^\star \in [1, 1+1/N)$, LCB achieves an S/N sub-optimality, and when (say) $C^\star \geq 1.1$, LCB enjoys a $\sqrt{SC^\star/N}$ rate; see Theorem 3.6. Both are shown in Theorem 3.7 to be minimax optimal in their respective regimes of $C^\star$, up to a $1/(1-\gamma)^2$ factor in sample complexity. And this leaves us with an interesting middle ground, i.e., the case when $C^\star \in (1 + 1/N, 1.1)$. Our lower bound still has a dependence $C^\star - 1$ as opposed to $C^\star$ in this regime, and we conjecture that LCB is able to close the gap in this regime.

Conjecture 1 (Adaptive optimality of LCB, Informal)**.** *The LCB approach, together with value iteration is adaptively optimal for solving offline MDPs for all ranges of $C^\star$.*

We discuss the conjecture in detail in Section 3.4.4, where we present an example showing that a variant of value iteration with LCB in the episodic case is able to achieve the optimal dependency on $C^\star$ and hence closing the gap between the upper and the lower bounds. A complete analysis of the LCB algorithm in the episodic MDP setting is presented in Section 3.10.

3.1 Background and problem formulation

We begin with reviewing some core concepts in Markov decision processes in Section 3.1.1. Then we introduce the data collection model and the learning objective for offline RL in Section 3.1.2. In the end, Section 3.1.3 is devoted to the formalization and discussions of the weaker concentrability coefficient assumption that notably allows us to bridge offline RL with imitation learning.

3.1.1 Markov decision processes

Infinite-horizon discounted Markov decision processes. We consider an infinite-horizon discounted Markov decision process (MDP) described by a tuple $M = (\mathcal{S}, \mathcal{A}, P, R, \rho, \gamma)$, where $\mathcal{S} = \{1, \dots, S\}$ is a finite state space, $\mathcal{A} = \{1, \dots, |\mathcal{A}|\}$ is a finite action space, $P : \mathcal{S} \times \mathcal{A} \mapsto \Delta(\mathcal{S})$ is a probability transition matrix, $R : \mathcal{S} \times \mathcal{A} \mapsto \Delta([0, 1])$ encodes a family of reward distributions with $r : \mathcal{S} \times \mathcal{A} \mapsto [0, 1]$ as the expected reward function, $\rho : \mathcal{S} \mapsto \Delta(\mathcal{S})$ is the initial state distribution, and $\gamma \in [0, 1)$ is a discount factor. Upon executing action a from state s, the agent receives a (random) reward distributed according to $R(s, a)$ and transits to the next state s' with probability $P(s'|s, a)$.

Policies and value functions. A stationary deterministic policy $\pi : \mathcal{S} \mapsto \mathcal{A}$ is a function that maps a state to an action. Correspondingly, the value function $V^\pi : \mathcal{S} \mapsto \mathbb{R}$ of the policy π is defined as the expected sum of discounted rewards starting at state s and following policy π. More precisely, we have

$$V^\pi(s) := \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r_t \,\middle|\, s_0 = s, a_t = \pi(s_t) \text{ for all } t \geq 0\right], \qquad \forall s \in \mathcal{S}, \tag{3.1}$$

where the expectation is taken over the trajectory generated according to the transition kernel $s_{t+1} \sim P(\cdot \mid s_t, a_t)$ and reward distribution $r_t \sim R(\cdot \mid s_t, a_t)$. Similarly, the quality function (Q-function or action-value function) $Q^\pi : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ of policy π is defined analogously:

$$Q^\pi(s, a) := \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r_t \,\middle|\, s_0 = s, a_0 = a, a_t = \pi(s_t) \text{ for all } t \geq 1\right] \qquad \forall s \in \mathcal{S}, a \in \mathcal{A}. \tag{3.2}$$

Denote

$$V_{\max} := (1 - \gamma)^{-1}. \tag{3.3}$$

It is easily seen that for any (s, a), one has $0 \leq V^\pi(s) \leq V_{\max}$ and $0 \leq Q^\pi(s, a) \leq V_{\max}$.

Oftentimes, it is convenient to define a scalar summary of the performance of a policy π. This can be achieved by defining the expected value of a policy π:

$$J(\pi) \coloneqq \mathbb{E}_{s\sim\rho}[V^\pi(s)] = \sum_{s\in\mathcal{S}} \rho(s)V^\pi(s). \tag{3.4}$$

It is well known that there exists a stationary deterministic policy $\pi^\star$ that simultaneously maximizes $V^\pi(s)$ for all $s \in \mathcal{S}$, and hence maximizing the expected value $J(\pi)$; see e.g., Puterman (1990, Chapter 6.2.4). We use shorthands $V^\star \coloneqq V^{\pi^\star}$ and $Q^\star \coloneqq Q^{\pi^\star}$ to denote the optimal value function and the optimal Q-function.

Discounted occupancy measures. The (normalized) state discounted occupancy measures $d_\pi : \mathcal{S} \mapsto [0, 1]$ and state-action discounted occupancy measures $d^\pi : \mathcal{S} \times \mathcal{A} \mapsto [0, 1]$ are respectively defined as

$$d_\pi(s) \coloneqq (1-\gamma)\sum_{t=0}^{\infty} \gamma^t \, \mathbb{P}_t(s_t = s; \pi), \qquad \forall s \in \mathcal{S}, \tag{3.5a}$$

$$d^\pi(s, a) \coloneqq (1-\gamma)\sum_{t=0}^{\infty} \gamma^t \, \mathbb{P}_t(s_t = s, a_t = a; \pi), \qquad \forall s \in \mathcal{S}, a \in \mathcal{A}, \tag{3.5b}$$

where we overload notation and write $\mathbb{P}_t(s_t = s; \pi)$ to denote the probability of visiting state $s_t = s$ (and similarly $s_t = s, a_t = a$) at step t after executing policy π and starting from $s_0 \sim \rho(\cdot)$.

3.1.2 Offline data and offline RL

Batch dataset. The current paper focuses on offline RL, where the agent cannot interact with the MDP and instead is given a *batch dataset* $\mathcal{D}$ consisting of tuples (s, a, r, s'), where $r \sim R(s, a)$ and $s' \sim P(\cdot \mid s, a)$. For simplicity, we assume (s, a) pairs are generated i.i.d. according to a data distribution μ over the state-action space $\mathcal{S} \times \mathcal{A}$, which is *unknown* to the agent.[2] Throughout the paper, we denote by $N(s, a) \geq 0$ the number of times a pair (s, a) is observed in $\mathcal{D}$ and by $N = |\mathcal{D}|$ the total number of samples.

The learning objective of offline RL. The goal of offline RL is to find a policy $\hat{\pi}$ — based on the batch data set $\mathcal{D}$— so as to minimize the expected sub-optimality with respect to the optimal policy $\pi^\star$:

$$\mathbb{E}_{\mathcal{D}}\left[J(\pi^\star) - J(\hat{\pi})\right] = \mathbb{E}_{\mathcal{D}}\left[\mathbb{E}_{s\sim\rho}[V^\star(s) - V^{\hat{\pi}}(s)]\right].$$

Here, the expectation is taken with respect to the randomness in the batch data.

[2] The i.i.d. assumption is motivated by the data randomization performed in experience replay (Mnih et al., 2015).

3.1.3 Assumptions on the dataset coverage

Definition 3.1 (Single policy concentrability). *Given a policy π, define C^π to be the smallest constant that satisfies*

$$\frac{d^\pi(s,a)}{\mu(s,a)} \le C^\pi, \qquad \forall s \in \mathcal{S}, a \in \mathcal{A}. \tag{3.6}$$

In words, C^π characterizes the *distribution shift* between the normalized occupancy measure induced by π and data distribution μ. For a stationary deterministic[3] optimal policy, $C^\star := C^{\pi^\star}$ is the "best" *concentrability coefficient* definition which is often much smaller than the widely-used uniform concentrability coefficient $C := \max_\pi C^\pi$ which takes the maximum over all policies π. A small C^π implies that data distribution covers (s,a) pairs visited by policy π, whereas a small C requires the coverage of (s,a) visited by all policies. Further discussion on different assumptions imposed on batch datasets in prior works is postponed to Section 4.5.

3.2 A warm-up: LCB in multi-armed bandits

In this section, we focus on the simplest example of an MDP, the multi-armed bandit model (MAB), to motivate and explain the LCB approach. More specifically, the multi-armed bandit model is a special case of the MDP described in Section 3.1.1 with $S = 1$ and $\gamma = 0$.

In the MAB setting, the offline dataset $\mathcal{D}$ is a set of tuples $\{(a_i, r_i)\}_{i=1}^N$ sampled independently from some joint distribution. Denote the marginal distribution of action a_i as μ. Let $r(a) := \mathbb{E}[r_i \mid a_i = a]$ be the expectation of the reward distribution for action a. Competing with the optimal policy that chooses action $a^\star$, the data coverage assumption simplifies to

$$\frac{1}{\mu(a^\star)} \le C^\star. \tag{3.7}$$

The goal of offline learning in MAB is to select an arm $\hat{a}$ that minimizes the expected sub-optimality

$$\mathbb{E}_{\mathcal{D}}[J(\pi^\star) - J(\hat{\pi})] = \mathbb{E}_{\mathcal{D}}[r(a^\star) - r(\hat{a})].$$

3.2.1 Why does the empirical best arm fail?

A natural choice for identifying the optimal action is to select the arm with the highest empirical mean reward. Mathematically, for all $a \in \mathcal{A}$, let $N(a) := \sum_{i=1}^N \mathbb{1}\{a_i = a\}$ and

$$\hat{r}(a) := \begin{cases} 0, & \text{if } N(a) = 0, \\ \dfrac{1}{N(a)} \displaystyle\sum_{i=1}^N r_i \, \mathbb{1}\{a_i = a\}, & \text{otherwise.} \end{cases}$$

[3] Throughout the paper, when we talk about optimal policies, we restrict ourselves to deterministic stationary policies.

The empirical best arm is then given by $\hat{a} \coloneqq \arg\max_a \hat{r}(a)$.

Though intuitive, the empirical best arm is quite *sensitive* to the arms which have small observation counts $N(a)$: a less-explored sub-optimal arm might have high empirical mean just by chance (due to large variance) and overwhelm the true optimal arm. To see this, let us consider the following scenario.

A failure instance for the empirical best arm. Let $a^\star = 1$ be the optimal arm with a deterministic reward $1/2$. For the remaining sub-optimal arms, we set the reward distribution to be a Bernoulli distributions on $\{0, 1\}$ with mean $1/4$. Consider even the benign case in which the optimal arm is drawn with dominant probability while the sub-optimal ones are sparsely drawn. Under such circumstances, there is a decent chance that one of the sub-optimal arms (say $a = 2$) is drawn for very few times (say just one time) and unfortunately the observed reward is 1, which renders $a = 2$ the empirical best arm. This clearly fails to achieve a low sub-optimality.

Indeed, this intuition about the failure of the empirical best arm can be formalized in the following proposition.

Proposition 3.1 (Failure of the empirical best arm)**.** *For any $\epsilon < 0.05$, $N \geq 500$, there exists a bandit problem with two arms such that for $\hat{a} = \arg\max_a \hat{r}(a)$, one has*

$$\mathbb{E}_{\mathcal{D}}[r(a^\star) - r(\hat{a})] \geq \epsilon.$$

It is worth pointing out that the above lower bound holds for any $\frac{1}{\mu(a^\star)} \leq C^\star$ with $C^\star - 1$ being a constant. See Section 3.7.1 for the proof of Proposition 3.1.

Proposition 3.1 reveals that even in the favorable case when $C^\star \approx 1$, returning the best empirical arm will have a constant error due to the high sensitivity to the less-explored sub-optimal arms. In contrast, the LCB approach, which we will introduce momentarily, will secure a sub-optimality of $\widetilde{O}(\sqrt{1/N})$ in this regime, hence reaching a drastic improvement over the vanilla empirical best arm approach.

3.2.2 LCB: The benefit of pessimism

Revisiting the failure instance for the best empirical arm approach, one soon realizes that it is not sensible to put every action on an equal footing: for the arms that are pulled less often, one should tune down the belief on its empirical mean and be pessimistic on its true reward. Strategically, this principle of pessimism can be deployed with the help of a penalty function $b(a)$ that shrinks as the number of counts $N(a)$ increases. Instead of returning an arm maximizing the empirical reward, the pessimism principle leads us to the following approach: return

$$\hat{a} \in \arg\max_a \ \hat{r}(a) - b(a).$$

Algorithm 2 LCB for multi-armed bandits

Input: Batch dataset $\mathcal{D} = \{(a_i, r_i)\}_{i=1}^N$, and a confidence level $\delta \in (0,1)$.
Set $N(a) = \sum_{i=1}^N \mathbb{1}\{a_i = a\}$ for all $a \in \mathcal{A}$.
for $a \in \mathcal{A}$ **do**
    if $N(a) = 0$ **then**
        Set the empirical mean reward $\hat{r}(a) \leftarrow\!\shortmid 0$.
        Set the penalty $b(a) \leftarrow\!\shortmid 1$.
    else
        Compute the empirical mean reward $\hat{r}(a) \leftarrow\!\shortmid \frac{1}{N(a)} \sum_{i=1}^N r_i \mathbb{1}\{a_i = a\}$.
        Compute the penalty $b(a) \leftarrow\!\shortmid \sqrt{\frac{\log(2|\mathcal{A}|/\delta)}{2N(a)}}$.
Return: $\hat{a} = \arg\max_a \hat{r}(a) - b(a)$.

Intuitively, one could view the right hand side $\hat{r}(a) - b(a)$ as a lower confidence bound (LCB) on the true mean reward $r(a)$. This LCB approach stands on the conservative side and seeks to find an arm with the largest lower confidence bound.

Algorithm 2 shows one instance of the LCB approach for MAB, in which the penalty function originiates from Hoeffding's inequality. We have the following performance guarantee for the LCB approach of Algorithm 2, whose proof can be found in Section 3.7.2.

Theorem 3.1 (LCB sub-optimality, MAB)**.** *Consider a multi-armed bandit and assume that*

$$\frac{1}{\mu(a^\star)} \leq C^\star,$$

for some $C^\star \geq 1$. Suppose that the sample size obeys $N \geq 8C^\star \log N$. Setting $\delta = 1/N$, then action $\hat{a}$ returned by Algorithm 2 obeys

$$\mathbb{E}_{\mathcal{D}}[r(a^\star) - r(\hat{a})] \lesssim \min\left(1, \sqrt{\frac{C^\star \log(2N|\mathcal{A}|)}{N}}\right). \tag{3.8}$$

Applying the performance guarantee (3.8) of LCB on the failure instance used in Proposition 3.1, one sees that LCB achieves a sub-optimality on the order of $\sqrt{(\log N)/N}$, which clearly beats the best empirical arm. This demonstrates the benefit of pessimism over the vanilla approach. Intuitively, the LCB approach applies larger penalties to the actions that are observed only a few times. Even if we have actions with huge fluctuations in their respective empirical rewards due to a small number of samples, the penalty term helps to rule them out.

In fact, our proof yields a stronger high probability performance bound for $\hat{a}$ returned by Algorithm 2: for any $\delta \in (0,1)$, as long as $N \geq 8C^\star \log(1/\delta)$, we have with probability at least $1 - 2\delta$ that

$$r(a^\star) - r(\hat{a}) \leq \min\left(1, 2\sqrt{\frac{C^\star \log(2|\mathcal{A}|/\delta)}{N}}\right). \tag{3.9}$$

Furthermore, for policy π that selects a fixed action a, if $\frac{1}{\mu(a)} \leq C^\pi$ for some C^π, the same analysis gives the following guarantee:

$$\mathbb{E}_{\mathcal{D}}[\max\{0, r(a) - r(\hat{a})\}] \lesssim \min\left(1, \sqrt{\frac{C^\pi \log(2N|\mathcal{A}|)}{N}}\right).$$

This result shows that the LCB algorithm can compete with *any covered* target policy that is not necessarily optimal, i.e., the output policy of the LCB algorithm performs nearly as well as the covered target policy.

3.2.3 Is LCB optimal for solving offline multi-armed bandits?

Given the performance upper bound (3.8) of the LCB approach, it is a natural to ask whether LCB is optimal for solving the bandit problem using offline data. To address this question, we resort to the usual minimax criterion. Since we are dealing with lower bounds, without loss of generality, we assume that the expert always takes the optimal action. Consequently, we can define the following family of multi-armed bandits:

$$\mathsf{MAB}(C^\star) = \{(\mu, R) \mid \frac{1}{\mu(a^\star)} \leq C^\star\}. \tag{3.10}$$

$\mathsf{MAB}(C^\star)$ includes all possible pairs of behavior distribution μ and reward distribution R such that the data coverage assumption $1/\mu(a^\star) \leq C^\star$ holds. It is worth noting that the optimal action $a^\star$ implicitly depends on the reward distribution R. With this definition in place, we define the worst-case risk of any estimator $\hat{a}$ to be

$$\sup_{(\mu,R)\in\mathsf{MAB}(C^\star)} \mathbb{E}_{\mathcal{D}}[r(a^\star) - r(\hat{a})]. \tag{3.11}$$

Here an estimator $\hat{a}$ is simply a measureable function of the data $\{(a_i, r_i)\}_{i=1}^N$ collected under the MAB instance μ and R.

It turns out that LCB is optimal up to a logarithmic factor when $C^\star \geq 2$, as shown in the following theorem.

Theorem 3.2 (Information-theoretic limit, MAB)**.** *For $C^\star \geq 2$, one has*

$$\inf_{\hat{a}} \sup_{(\mu,R)\in\mathsf{MAB}(C^\star)} \mathbb{E}_{\mathcal{D}}[r(a^\star) - r(\hat{a})] \gtrsim \min\left(1, \sqrt{\frac{C^\star}{N}}\right). \tag{3.12}$$

For $C^\star \in (1, 2)$, one has

$$\inf_{\hat{a}} \sup_{(\mu,R)\in\mathsf{MAB}(C^\star)} \mathbb{E}_{\mathcal{D}}[r(a^\star) - r(\hat{a})] \gtrsim \exp\left(-(2 - C^\star) \cdot \log\left(\frac{2}{C^\star - 1}\right) \cdot N\right).$$

See Section 3.7.3 for the proof.

3.2.4 Imitation learning in bandit: The most played arm achieves a better rate

From the above analysis, we know that when $C^\star \geq 2$, the best possible expected sub-optimality is $\sqrt{C^\star/N}$, which is achieved by LCB. On the other hand, if we know that $1/\mu(a^\star) \leq C^\star$ where $C^\star \in [1, 2)$, we can use imitation learning to further improve the rate. The algorithm for bandit is straightforward: pick the arm most frequently selected in dataset, i.e., $\hat{a} = \arg\max_a N(a)$. The performance guarantee of the most played arm is stated in the following proposition.

Proposition 3.2 (Sub-optimality of the most played arm)**.** *Assume that* $\frac{1}{\mu(a^\star)} \leq C^\star$ *for some* $C^\star \in [1, 2)$*. For* $\hat{a} = \arg\max_a N(a)$*, we have*

$$\mathbb{E}_{\mathcal{D}}[r(a^\star) - r(\hat{a})] \leq \exp\left(-N \cdot \mathsf{KL}\left(\mathrm{Bern}\left(\frac{1}{2}\right) \,\middle\|\, \mathrm{Bern}\left(\frac{1}{C^\star}\right)\right)\right). \tag{3.13}$$

The proof is deferred to Section 3.7.4.

When $C^\star \in [1, 2)$, one can see that the rate for the most played arm achieves an exponential dependence on N, whereas the upper bound for LCB is only $1/\sqrt{N}$. On the other hand, the most played arm algorithm completely fails when $C^\star > 2$, while LCB still keeps the rate $1/\sqrt{N}$.

In terms of the dependence on $C^\star$, the KL divergence above evaluates to $\log(C^\star/2) + \log(1/(C^\star - 1))/2$ when the expert policy is optimal. One can see that as $C^\star \to 1$, the rate increases to the order of $1/(C^\star - 1)^N$, which matches the lower bound in Theorem 3.2 in terms of the dependence on $C^\star - 1$.

3.2.5 Non-adaptivity of LCB

One may ask whether LCB can achieve optimal rate under both cases of $C^\star \in [1, 2)$ and $C^\star \geq 2$. Unfortunately, we show in the following theorem that no matter how we set the parameter δ in Algorithm 2, LCB cannot be optimally adaptive in both regimes.

Theorem 3.3 (Non-adaptivity of LCB, MAB)**.** *Let* $C^\star = 1.5$*. There exists a two-armed bandit instance* $(\mu_0, R_0) \in \mathsf{MAB}(C^\star)$ *such that Algorithm 2 with* $L \coloneqq \sqrt{\log(2|\mathcal{A}|/\delta)/2}$ *satisfies*

$$\mathbb{E}_{\mathcal{D}}[r(a^\star) - r(\hat{a})] \gtrsim \min\left(\frac{\sqrt{L}}{N}, \frac{1}{\sqrt{N}}\right) \cdot \exp(-32L).$$

On the other hand, when $C^\star = 6$*, there exists* $(\mu_1, R_1) \in \mathsf{MAB}(C^\star)$ *such that*

$$\mathbb{E}_{\mathcal{D}}[r(a^\star) - r(\hat{a})] \gtrsim \min\left(1, \sqrt{\frac{L}{N}}\right).$$

The proof is deferred to Section 3.7.5.

The theorem above can be understood in the following way: intuitively, a larger L means that we put higher weight on the penalty of the arm instead of the empirical average of the arm. As $L \to \infty$, the LCB algorithm recovers the most played arm algorithm; while as $L \to 0$, the LCB algorithm recovers the empirical best arm algorithm.

When $C^\star \in (1, 2)$, we know from Theorem 3.2 that the most played arm achieves an exponential rate in N. In order to match the rate, we need to select δ such that $L \gtrsim N^\alpha$ for some $\alpha > 0$. However, under this choice of L, the algorithm fails to achieve $1/\sqrt{N}$ rate when $C^\star \geq 6$, which can be done by setting $\delta = 1/N$ (and thus $L = \log(2|\mathcal{A}|N)$) according to Theorem 3.1. This shows that it is impossible for LCB to achieve optimal rate under both cases of $C^\star \in (1, 2)$ and $C^\star \geq 2$ simultaneously.

3.3 LCB in contextual bandits

In this section, we take the analysis one step further by studying offline learning in contextual bandits (CB). As we will see shortly, simply going beyond one state turns the tables in favor of the minimax optimality of LCB.

Formally, contextual bandits can be viewed as a special case of MDP described in Section 3.1.1 with $\gamma = 0$. In CB setting, the batch dataset $\mathcal{D}$ is a set of tuples $\{(s_i, a_i, r_i)\}_{i=1}^N$ sampled independently according to $(s_i, a_i) \sim \mu$, and $r_i \sim R(\cdot \mid s_i, a_i)$. Competing with an optimal policy, the data coverage assumption in the CB case simplifies to

$$\max_s \frac{\rho(s)}{\mu(s, \pi^\star(s))} \leq C^\star.$$

The offline learning objective in CB turns into finding a policy $\hat{\pi}$ based on the batch dataset that minimizes the expected sub-optimality

$$\mathbb{E}_{\mathcal{D}}[J(\pi^\star) - J(\hat{\pi})] = \mathbb{E}_{\mathcal{D},\rho}\left[r(s, \pi^\star(s)) - r(s, \hat{\pi}(s))\right].$$

3.3.1 Algorithm and its performance guarantee

The pessimism principle introduced in the MAB setting can be naturally extended to CB. First, the empirical expected reward is computed for all state-action pairs $(s, a) \in \mathcal{S} \times \mathcal{A}$ according to

$$\hat{r}(s, a) := \begin{cases} 0, & \text{if } N(s, a) = 0, \\ \dfrac{1}{N(s, a)} \displaystyle\sum_{i=1}^N r_i \, \mathbb{1}\{(s_i, a_i) = (s, a)\}, & \text{otherwise.} \end{cases}$$

Algorithm 3 LCB for contextual bandits

Input: Batch dataset $\mathcal{D} = \{(s_i, a_i, r_i)\}_{i=1}^N$, and confidence level δ.
Set $N(s,a) = \sum_{i=1}^N \mathbb{1}\{(s_i, a_i) = (s,a)\}$ for all $a \in \mathcal{A}, s \in \mathcal{S}$.
for $s \in \mathcal{S}, a \in \mathcal{A}$ **do**
    if $N(s,a) = 0$ **then**
        Compute the empirical reward $\hat{r}(s,a) \leftarrow\!\shortmid 0$.
        Compute the penalty $b(s,a) = 1$.
    else
        Compute the empirical reward $\hat{r}(s,a) \leftarrow\!\shortmid \frac{1}{N(s,a)} \sum_{i=1}^N r_i \, \mathbb{1}\{(s_i, a_i) = (s,a)\}$.
        Compute the penalty $b(s,a) = \sqrt{\frac{2000 \log(2S|\mathcal{A}|/\delta)}{N(s,a)}}$.
Return: $\hat{\pi}(s) \in \arg\max_a \hat{r}(s,a) - b(s,a)$ for each $s \in \mathcal{S}$.

Pessimism is then applied through a penalty function $b(s,a)$ and for every state s the algorithm returns

$$\hat{\pi}(s) \in \arg\max_a \hat{r}(s,a) - b(s,a).$$

Algorithm 3 generalizes the LCB instance given in Algorithm 2 to the CB setting.

The following theorem establishes an upper bound on the expected sub-optimality of the policy returned by Algorithm 3; see Section 3.8.1 for a complete proof.

Theorem 3.4 (LCB sub-optimality, CB)**.** *Consider a contextual bandit with $S \geq 2$ and assume that*

$$\max_s \frac{\rho(s)}{\mu(s, \pi^\star(s))} \leq C^\star,$$

for some $C^\star \geq 1$. Setting $\delta = 1/N$, the policy $\hat{\pi}$ returned by Algorithm 3 obeys

$$\mathbb{E}_{\mathcal{D}}[J(\pi^\star) - J(\hat{\pi})] \lesssim \min\left(1, \widetilde{O}\left(\sqrt{\frac{S(C^\star - 1)}{N}} + \frac{S}{N}\right)\right).$$

It is interesting to note that the sub-optimality bound in Theorem 3.4 consists of two terms. The first term is the usual statistical estimation rate of $1/\sqrt{N}$. The second term is due to *missing mass*, which captures the suboptimality incurred in states for which an optimal arm is never observed in the batch dataset. More importantly, the dependency of the first term on data composition is $C^\star - 1$ instead of $C^\star$. When $C^\star$ is close to one, LCB enjoys a faster rate of $1/N$, reminiscent of the rates achieved by behavioral cloning in imitation learning, without the knowledge of $C^\star$ or the behavior policy. Furthermore, the convergence rate smoothly transitions from $1/N$ to $1/\sqrt{N}$ as $C^\star$ increases.

3.3.2 Optimality of LCB for solving offline contextual bandits

In this section, we establish an information-theoretic lower bound for the contextual bandit setup described above. Define the following family of contextual bandits problems

$$\mathsf{CB}(C^\star) := \{(\rho, \mu, R) \mid \max_s \frac{\rho(s)}{\mu(s, \pi^\star(s))} \le C^\star\}.$$

Note that the optimal policy $\pi^\star$ implicitly depends on the reward distribution R.

Let $\hat{\pi} : \mathcal{S} \mapsto \mathcal{A}$ be an arbitrary estimator of the best arm $\pi(s)$ for any state s, which is a measurable function of the data $\{(s_i, a_i, r_i)\}_{i=1}^N$. The worst-case risk of $\hat{\pi}$ is defined as

$$\sup_{(\rho,\mu,R)\in\mathsf{CB}(C^\star)} \mathbb{E}_{\mathcal{D}}[J(\pi^\star) - J(\hat{\pi})].$$

We have the following minimax lower bound for offline learning in contextual bandits with $S \ge 2$; see Section 3.8.2 for a proof. Note that the case of $S = 1$ is already addressed in Theorem 3.2.

Theorem 3.5 (Information-theoretic limit, CB)**.** *Assume that $S \ge 2$. For any $C^\star \ge 1$, one has*

$$\inf_{\hat{\pi}} \sup_{(\rho,\mu,R)\in\mathsf{CB}(C^\star)} \mathbb{E}_{\mathcal{D}}[J(\pi^\star) - J(\hat{\pi})] \gtrsim \min\left(1, \sqrt{\frac{S(C^\star - 1)}{N}} + \frac{S}{N}\right).$$

Comparing Theorem 3.5 with Theorem 3.4, one readily sees that the LCB approach enjoys a near-optimal rate in contextual bandits with $S \ge 2$, regardless of the data composition parameter $C^\star$. This is in stark contrast to the MAB case.

On a closer inspection, in the $C^\star \in [1, 2)$ regime, there is a clear separation between the information-theoretic difficulties of offline learning in MAB, which has an exponential rate in N, and CB with at least 2 states, which has a $1/N$ rate. The reason behind this separation is the possibility of missing mass when $S \ge 2$. Informally, when there is only one state, the probability that an optimal action is never observed in the dataset decays exponentially. On the other hand, when there are more than one states, the probability that an optimal action is never observed for at least one state decays with the rate of $1/N$.

Assume hypothetically that we are provided with the knowledge that $C^\star \in (1, 2)$. Recall that with such a knowledge, the most played arm achieves a faster rate in the MAB setting. Under this circumstance, one might wonder whether simply picking the most played arm in every state also achieves a fast rate in the CB setting. Strikingly, the answer is negative as the following proposition shows that the most played arm fails to achieve a vanishing rate when $C^\star \in (1, 2)$. The proof of this theorem is deferred to Section 3.8.3.

Proposition 3.3 (Failure of the most played arm, CB)**.** *For any $C^\star \in (1, 2)$, there exists a contextual bandit problem $(\rho, \mu, R) \in \mathsf{CB}(C^\star)$ such that for the policy $\hat{\pi}(s) = \arg\max_a N(s, a)$,*

$$\lim_{N\to\infty} \mathbb{E}_{\mathcal{D}}[J(\pi^\star) - J(\hat{\pi})] \ge C^\star - 1.$$

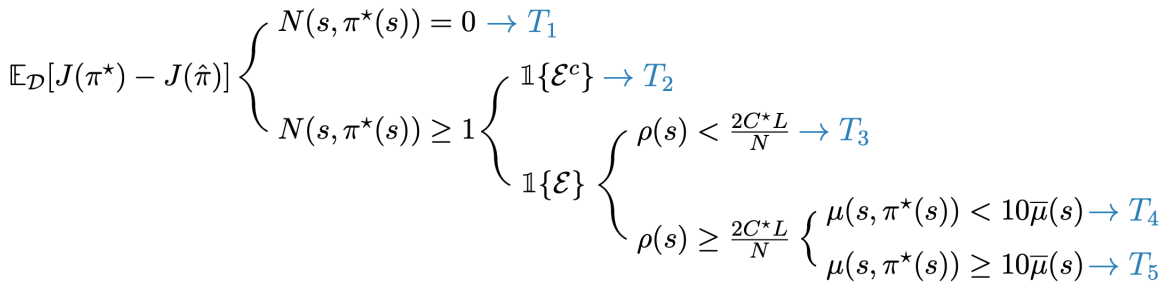

Figure 3.3: Decomposition of the sub-optimality of the policy $\hat{\pi}$ returned by Algorithm 3.

We briefly describe the intuition here. Under concentrability assumption, we can move at most $C^\star - 1$ mass from $d^\star$ to sub-optimal actions. Thus we can design a specific contextual bandit instance such that a $C^\star - 1$ fraction of the states pick wrong actions by choosing the most played arm instead. This shows that even when $C^\star \in (1, 2)$, the most played arm approach for CB does not have a decaying rate in N, whereas in the MAB case it converges exponentially fast.

3.3.3 Architecture of the proof

We pause to lay out the main steps to prove the upper bound in Theorem 3.4. It is worth pointing out that following the MAB sub-optimality analysis as detailed in Section 3.7.2 only yields a crude upper bound of $\sqrt{C^\star S/N} + S/N$ on the sub-optimality of $\hat{\pi}$. When $C^\star$ is close to one, i.e., when we have access to a nearly-expert dataset, such analysis only gives a $\sqrt{S/N}$ rate. This rate is clearly worse than the rate S/N achieved by the imitation learning algorithms. Therefore, special considerations are required for analyzing the sub-optimality of LCB in contextual bandits in order to establish the tight dependence of $\sqrt{(C^\star - 1)S/N} + S/N$ instead of $\sqrt{C^\star S/N}$.

We achieve this goal by directly analyzing the policy sub-optimality via a gradual decomposition of the sub-optimality of $\hat{\pi}$ as illustrated in Figure 3.3. The decomposition steps are described below.

First level of decomposition. In the first level of decomposition, we separate the error based on whether $N(s, \pi^\star(s))$ is zero for a certain state s. When $N(s, \pi^\star(s)) = 0$, there is absolutely no basis for the LCB approach to figure out the correct action $\pi^\star(s)$. Fortunately, this type of error, incurred by *missing mass*, can be bounded by

$$T_1 \lesssim \frac{C^\star S}{N}. \tag{3.14}$$

From now on, we focus on the case in which the expert action $\pi^\star(s)$ is seen for every state s.

Second level of decomposition. The second level of decomposition hinges on the following clean/good event:

$$\mathcal{E} \coloneqq \{\forall s, a : \ |r(s,a) - \hat{r}(s,a)| \leq b(s,a)\}. \tag{3.15}$$

In words, the event $\mathcal{E}$ captures the scenario in which the penalty function provides valid confidence bounds for every state-action pair. Standard concentration arguments tell us that $\mathcal{E}$ takes place with high probability, i.e., the term T_2 in the figure is no larger than δ. By setting δ small, say $1/N$, we are allowed to concentrate on the case when $\mathcal{E}$ holds.

Third level of decomposition. The third level of decomposition relies on the observation that states with small weights (i.e., $\rho(s)$ is small) have negligible effects on the sub-optimality $J(\pi^\star) - J(\hat{\pi})$. More specifically, the aggregated contribution T_3 from the states with $\rho(s) \lesssim \frac{C^\star L}{N}$ is upper bounded by

$$T_3 \lesssim \frac{C^\star S L}{N}. \tag{3.16}$$

This allows us to focus on the states with large weights. We record an immediate consequence of large $\rho(s)$ and the data coverage assumption, that is $\mu(s, \pi^\star(s)) \geq \rho(s)/C^\star \asymp L/N$.

Fourth level of decomposition. Now comes the most important part of the error decomposition, which is not present in the MAB analysis. We decompose the error based on whether the optimal action has a higher data probability $\mu(s, \pi^\star(s))$ than the total probability of sub-optimal actions $\overline{\mu}(s) \coloneqq \sum_{a \neq \pi^\star(s)} \mu(s,a)$. In particular, when $\mu(s, \pi^\star(s)) < 10\overline{\mu}(s)$, we can repeat the analysis of MAB and show that

$$T_4 \lesssim \sqrt{\frac{S(C^\star - 1)L}{N}}.$$

Here, the appearance of $C^\star - 1$, as opposed to $C^\star$ is due to the restriction $\mu(s, \pi^\star(s)) < 10\overline{\mu}(s)$. One can verify that $\mu(s, \pi^\star(s)) < 10\overline{\mu}(s)$ together with the data coverage assumption ensures that

$$\sum_{s: \rho(s) \geq 2C^\star L/N, \mu(s,\pi^\star(s)) < 10\overline{\mu}(s)} \rho(s) \lesssim C^\star - 1.$$

On the other hand, when $\mu(s, \pi^\star(s)) \geq 10\overline{\mu}(s)$, i.e., when the optimal action is more likely to be seen in the dataset, the penalty function $b(s, \pi^\star(s))$ associated with the optimal action would be much smaller than those of the sub-optimal actions. Thanks to the LCB approach, the optimal action will be chosen with high probability, i.e., $T_5 \lesssim 1/N^{10}$.

Putting the pieces together, we arrive at the desired rate $O(\sqrt{\frac{S(C^\star - 1)}{N}} + \frac{S}{N})$.

3.4 LCB in Markov decision processes

Now we are ready to instantiate the LCB principle to the full-fledged Markov decision process. We propose a variant of value iteration with LCB (VI-LCB) in Section 3.4.1 and present its performance guarantee in Section 3.4.2. Section 3.4.3 is devoted to the information-theoretic lower bound for offline learning in MDPs, which leaves us with a regime in which it is currently unclear whether LCB for MDP is optimal or not. However, we conjecture that VI-LCB is optimal for all ranges of $C^\star$. We conclude our discussion in Section 3.4.4 with an explanation about the technical difficulty of closing the gap and a preview to a simple episodic example where we manage to prove the optimality of LCB with a rather *intricate* analysis.

Additional notation. We present the algorithm and results in this section with the help of some matrix notation for MDPs. For a function $f : \mathcal{X} \mapsto \mathbb{R}$, we overload the notation and write $f \in \mathbb{R}^{|\mathcal{S}|}$ to denote a vector with elements $f(x)$, e.g., V, Q, and r. We write $P \in \mathbb{R}^{S|\mathcal{A}| \times S}$ to represent the probability transition matrix whose (s,a)-th row denoted by $P_{s,a}$ is a probability vector representing $P(\cdot \mid s,a)$. We use $P^\pi \in \mathbb{R}^{S|\mathcal{A}| \times S|\mathcal{A}|}$ to denote a transtion matrix induced by policy π whose $(s,a) \times (s',a')$ element is equal to $P(s'|s,a)\pi(a'|s')$. We write $\rho^\pi \in \mathbb{R}^{S|\mathcal{A}|}$ to denote the initial distribution induced by policy π whose (s,a) element is equal to $\rho(s)\pi(a|s)$.

3.4.1 Offline value iteration with LCB

Our algorithm design builds upon the classic value iteration algorithm. In essence, value iteration updates the value function $V \in \mathbb{R}^S$ using

$$\begin{aligned} Q(s,a) &\leftarrow\!\shortmid r(s,a) + \gamma P_{s,a} \cdot V, \qquad \text{for all } (s,a), \\ V(s) &\leftarrow\!\shortmid \max_a Q(s,a), \qquad \text{for all } s. \end{aligned}$$

Note, however, with offline data, we do not have access to the expected reward $r(s,a)$ and the true transition dynamics $P_{s,a}$. One can naturally replace them with the empirical counterparts $\hat{r}(s,a)$ and $\hat{P}_{s,a}$ estimated from offline data $\mathcal{D}$, and arrive at the empirical value iteration:

$$\begin{aligned} Q(s,a) &\leftarrow\!\shortmid \hat{r}(s,a) + \gamma \hat{P}_{s,a} \cdot V, \qquad \text{for all } (s,a), \\ V(s) &\leftarrow\!\shortmid \max_a Q(s,a), \qquad \text{for all } s. \end{aligned}$$

Mimicking the algorithmic design for MABs and CBs, we can subtract a penalty function $b(s,a)$ from the Q update as the finishing touch, which yields the value iteration algorithm with LCB:

$$Q(s,a) \leftarrow\!\shortmid \hat{r}(s,a) - b(s,a) + \gamma \hat{P}_{s,a} \cdot V, \qquad \text{for all } (s,a), \tag{3.17}$$

$$V(s) \leftarrow\!\shortmid \max_a Q(s,a), \qquad \text{for all } s. \tag{3.18}$$

Algorithm 4 Offline value iteration with LCB (VI-LCB)

Inputs: Batch dataset $\mathcal{D}$, discount factor γ, and confidence level δ.
Set $T \coloneqq \frac{\log N}{1-\gamma}$.
Randomly split $\mathcal{D}$ into $T+1$ sets $\mathcal{D}_t = \{(s_i, a_i, r_i, s_i')\}_{i=1}^m$ for $t \in \{0, 1, \ldots, T\}$ with $m \coloneqq N/(T+1)$.
Set $m_0(s,a) \coloneqq \sum_{i=1}^m \mathbb{1}\{(s_i, a_i) = (s,a)\}$ based on dataset $\mathcal{D}_0$.
For all $a \in \mathcal{A}$ and $s \in \mathcal{S}$, initialize $Q_0(s,a) = 0$, $V_0(s) = 0$ and set $\pi_0(s) = \arg\max_a m_0(s,a)$.
for $t = 1, \ldots, T$ **do**
    Initialize $r_t(s,a) = 0$ and set $P_{s,a}^t$ to be a random probability vector.
    Set $m_t(s,a) \coloneqq \sum_{i=1}^m \mathbb{1}\{(s_i, a_i) = (s,a)\}$ based on dataset $\mathcal{D}_t$.
    Compute penalty $b_t(s,a)$ for $L = 2000 \log(2(T+1)S|\mathcal{A}|/\delta)$

$$b_t(s,a) \coloneqq V_{\max} \cdot \sqrt{\frac{L}{m_t(s,a) \vee 1}}. \tag{3.19}$$

    for $(s,a) \in (\mathcal{S}, \mathcal{A})$ **do**
        if $m_t(s,a) \geq 1$ **then**
            Set $P_{s,a}^t$ to be empirical transitions and $r_t(s,a)$ be empirical average of rewards.
        Set $Q_t(s,a) \leftarrow\!\shortmid r_t(s,a) - b_t(s,a) + \gamma P_{s,a}^t \cdot V_{t-1}$.
    Compute $V_t^{\text{mid}} \leftarrow \max_a Q_t(s,a)$ and $\pi_t^{\text{mid}}(s) \in \arg\max_a Q_t(s,a)$.
    for $s \in \mathcal{S}$ **do**
        if $V_t^{\text{mid}}(s) \leq V_{t-1}(s)$ **then** $V_t(s) \leftarrow V_{t-1}(s)$ and $\pi_t(s) \leftarrow \pi_{t-1}(s)$.
        else $V_t(s) \leftarrow V_t^{\text{mid}}(s)$ and $\pi_t(s) \leftarrow \pi_t^{\text{mid}}(s)$.
Return $\hat{\pi} \coloneqq \pi_T$.

Algorithm 4 uses the update rule (3.17) as its key component as well as a few other tricks:

- Data splitting: Instead of using the full offline data $\mathcal{D} = \{(s_i, a_i, r_i, s_i')\}_{i=1}^N$ to form the empirical estimates $\hat{r}(s,a)$ and $\hat{P}_{s,a}$, Algorithm 4 deploys data splitting where each iteration (3.17) uses different samples to perform the update. This procedure is not needed in practice, however it is helpful in alleviating the dependency issues in the analysis, resulting in the removal of an extra factor of S in the sample complexity.

- Monotonic update: Unlike traditional value iteration methods, Algorithm 4 involves a monotonic improvement step, in which the value function V and the policy π are updated only when the corresponding value function is larger than that in the previous iteration. This extra step was first proposed in the work Sidford et al. (2018a) for reinforcement learning with access to a generative model. In a nutshell, the key benefit of the monotonic update is to shave a $1/(1-\gamma)$ factor in the sample complexity; we refer

the interested reader to the original work Sidford et al. (2018a) for further discussions on this step.

3.4.2 Performance guarantees of VI-LCB

Now we turn to the performance guarantee for the VI-LCB algorithm (cf. Algorithm 4).

Theorem 3.6 (LCB sub-optimality, MDP)**.** *Consider a Markov decision process and assume that*

$$\max_{s,a} \frac{d^\star(s,a)}{\mu(s,a)} \leq C^\star.$$

Then, for all $C^\star \geq 1$, policy $\hat{\pi}$ returned by Algorithm 4 with $\delta = 1/N$ achieves

$$\mathbb{E}_{\mathcal{D}}\left[J(\pi^\star) - J(\hat{\pi})\right] \lesssim \min\left(\frac{1}{1-\gamma}, \sqrt{\frac{SC^\star}{(1-\gamma)^5 N}}\right). \tag{3.20}$$

In addition, if $1 \leq C^\star \leq 1 + \frac{L\log(N)}{200(1-\gamma)N}$, we have a tighter performance upper bound

$$\mathbb{E}_{\mathcal{D}}\left[J(\pi^\star) - J(\hat{\pi})\right] \lesssim \min\left(\frac{1}{1-\gamma}, \frac{S}{(1-\gamma)^4 N}\right). \tag{3.21}$$

We will shortly provide a proof sketch of Theorem 3.6; a complete proof is deferred to Section 3.9.5. The upper bound shows that for all regime of $C^\star \geq 1$, we can guarantee a rate of $\widetilde{O}(\sqrt{SC^\star/((1-\gamma)^5 N)})$, which is similar to the rate of contextual bandit when the $C^\star = 1 + \Omega(1)$ by taking $\gamma = 0$. When $C = 1 + O(\log(N)/N)$, we can show a rate $S/((1-\gamma^4)N)$, which also recovers the result in contextual bandit case. However, in the regime of $C^\star \in [1 + \Omega(\log(N)/N), 1 + O(1)]$, contextual bandit gives $\sqrt{S(C^\star - 1)/N}$, while we fail to give the same dependence on $C^\star$ in this case. We defer the further discussion on the sub-optimality of this regime to Section 3.4.4.

Remark 3.1. *Relaxation of the concentrability assumption is possible by allowing the ratio to hold only for a subset $\mathcal{C}$ of state-action pairs and characterizing the sub-optimality incurred by $(s,a) \in \mathcal{C}$ via a missing mass analysis dependent on a constant ξ such that $\sum_{(s,a)\notin\mathcal{C}} d^\star(s,a) \leq \xi$.*

Proof sketch for Theorem 3.6. For the general case of $C^\star \geq 1$, we first define the clean event of interest as below.

$$\mathcal{E}_{\mathrm{MDP}} \coloneqq \left\{\forall s,a,t: \left|r(s,a) - r_t(s,a) + \gamma\left(P_{s,a} - P^t_{s,a}\right)\cdot V_{t-1}\right| \leq b_t(s,a)\right\}. \tag{3.22}$$

In words, on the event $\mathcal{E}_{\mathrm{MDP}}$, the penalty function $b_t(s,a)$ well captures the statistical fluctuations of the Q-function estimate $r_t(s,a) + \gamma P^t_{s,a}\cdot V_{t-1}$. The following lemma shows that this event happens with high probability. The proof is postponed to Section 3.9.2.

Lemma 3.1 (Clean event probability, MDP)**.** *One has* $\mathbb{P}(\mathcal{E}_{MDP}) \geq 1 - \delta$.

In the above lemma, concentration of V_t is only needed instead of any value function V such as required in the work Yu et al. (2020). For the latter to hold, one needs to introduce another factor of $\sqrt{S}$ by taking a union bound. We avoid a union bound by exploiting the independence of $P^t_{s,a}$ and V_t obtained by randomly splitting the dataset. This is key to obtaining an optimal dependency on the state size S.

Under the clean event, we can show that the monotonically increasing value function V_t always lower bounds the value of the corresponding policy π_t, along with a recursive inequality on the sub-optimality of Q_{t+1} w.r.t. $Q^\star$ to penalty and sub-optimality of the previous step.

Proposition 3.4 (Contraction properties of Algorithm 4)**.** *Let* π *be an arbitrary policy. On the event* $\mathcal{E}_{MDP}$*, one has for all* $s \in \mathcal{S}, a \in \mathcal{A}$*, and* $t \in \{1, \ldots, T\}$*:*

$$V_{t-1} \leq V_t \leq V^{\pi_t} \leq V^\star, \quad Q_t \leq r + \gamma P V_{t-1}, \quad \text{and} \quad Q^\pi - Q_t \leq \gamma P^\pi (Q^\pi - Q_{t-1}) + 2b_t.$$

By recursively applying the last inequality, we can derive a value difference lemma. The following lemma relates the sub-optimality to the penalty term b_t, of which we have good control:

Lemma 3.2 (Value difference for Algorithm 4)**.** *Let* π *be an arbitrary policy. On the event* $\mathcal{E}_{MDP}$*, one has for all* $t \in \{1, \ldots, T\}$

$$J(\pi) - J(\pi_t) \leq \frac{\gamma^t}{1-\gamma} + 2\sum_{i=1}^{t} \mathbb{E}_{\nu^\pi_{t-i}}[b_i(s,a)].$$

Here, $\nu^\pi_k := \gamma^k \rho^\pi (P^\pi)^k$ *for* $k \geq 0$.

The proof is provided in Section 3.9.4. The value difference bound has two terms: the first term is due to convergence error of value iteration and the second term is the error caused by subtracting penalties $b_i(s,a)$ in each iteration i from the rewards. By plugging in b_i and choosing t appropriately we can get the desired performance guarantee.

For the case of $1 \leq C^\star \leq 1 + \frac{L \log(N)}{200(1-\gamma)N}$, we adopt a similar decomposition as the contextual bandit analysis sketched in Section 3.3.3. The only difference is that since $C^\star$ is small enough, we know that all the sub-optimal actions have very small mass in the μ. Thus LCB enjoys a rate of $1/N$ as the imitation learning case.

3.4.3 Information-theoretic lower bound for offline RL in MDPs

In this section, we focus on the statistical limits of offline learning in MDPs.

Define the following family of MDPs

$$\mathsf{MDP}(C^\star) = \{(\rho, \mu, P, R) \mid \max_{s,a} \frac{d^\star(s,a)}{\mu(s,a)} \leq C^\star\}.$$

Note that here the normalized discounted occupancy measure $d^\star$ depends implicitly on the specification of the MDP, i.e., ρ, P, and R.

We have the following minimax lower bound for offline policy learning in MDPs, with the proof deferred to Section 3.9.6.

Theorem 3.7 (Information-theoretic limit, MDP)**.** *For any $C^\star \geq 1, \gamma \geq 0.5$, one has*

$$\inf_{\hat{\pi}} \sup_{(\rho,\mu,P,R)\in \mathsf{MDP}(C^\star)} \mathbb{E}_{\mathcal{D}}[J(\pi^\star) - J(\hat{\pi})] \gtrsim \min\left(\frac{1}{1-\gamma}, \frac{S}{(1-\gamma)^2 N} + \sqrt{\frac{S(C^\star - 1)}{(1-\gamma)^3 N}}\right).$$

Several remarks are in order.

Imitation learning and offline learning. It is interesting to note that similar to the lower bound for contextual bandits, the statistical limit involves two separate terms $\frac{S}{(1-\gamma)^2 N}$ and $\sqrt{\frac{S(C^\star-1)}{(1-\gamma)^3 N}}$. The first term captures the imitation learning regime under which a fast rate $1/N$ is expected, while the second term deals with the large $C^\star$ regime with a parametric rate $1/\sqrt{N}$. More interestingly, the dependence on $C^\star$ appears to be $C^\star - 1$, which is different from the performance upper bound of VI-LCB in Theorem 3.6. We will comment more on this in the coming section.

Dependence on the effective horizon $1/(1-\gamma)$. Comparing the upper bound in Theorem 3.6 with the lower bound in Theorem 3.7, one sees that the sample complexity of VI-LCB for all regimes of C^π is loose by an extra $1/(1-\gamma)^2$ factor in sample complexity. The horizon dependency has been addressed by modifications to the VI-LCB algorithm in follow-up works by Xie et al. (2021) and Kumar et al. (2022). The first work uses Bernstein-based penalty and variance reduction similar to the technique used in the work Sidford et al. (2018a). The second work shows that the vanilla version of VI-LCB without data splitting improves horizon dependency using an s-absorbing MDP construction similar to the work (Agarwal et al., 2020c).

3.4.4 What happens when $C^\star \in [1 + \Omega(1/N), 1 + O(1)]$?

Now we return to the discussion on the dependency on $C^\star$. Ignore the dependency on $1/(1-\gamma)$ for the moment. By comparing Theorems 3.6 and 3.7, one realizes that VI-LCB is optimal both when $C^\star \geq 1 + \Theta(1)$ and when $C^\star \leq 1 + \Theta(1/N)$. However, in the middling regime when $C^\star \in [1 + \Omega(1/N), 1 + O(1)]$, the upper and lower bounds differ in their dependency on $C^\star$. More specifically, the upper bound presented in Theorem 3.6 is $\sqrt{SC^\star/N}$, while the lower bound in Theorem 3.7 is $S/N + \sqrt{S(C^\star - 1)/N}$.

Technical hurdle. We *conjecture* that VI-LCB is optimal even this regime and the current gap is an artifact of our analysis. However, we would like to point out that, although we manage to close the gap in contextual bandits, the case with MDPs is significantly more challenging due to error propagation. Naively applying the decomposition in the contextual bandit case fails to achieve the $C^\star - 1$ dependence in this regime. Take the term T_5 in Figure 3.3 as an example. For contextual bandits, given the selection rule is

$$\hat{\pi}(s) \leftarrow\!\shortmid \arg\max_a \hat{r}(s,a) - \sqrt{\frac{L}{N(s,a)}}, \tag{3.23}$$

it is straightforward to check that as long as the optimal action is taken with much higher probability than the sub-optimal ones, i.e., $\mu(s,\pi^\star(s)) \gg \sum_{a\neq\pi^\star(s)} \mu(s,a)$, the LCB approach will pick the right action regardless of the value gap $r(s,\pi^\star(s)) - r(s,a)$. In contrast, due to the recursive update $Q(s,a) \leftarrow\!\shortmid r_t(s,a) - \sqrt{\frac{L}{N_t(s,a)}} + \gamma P^t_{s,a}\cdot V_{t-1}$, LCB picks the right action if

$$r_t(s,\pi^\star(s)) - \sqrt{\frac{L}{N_t(s,\pi^\star(s))}} + \gamma P^t_{s,\pi^\star(s)}\cdot V_{t-1} > r_t(s,a) - \sqrt{\frac{L}{N_t(s,a)}} + \gamma P^t_{s,a}\cdot V_{t-1},$$

for all $a \neq \pi^\star(s)$. The presence of the value estimate from the previous step, i.e., V_{t-1} (which is absent in CBs) drastically changes the picture: even if we know that $\mu(s,\pi^\star(s)) \gg \sum_{a\neq\pi^\star(s)} \mu(s,a)$ and hence $N_t(s,\pi^\star(s)) \gg N_t(s,a)$, the current analysis does not guarantee the above inequality to hold. It is likely that for the value gap $Q^\star(s,\pi^\star(s)) - Q^\star(s,a)$ to affect whether the LCB algorithm chooses the optimal action. How to study the interplay between the value gap and the policy chosen by LCB forms the main obstacle to obtaining tight performance guarantees when $C^\star \in [1+\Omega(1/N), 1+O(1)]$.

A confirmation from an episodic MDP. In Section 3.10.6 we present an episodic example with the intention to demonstrate that (1) a variant of VI-LCB in the episodic case is able to achieve the optimal dependency on $C^\star$ and hence closing the gap between the upper and the lower bounds, and (2) the tight analysis of the sub-optimality is rather intricate and depends on a delicate decomposition based on the value gap $Q^\star(s,\pi^\star(s)) - Q^\star(s,a)$.

As a preview, we illustrate the episodic MDP with $H = 3$ in Figure 3.4. It turns out that when tackling the term similar to T_5 in Figure 3.3, a further decomposition based on the value gap is needed. In a nutshell, we decompose the error into two cases: (1) when $Q^\star(s,1) - Q^\star(s,2)$ is large, and (2) when $Q^\star(s,1) - Q^\star(s,2)$ is small. Intuitively, in the latter case, the contribution to the sub-optimality is well controlled, and in the former one, we manage to show that VI-LCB selects the right action with high probability. What is more interesting and suprising is that the right threshold for value gap is given by $\sqrt{(C^\star-1)/N}$. Ultimately, this allows us to achieve the optimal dependency on $C^\star$.

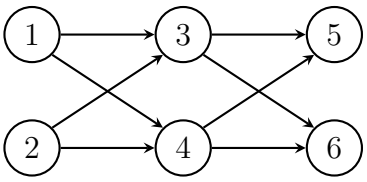

Figure 3.4: An episodic MDP with $H = 3$, two states per level, and two actions $\mathcal{A} = \{1, 2\}$ available from every state. The rewards are assumed to be deterministic and bounded. Action 1 is assumed to be optimal in all states and that $\mu(s, 1) \geq 9\mu(s, 2)$.

3.5 Related work

In this section we review additional related works. In Section 3.5.1, we discuss various assumptions on the batch dataset that have been proposed in the literature. In Section 3.5.2, we review conservative methods in offline RL. We conclude this section by comparing existing lower bounds with the ones presented in this paper.

3.5.1 Assumptions on batch dataset

One of the main challenges in offline RL is the insufficient coverage of the dataset caused by lack of online exploration (Wang et al., 2020a; Zanette, 2020; Szepesvári, 2010) and in particular the *distribution shift* in which the occupancy density of the behavior policy and the one induced by the learned policy are different. This effect can be characterized using concentrability coefficients (Munos, 2007) which impose bounds on the density ratio (importance weights).

Most concentrability requirements imposed in existing offline RL involve taking a supremum of the density ratio over all state-action pairs and all policies, i.e., $\max_\pi C^\pi$ (Scherrer, 2014; Chen and Jiang, 2019; Jiang, 2019; Wang et al., 2019a; Liao et al., 2020; Liu et al., 2019; Zhang et al., 2020a) and some definitions are more complex and stronger assuming a bounded ratio per time step (Szepesvári and Munos, 2005; Munos, 2007; Antos et al., 2008; Farahmand et al., 2010; Antos et al., 2007). A more stringent definition originally proposed by Munos (2003) also imposes exploratoriness on state marginals. This definition is recently used by Xie and Jiang (2020) to develop an efficient offline RL algorithm with general function approximation and only realizability. The MABO algorithm proposed by Xie and Jiang (2020) and the related algorithms by Feng et al. (2019) and Uehara et al. (2020) use a milder definition based on a *weighted* norm of density ratios as opposed to the infinity norm. In contrast, to compete with an optimal policy, we only require coverage over states and actions visited by that policy, which is referred to as the "best" concentrability coefficient (Scherrer, 2014; Geist et al., 2017; Agarwal et al., 2020d; Xie and Jiang, 2020).

Another related assumption is the uniformly lower bounded data distribution. For example, some works consider access to a generative model with an equal number of samples on all state-action pairs (Sidford et al., 2018a,b; Agarwal et al., 2020c; Li et al., 2020). As

discussed before, this assumption is significantly stronger than assuming $C^\star$ is bounded. Furthermore, one can modify the analysis of the LCB algorithm to show optimal data composition dependency in this case as well.

3.5.2 Conservatism in offline RL

In practice, such high coverage assumptions on batch dataset also known as data diversity (Levine et al., 2020) often fail to hold (Gulcehre et al., 2020; Agarwal et al., 2020e; Fu et al., 2020). Several methods have recently emerged to address such strong data requirements. The first category involves policy regularizers or constraints to ensure closeness between the learned policy and the behavior policy (Fujimoto et al., 2019b; Wu et al., 2019; Jaques et al., 2019; Peng et al., 2019; Siegel et al., 2020; Wang et al., 2020c; Kumar et al., 2019; Fujimoto et al., 2019a; Ghasemipour et al., 2020; Nachum et al., 2019b; Zhang et al., 2020b; Nachum et al., 2019a; Zhang et al., 2020c). These methods are most suited when the batch dataset is nearly-expert (Wu et al., 2019; Fu et al., 2020) and sometimes require the knowledge of the behavior policy.

Another category includes the value-based methods. Kumar et al. (2020) propose conservative Q-learning through value regularization and demonstrate empirical success. Liu et al. (2020) propose a variant of fitted Q-iteration with a conservative update called MSB-QI. This algorithm effectively requires the data distribution to be uniformly lower bounded on the state-action pairs visited by any competing policy. Moreover, the sub-optimality of MSB-QI has a $1/(1-\gamma)^4$ horizon dependency compared to ours which is $1/(1-\gamma)^{2.5}$.

The last category involves learning pessimistic models such as Kidambi et al. (2020), Yu et al. (2020) and Yu et al. (2021) all of which demonstrate empirical success. From a theoretical perspective, the recent work Jin et al. (2020c) studies pessimism in offline RL in episodic MDPs and function approximation setting. The authors present upper and lower bounds for linear MDPs with a suboptimality gap of dH, where d is the feature dimension and H is the horizon. Specialized to the tabular case, this gap is equal to SAH, compared to ours which is only H. Furthermore, this work does not study the adaptivity of pessimism to data composition.

Another recent work by Yin et al. (2021) studies pessimism in tabular MDP setting and proves matching upper and lower bounds. However, their approach requires a uniform lower bound on the data distribution that traces an optimal policy. This assumption is stronger than ours; for example, it requires optimal actions to be included in the states not visited by an optimal policy. Furthermore, this characterization of data coverage does not recover the imitation learning setting: if the behavior policy is exactly equal to the optimal policy, data distribution lower bound can still be small.

3.5.3 Information-theoretic lower bounds

There exists a large body of literature providing information-theoretic lower bounds for RL under different settings; see e.g., Dann and Brunskill (2015); Krishnamurthy et al. (2016);

Jiang et al. (2017); Jin et al. (2018); Azar et al. (2013); Ma et al. (2021); Lattimore and Hutter (2012); Domingues et al. (2020); Duan et al. (2020); Zanette (2020); Wang et al. (2020a). In the generative model setting with uniform samples, Azar et al. (2013) proves a lower bound on value sub-optimality which is later extended to policy sub-optimality by Sidford et al. (2018a). For the offline RL setting, Kidambi et al. (2020) prove a lower bound only considering the data and policy occupancy support mismatch without dependency on sample size. Jin et al. (2020c) gives a lower bound for linear MDP setting but which does not give a tight dependency on parameters when specialized to the tabular setting. In Yin et al. (2020, 2021), a hard MDP is constructed with a dependency on the data distribution lower bound. In contrast, our lower bounds depend on $C^\star$, which has not been studied in the past, and holds for the entire data spectrum. In the imitation learning setting, (Xu et al., 2020) considers discounted MDP setting and shows a lower bound on the performance of the behavior cloning algorithm. We instead present an information-theoretic lower bound for any algorithm for $C^\star = 1$ which is based on adapting the construction of Rajaraman et al. (2020) to the discounted case.

3.6 Discussion

In this paper, we propose a new batch RL framework based on the single policy concentrability coefficient (e.g., $C^\star$) that smoothly interpolates the two extremes of data composition encountered in practice, namely the expert data and uniform coverage data. Under this new framework, we pursue the statistically optimal algorithms that can even be implemented without the knowledge of the exact data composition. More specifically, focusing on the lower confidence bound (LCB) approach inspired by the principle of pessimism, we find that LCB is adaptively minimax optimal for addressing the offline contextual bandit problems and the optimal rate naturally bridges the $1/N$ rate when data is close to following the expert policy and the $1/\sqrt{N}$ rate in the typical offline RL case. Here N denotes the number of samples in the batch dataset. We also investigate the LCB approach in the offline multi-armed bandit problems and Markov decision processes. The message is somewhat mixed. For bandits, LCB is shown to be optimal for a wide range of data compositions, however, LCB without the knowledge of data composition, is provably non-adaptive in the near-expert data regime. When it comes to MDPs, we show that LCB is adaptively rate-optimal when $C^\star$ is extremely close to 1, and when $C^\star \geq 1+$ constant. Contrary to bandits, we conjecture that LCB is optimal across the spectrum of data composition, which is left for future work.

3.7 Proofs for multi-armed bandits

In Section 3.7.1, we prove Proposition 3.1 that demonstrates the failure of the best empirical arm when solving offline MABs. Section 3.7.2 is devoted to the proof of Theorem 3.1, which supplies the performance upper bound of the LCB approach. This upper bound is

accompanied by a minimax lower bound given in Section 3.7.3. In the end, we provably show the lack of adaptivity of the LCB approach in Section 3.7.4.

3.7.1 Proof of Proposition 3.1

We start by introducing the bandit instance under consideration. Set $|\mathcal{A}|= 2$, $a^\star = 1$, $\mu(1) = (N-1)/N$, and $\mu(2) = 1/N$. As for the reward distributions, for the optimal arm $a^\star = 1$, we let $R(1) = 2\epsilon$ almost surely. In contrast, for arm 2 we set

$$R(2) = \begin{cases} 2.1\epsilon, & \text{w.p. } 0.5, \\ 0, & \text{w.p. } 0.5. \end{cases}$$

It is easy to check that indeed $a^\star = 1$ is the optimal arm to choose. Our goal is to show that for this particular bandit problem, given N offline data from μ and R, the empirical best arm $\hat{a}$ will perform poorly with high probability.

To see this, consider the following event

$$\mathcal{E}_1 := \{N(2) = 1\}.$$

We have

$$\mathbb{P}(\mathcal{E}_1) = N \cdot \mu(1)^{N-1} \cdot \mu(2) = (1 - 1/N)^{N-1}.$$

As long as N is sufficiently large (say $N \geq 500$), we have $\mathbb{P}(\mathcal{E}_1) \geq 0.36$ for any $0 \leq n \leq N$, and thus $\mathbb{P}(\mathcal{E}_1) \geq 0.36$.

Now we are in position to develop a performance lower bound for the empirical best arm $\hat{a}$. By construction, we have $r(1) - r(2) = 0.95\epsilon$. Therefore the sub-optimality is given by

$$\begin{aligned} \mathbb{E}_{\mathcal{D}}[r(a^\star) - r(\hat{a})] &= 0.95\epsilon \cdot \mathbb{P}(\hat{a} \neq a^\star) \\ &\geq 0.95\epsilon \cdot \mathbb{P}(\mathcal{E}_1 \cap \hat{r}(2) = 2.1\epsilon) \\ &\geq 0.95\epsilon \cdot 0.18 > 0.1\epsilon. \end{aligned}$$

Rescaling the value of ϵ finishes the proof.

3.7.2 Proof of Theorem 3.1

Before embarking on the main proof, we record two useful lemmas. The first lemma sandwiches the true mean reward by the empirical one and the penalty function, which directly follows from Hoeffding's inequality and a union bound. For completeness, we provide the proof at the end of this subsection.

Lemma 3.3. *With probability at least $1 - \delta$, we have*

$$\hat{r}(a) - b(a) \leq r(a) \leq \hat{r}(a) + b(a), \quad \textit{for all } 1 \leq a \leq |\mathcal{A}|. \tag{3.24}$$

The second one is a simple consequence of the Chernoff bound for binomial random variables.

Lemma 3.4. *With probability at least* $1 - \exp(-N\mu(a^\star)/8)$*, one has*

$$N(a^\star) \geq \frac{1}{2} N\mu(a^\star). \tag{3.25}$$

Denote by $\mathcal{E}$ the event that both relations (3.24) and (3.25) hold. Conditioned on $\mathcal{E}$, one has

$$r(a^\star) \leq \hat{r}(a^\star) + b(a^\star) = \hat{r}(a^\star) - b(a^\star) + 2b(a^\star).$$

In view of the definition of $\hat{a}$, we have $\hat{r}(a^\star) - b(a^\star) \leq \hat{r}(\hat{a}) - b(\hat{a})$, and hence

$$r(a^\star) \leq \hat{r}(\hat{a}) - b(\hat{a}) + 2b(a^\star) \leq r(\hat{a}) + 2b(a^\star),$$

where the last inequality holds under the event $\mathcal{E}$ (in particular the bound (3.24) on $\hat{a}$). Now we are left with the term $b(a^\star)$. It suffices to lower bound $N(a^\star)$. Note that the event $\mathcal{E}$ (cf. the lower bound (3.25)) ensures that

$$N(a^\star) \geq \frac{1}{2} N\mu(a^\star) \geq \frac{N}{2C^\star} > 0.$$

As a result, we conclude

$$b(a^\star) = \sqrt{\frac{\log(2|\mathcal{A}|/\delta)}{2N(a^\star)}} \leq \sqrt{\frac{\log(2|\mathcal{A}|/\delta)}{N\mu(a^\star)}},$$

which further implies

$$r(a^\star) \leq r(\hat{a}) + 2\sqrt{\frac{\log(2|\mathcal{A}|/\delta)}{N\mu(a^\star)}} \tag{3.26}$$

whenever the event $\mathcal{E}$ holds. It is easy to check that under the assumption $N \geq 8C^\star \log(1/\delta)$, we have $\mathbb{P}(\mathcal{E}) \geq 1 - 2\delta$. This finishes the proof of the high probability claim.

In the end, we can compute the expected sub-optimality as

$$\begin{aligned}\mathbb{E}_{\mathcal{D}}[r(a^\star) - r(\hat{a})] &= \mathbb{E}_{\mathcal{D}}[(r(a^\star) - r(\hat{a}))\, 1\{\mathcal{E}\}] + \mathbb{E}_{\mathcal{D}}[(r(a^\star) - r(\hat{a}))\, 1\{\mathcal{E}^c\}] \\ &\leq 2\sqrt{\frac{\log(2|\mathcal{A}|/\delta)}{N\mu(a^\star)}}\mathbb{P}(\mathcal{E}) + \mathbb{P}(\mathcal{E}^c).\end{aligned}$$

Here the inequality uses the bound (3.26) and the fact that $r(a^\star) - r(\hat{a}) \leq 1$. We continue bounding the sub-optimality by

$$\mathbb{E}_{\mathcal{D}}[r(a^\star) - r(\hat{a})] \leq 2\sqrt{\frac{\log(2|\mathcal{A}|/\delta)}{N\mu(a^\star)}} + 2\delta \leq 2\sqrt{\frac{C^\star \log(2|\mathcal{A}|/\delta)}{N}} + 2\delta.$$

Here the last relation uses $\mu(a^\star) \geq 1/C^\star$. Taking $\delta = 1/N$ completes the proof.

Proof of Lemma 3.3. Consider a fixed action a. If $N(a) = 0$, one trivially has $\hat{r}(a) - b(a) = -1 \leq r(a) \leq \hat{r}(a) + b(a) = 1$. When $N(a) > 0$, applying Hoeffding's inequality, one sees that

$$\mathbb{P}\left(|\hat{r}(a) - r(a)| \geq \sqrt{\frac{\log(2|\mathcal{A}|/\delta)}{2N(a)}} \mid N(a)\right) \leq \frac{\delta}{|\mathcal{A}|}.$$

Since this claim holds for all possible $N(a)$, we have for any fixed action a

$$\mathbb{P}\left(|\hat{r}(a) - r(a)| \geq b(a)\right) \leq \frac{\delta}{|\mathcal{A}|}.$$

A further union bound over the action space yields the advertised claim. □

3.7.3 Proof of Theorem 3.2

We separate the proof into two cases: $C^\star \geq 2$ and $C^\star \in (1, 2)$. For both cases, our lower bound proof relies on the classic Le Cam's two-point method (Yu, 1997; Le Cam, 2012). In essence, we construct two MAB instances in the family $\mathsf{MAB}(C^\star)$ with different optimal rewards that are difficult to distinguish given the offline dataset.

The case of $C^\star \geq 2$. We consider a simple two-armed bandit. For the behavior policy, we set $\mu(2) = 1/C^\star$ and $\mu(1) = 1 - 1/C^\star$. Since we are constructing lower bound instances, it suffices to consider Bernoulli distributions supported on $\{0, 1\}$. In particular, we consider the following two possible sets for the Bernoulli means

$$f_1 = (\frac{1}{2}, \frac{1}{2} - \delta); \qquad f_2 = (\frac{1}{2}, \frac{1}{2} + \delta),$$

with $\delta \in [0, 1/4]$. Indeed, $(\mu, f_1), (\mu, f_2) \in \mathrm{MAB}(C^\star)$ with the proviso that $C^\star \geq 2$. Denote the loss/sub-optimality of an estimator $\hat{a}$ to be

$$\mathcal{L}(\hat{a}; f) \coloneqq r(a^\star) - r(\hat{a}), \tag{3.27}$$

where the optimal action $a^\star$ implicitly depends on the reward distribution f. Clearly, for any estimator $\hat{a}$, we have

$$\mathcal{L}(\hat{a}; f_1) + \mathcal{L}(\hat{a}; f_2) \geq \delta.$$

Therefore Le Cam's method tells us that

$$\inf_{\hat{a}} \sup_{(\mu,R)\in\mathsf{MAB}(C^\star)} \mathbb{E}_{\mathcal{D}}[r(a^\star) - r(\hat{a})] \geq \inf_{\hat{a}} \sup_{f\in f_1,f_2} \mathbb{E}_{\mathcal{D}}[\mathcal{L}(\hat{a}; f)] \geq \frac{\delta}{4} \cdot \exp(-\mathsf{KL}(\mathbb{P}_{\mu\otimes f_1} \| \mathbb{P}_{\mu\otimes f_2})).$$

Here $\mathsf{KL}(\mathbb{P}_{\mu\otimes f_1}\|\mathbb{P}_{\mu\otimes f_2})$ denotes the KL divergence between the two MAB instances with N samples. Direct calculations yield

$$\mathsf{KL}(\mathbb{P}_{\mu\otimes f_1}\|\mathbb{P}_{\mu\otimes f_2}) \le \frac{N\mathsf{KL}(\mathbb{P}_{f_1}\|\mathbb{P}_{f_2})}{C^\star} \le \frac{N(2\delta)^2}{C^\star(1/4-\delta^2)} \le 200N\delta^2/C^\star.$$

Here we use the fact that for two Bernoulli distribution, $\mathsf{KL}(\mathrm{Bern}(p)\|\mathrm{Bern}(q)) \le (p-q)^2/[q(1-q)]$ and that $\delta \in [0, 1/4]$. Taking

$$\delta = \min\left\{\frac{1}{4}, \sqrt{\frac{C^\star}{N}}\right\}$$

yields the desired lower bound for $C^\star \ge 2$.

The case of $C^\star \in (1,2)$. Recall that when $C^\star \ge 2$, we construct the same behavior distribution μ for two different reward distributions f_1, f_2. In contrast, in the case of $C^\star \in [1,2)$, we construct instances that are different in both the reward distributions as well as the behavior distribution. More specifically, let $\mu_1(1) = 1/C^\star$, $\mu_1(2) = 1-1/C^\star$, $f_1 = (\frac{1}{2}+\delta, \frac{1}{2})$ for some $\delta > 0$ which will be specified later. Similarly, we let $\mu_2(1) = 1-1/C^\star$, $\mu_2(2) = 1/C^\star$, $f_2 = (\frac{1}{2}, \frac{1}{2}+\delta)$. It is straightforward to check that $(\mu_1, f_1), (\mu_2, f_2) \in \mathsf{MAB}(C^\star)$. Clearly, for any estimator $\hat{a}$, we have

$$\mathcal{L}(\hat{a}; f_1) + \mathcal{L}(\hat{a}; f_2) \ge \delta.$$

Again, applying Le Cam's method, we have

$$\inf_{\hat{a}} \sup_{(\mu,R)\in\mathsf{MAB}(C^\star)} \mathbb{E}_{\mathcal{D}}[r(a^\star) - r(\hat{a})] \ge \frac{\delta}{4}\cdot\exp(-\mathsf{KL}(\mathbb{P}_{\mu_1\otimes f_1}\|\mathbb{P}_{\mu_2\otimes f_2})). \tag{3.28}$$

Note that

$$\begin{aligned}\mathsf{KL}(\mathbb{P}_{\mu_1\otimes f_1}\|\mathbb{P}_{\mu_2\otimes f_2}) &\le N\cdot\Big(\frac{\frac{1}{2}+\delta}{C^\star}\log(\frac{1+2\delta}{C^\star-1}) + \frac{\frac{1}{2}-\delta}{C^\star}\log(\frac{1-2\delta}{C^\star-1}) \\ &\quad + \frac{1-\frac{1}{C^\star}}{2}\log(\frac{C^\star-1}{1+2\delta}) + \frac{1-\frac{1}{C^\star}}{2}\log(\frac{C^\star-1}{1-2\delta})\Big) \\ &= N\cdot\left(\left(\frac{1+\delta}{C^\star}-\frac{1}{2}\right)\log\left(\frac{1+2\delta}{C^\star-1}\right) + \left(\frac{1-\delta}{C^\star}-\frac{1}{2}\right)\log\left(\frac{1-2\delta}{C^\star-1}\right)\right).\end{aligned}$$

Taking $\delta = \frac{2-C^\star}{2}$, we get $\mathsf{KL}(\mathbb{P}_{\mu_1\otimes f_1}\|\mathbb{P}_{\mu_2\otimes f_2}) \le N\cdot\frac{2-C^\star}{C^\star}\cdot\log\left(\frac{2}{C^\star-1}\right)$. Thus we know that

$$\inf_{\hat{a}} \sup_{(\mu,R)\in\mathsf{MAB}(C^\star)} \mathbb{E}_{\mathcal{D}}[r(a^\star) - r(\hat{a})] \gtrsim \exp\left(-(2-C^\star)\cdot\log\left(\frac{2}{C^\star-1}\right)\cdot N\right). \tag{3.29}$$

This finishes the proof of the lower bound for $C^\star \in (1,2)$.

3.7.4 Proof of Proposition 3.2

To begin with, we have $\mathbb{E}[r(a^\star) - r(\hat{a})] \leq \mathbb{P}(\hat{a} \neq a^\star)$, where we have used the fact that the rewards are bounded between 0 and 1. Thus it is sufficient to control $\mathbb{P}(\hat{a} \neq a^\star)$, which obeys

$$\mathbb{P}(\hat{a} \neq a^\star) = \mathbb{P}(\exists a \neq a^\star, N(a) \geq N(a^\star)) \leq \mathbb{P}(N - N(a^\star) \geq N(a^\star)) = \mathbb{P}(N(a^\star) \leq \frac{N}{2}).$$

Applying the Chernoff bound for binomial random variables yields

$$\mathbb{P}(N(a^\star) \leq \frac{N}{2}) \leq \exp\left(-N \cdot \mathsf{KL}\left(\mathrm{Bern}\left(\frac{1}{2}\right) \,\middle\|\, \mathrm{Bern}\left(\frac{1}{C^\star}\right)\right)\right).$$

Taking the previous steps collectively to arrive at the desired conclusion.

3.7.5 Proof of Theorem 3.3

We prove the case when $C^\star = 1.5$ and when $C^\star = 6$ separately.

The case when $C^\star = 1.5$. We begin by introducing the MAB problem.

The bandit instance. Consider a two-armed bandit problem with the optimal arm denoted by $a^\star$ and the sub-optimal arm a. We set $\mu(a^\star) = 1/C^\star$, and $\mu(a) = 1 - 1/C^\star$ in accordance with the requirement $1/\mu(a^\star) \leq C^\star$. We consider the following reward distributions: the optimal arm $a^\star$ has a deterministic reward equal to $1/2$ whereas the sub-optimal arm has a reward distribution of $\mathsf{Bern}(1/2 - g)$ for some $g \in (0, 1/3)$, which will be specified momentarily. It is straightforward to check that the arm $a^\star$ is indeed optimal and the MAB problem (μ, R) belongs to $\mathsf{MAB}(C^\star)$.

Lower bounding the performance of LCB. For the two-armed bandit problem introduced above, we have

$$\begin{aligned}
\mathbb{E}_{\mathcal{D}}[r(a^\star) - r(\hat{a})] &= g \cdot \mathbb{P}(\text{LCB chooses arm } a) \\
&= g \sum_{k=0}^{N} \mathbb{P}(\text{LCB chooses arm } a \mid N(a) = k)\mathbb{P}(N(a) = k) \\
&\geq g \sum_{k=N\mu(a)/2}^{2N\mu(a)} \mathbb{P}(\text{LCB chooses arm } a \mid N(a) = k)\mathbb{P}(N(a) = k), \qquad (3.30)
\end{aligned}$$

where we restrict ourselves to the event

$$\mathcal{E} := \{\frac{1}{2}N\mu(a) \leq N(a) \leq 2N\mu(a)\}.$$

It turns out that when $1 \le k \le 2N\mu(a)$, one has

$$\mathbb{P}(\text{LCB chooses arm } a \mid N(a) = k) \ge \frac{1}{\sqrt{4N\mu(a)}} \cdot \exp\left(-\frac{(g\sqrt{2N\mu(a)} + \sqrt{L})^2}{\frac{1}{4} - g^2}\right). \quad (3.31)$$

Combine inequalities (3.30) and (3.31) to obtain

$$\mathbb{E}_{\mathcal{D}}[r(a^\star) - r(\hat{a})] \ge g\frac{1}{\sqrt{4N\mu(a)}} \cdot \exp\left(-\frac{(g\sqrt{2N\mu(a)} + \sqrt{L})^2}{\frac{1}{4} - g^2}\right) \mathbb{P}(\mathcal{E}).$$

Setting $g = \min\{1/3, \sqrt{L/(2N\mu(a))}\}$ yields

$$\begin{aligned}\mathbb{E}_{\mathcal{D}}[r(a^\star) - r(\hat{a})] &\ge \frac{\min\left(\sqrt{L/(2N\mu(a))}, \frac{1}{3}\right)}{\sqrt{4N\mu(a)}} \cdot \exp\left(-32L\right) \mathbb{P}(\mathcal{E}) \\ &\ge \min\left(\frac{\sqrt{L}}{8N\mu(a)}, \frac{1}{12\sqrt{N\mu(a)}}\right) \cdot \exp\left(-32L\right),\end{aligned}$$

where the last inequality uses Chernoff's bound, i.e., $\mathbb{P}(\mathcal{E}) \ge 1 - 2\exp(-N\mu(a)/8) \ge \frac{1}{2}$. Substituting the definition of L and $\mu(a)$ completes the proof.

Proof of the lower bound (3.31). By the definition of LCB, we have

$$\begin{aligned}\mathbb{P}(\text{LCB chooses arm } a \mid N(a) = k) &= \mathbb{P}\left(1/2 - \sqrt{L/N(a^\star)} \le \hat{r}(a) - \sqrt{L/N(a)} \mid N(a) = k\right) \\ &\ge \mathbb{P}\left(\hat{r}(a) \ge 1/2 + \sqrt{L/N(a)} \mid N(a) = k\right) \\ &\ge \frac{1}{\sqrt{2k}} \cdot \exp\left(-k \cdot \mathsf{KL}\left(\frac{1}{2} - \sqrt{\frac{L}{k}} \middle\| \frac{1}{2} + g\right)\right) \\ &\ge \frac{1}{\sqrt{2k}} \cdot \exp\left(-\frac{k(g + \sqrt{\frac{L}{k}})^2}{\frac{1}{4} - g^2}\right).\end{aligned}$$

Here, the penultimate inequality comes from a lower bound for Binomial tails (Robert, 1990) and the last inequality uses the elementary fact that $\mathsf{KL}(p\|q) \le (p-q)^2/q(1-q)$. One can easily see that the probability lower bound is decreasing in k and hence when $N(a) = k \le 2N\mu(a)$, we have

$$\mathbb{P}(\text{LCB chooses the arm } a \mid N(a) = k) \ge \frac{1}{\sqrt{4N\mu(a)}} \cdot \exp\left(-\frac{(g\sqrt{2N\mu(a)} + \sqrt{L})^2}{\frac{1}{4} - g^2}\right).$$

This completes the proof. □

The case when $C^\star = 6$. We now prove the lower bound for the case of $C^\star = 6$.

The bandit instance. Consider a two-armed bandit problem with $\mu(a^\star) = \frac{1}{C^\star}$ for the optimal arm and $\mu(a) = 1 - \frac{1}{C^\star}$ for the sub-optimal arm, which satisfies the concentrability requirement. We set the following reward distributions: the optimal arm $a^\star$ is distributed according to $\mathsf{Bern}(1/2)$ and the sub-optimal arm has a deterministic reward equal to $1/2 - g$ for some $g \in (0, 1/2)$, which will be specified momentarily. It is immediate that $a^\star$ is optimal in this construction and that the MAB problem (μ, R) belongs to $\mathsf{MAB}(C^\star)$.

Lower bounding the performance of LCB. Similar arguments as before give

$$\mathbb{E}_{\mathcal{D}}[r(a^\star) - r(\hat{a})] \geq g \sum_{k=N\mu(a^\star)/2}^{2N\mu(a^\star)} \mathbb{P}(\text{LCB chooses arm } a \mid N(a^\star) = k)\mathbb{P}(N(a^\star) = k), \quad (3.32)$$

where we restrict ourselves to the event (with abuse of notation)

$$\mathcal{E} := \{\frac{1}{2}N\mu(a^\star) \leq N(a^\star) \leq 2N\mu(a^\star)\}.$$

By the definition of LCB, when $C^\star = 6$ and $\frac{1}{2}N\mu(a^\star) \leq k \leq 2N\mu(a^\star) \leq \frac{1}{3}N$, one has

$$\begin{aligned}
\mathbb{P}(\text{LCB chooses arm } a \mid N(a^\star) = k) &= \mathbb{P}\left(\hat{r}(a^\star) - \sqrt{L/N(a^\star)} \leq \frac{1}{2} - g - \sqrt{L/N(a)} \mid N(a^\star) = k\right) \\
&= \mathbb{P}\left(\hat{r}(a^\star) \leq 1/2 - g + \sqrt{L/k} - \sqrt{L/(N-k)} \mid N(a^\star) = k\right) \\
&\geq \mathbb{P}\left(\hat{r}(a^\star) \leq 1/2 - g + \sqrt{3L/N} - \sqrt{3L/(2N)} \mid N(a^\star) = k\right) \\
&> \mathbb{P}\left(\hat{r}(a^\star) \leq 1/2 - g + \sqrt{\frac{L}{4N}} \mid N(a^\star) = k\right).
\end{aligned}$$

We set $g = \min\{\sqrt{L/(4N)}, 1/2\}$. Under this choice of g, we always have

$$\mathbb{P}(\text{LCB chooses arm } a \mid N(a^\star) = k) \geq \frac{1}{2}. \quad (3.33)$$

Combine the inequalities (3.32) and (3.33) to obtain

$$\mathbb{E}_{\mathcal{D}}[r(a^\star) - r(\hat{a})] \geq g \cdot \frac{1}{2} \cdot \mathbb{P}(\mathcal{E}) \geq \frac{\min(1, \sqrt{L/N})}{8}.$$

3.8 Proofs for contextual bandits

In Section 3.8.1, we prove the sub-optimality guarantee of the LCB approach for contextual bandits stated in Theorem 3.4. In Section 3.8.2 we prove Theorem 3.5—a minimax lower bound for contextual bandits. In the end, we prove the failure of the most played arm approach in Section 3.8.3.

3.8.1 Proof of Theorem 3.4

We prove a stronger version of Theorem 3.4: Fix a deterministic expert policy π that is not necessarily optimal. We assume that

$$\max_s \frac{\rho(s)}{\mu(s,\pi(s))} \leq C^\pi.$$

Setting $\delta = 1/N$, the policy $\hat{\pi}$ returned by Algorithm 3 obeys

$$\mathbb{E}_{\mathcal{D}}[J(\pi) - J(\hat{\pi})] \lesssim \min\left(1, \widetilde{O}\left(\sqrt{\frac{S(C^\pi - 1)}{N}} + \frac{S}{N}\right)\right).$$

The statement in Theorem 3.4 can be recovered when we take $\pi = \pi^\star$.

We begin with defining a good event

$$\mathcal{E} \coloneqq \{\forall s, a : \ |r(s,a) - \hat{r}(s,a)| \leq b(s,a)\}, \tag{3.34}$$

on which the penalty function $b(s,a)$ provides a valid upper bound on the reward estimation error $r(s,a) - \hat{r}(s,a)$. With this definition in place, we state a key decomposition of the sub-optimality of the LCB method:

$$\begin{aligned}
&\mathbb{E}_{\mathcal{D}}\left[\sum_s \rho(s)\left[r(s,\pi(s)) - r(s,\hat{\pi}(s))\right]\right] \\
&= \mathbb{E}_{\mathcal{D}}\left[\sum_s \rho(s)\left[r(s,\pi(s)) - r(s,\hat{\pi}(s))\right]\mathbb{1}\{N(s,\pi(s)) = 0\}\right] =: T_1 \\
&\quad + \mathbb{E}_{\mathcal{D}}\left[\sum_s \rho(s)\left[r(s,\pi(s)) - r(s,\hat{\pi}(s))\right]\mathbb{1}\{N(s,\pi(s)) \geq 1\}\mathbb{1}\{\mathcal{E}\}\right] \coloneqq T_2 \\
&\quad + \mathbb{E}_{\mathcal{D}}\left[\sum_s \rho(s)\left[r(s,\pi(s)) - r(s,\hat{\pi}(s))\right]\mathbb{1}\{N(s,\pi(s)) \geq 1\}\mathbb{1}\{\mathcal{E}^c\}\right] \coloneqq T_3.
\end{aligned}$$

In words, the term T_1 corresponds to the error induced by missing mass, i.e., when the expert action $\pi(s)$ is not seen in the data $\mathcal{D}$. The second term T_2 denotes the error when the good event $\mathcal{E}$ takes place. The last term T_3 denotes the sub-optimality incurred under the complement event $\mathcal{E}^c$.

To avoid cluttered notation, we denote $L \coloneqq 2000\sqrt{2\log(S|\mathcal{A}|N)}$ such that $b(s,a) = \sqrt{L/N(s,a)}$ when $N(s,a) \geq 1$. These three error terms obey the following upper bounds, whose proofs are provided in subsequent subsections:

$$T_1 \leq \frac{4SC^\pi}{9N}; \tag{3.35a}$$

$$T_2 \lesssim \frac{SC^\pi}{N}L + \sqrt{\frac{S(C^\pi - 1)}{N}L} + \frac{1}{N^9}; \tag{3.35b}$$

$$T_3 \leq \frac{1}{N}. \tag{3.35c}$$

Combining the above three bounds together with the fact that $\mathbb{E}_{\mathcal{D}}[J(\pi) - J(\hat{\pi})] \leq 1$ yields that

$$\mathbb{E}_{\mathcal{D}}[J(\pi) - J(\hat{\pi})] \lesssim \min\left(1, \widetilde{O}\left(\sqrt{\frac{S(C^\pi - 1)}{N}} + \frac{SC^\pi}{N}\right)\right).$$

Note that if $C^\pi \geq 2$, the first term $\sqrt{\frac{S(C^\pi-1)}{N}}$ always dominates. Conversely, if $C^\pi < 2$, we can omit the extra C^π in the second term $\frac{SC^\pi}{N}$. This gives the desired claim in Theorem 3.4.

Proof of the bound (3.35a) on T_1

Since $r(s, \pi(s)) - r(s, \hat{\pi}(s)) \leq 1$ for any $\hat{\pi}(s)$, one has

$$\begin{aligned} T_1 &\leq \mathbb{E}_{\mathcal{D}}\left[\sum_s \rho(s)\mathbb{1}\{N(s, \pi(s)) = 0\}\right] = \sum_s \rho(s)\mathbb{P}(N(s, \pi(s)) = 0) \\ &= \sum_s \rho(s)(1 - \mu(s, \pi(s)))^N. \end{aligned}$$

Recall the assumption that $\max_s \frac{\rho(s)}{\mu(s,\pi(s))} \leq C^\pi$. We can continue the upper bound of T_1 to obtain

$$T_1 \leq \sum_s C^\pi \mu(s, \pi(s))(1 - \mu(s, \pi(s)))^N \leq \sum_s C^\pi \frac{4}{9N} = \frac{4}{9N} SC^\pi.$$

Here, the last inequality holds since $\max_{x\in[0,1]} x(1-x)^N \leq 4/(9N)$.

Proof of the bound (3.35b) on T_2

For any state $s \in \mathcal{S}$, define the total mass on sub-optimal actions to be

$$\bar{\mu}(s) \coloneqq \sum_{a: a \neq \pi(s)} \mu(s, a).$$

We can then partition the state space into the following three disjoint sets:

$$\mathcal{S}_1 \coloneqq \left\{s \mid \rho(s) < \frac{2C^\pi L}{N}\right\}, \tag{3.36a}$$

$$\mathcal{S}_2 \coloneqq \left\{s \mid \rho(s) \geq \frac{2C^\pi L}{N}, \mu(s, \pi(s)) \geq 10\bar{\mu}(s)\right\}, \tag{3.36b}$$

$$\mathcal{S}_3 \coloneqq \left\{s \mid \rho(s) \geq \frac{2C^\pi L}{N}, \mu(s, \pi(s)) < 10\bar{\mu}(s)\right\}. \tag{3.36c}$$

The set $\mathcal{S}_1$ includes the states that are "less important" in evaluating the performance of LCB. The set $\mathcal{S}_2$ captures the states for which the expert action $\pi(s)$ is drawn more frequently under the behavior distribution μ.

With this partition at hand, we can decompose the term T_2 accordingly:

$$\begin{aligned} T_2 = &\sum_{s\in\mathcal{S}_1} \rho(s)\mathbb{E}_{\mathcal{D}}\left[[r(s,\pi(s)) - r(s,\hat{\pi}(s))]\,\mathbb{1}\{N(s,\pi(s))\geq 1\}\mathbb{1}\{\mathcal{E}\}\right] =: T_{2,1} \\ &+ \sum_{s\in\mathcal{S}_2} \rho(s)\mathbb{E}_{\mathcal{D}}\left[[r(s,\pi(s)) - r(s,\hat{\pi}(s))]\,\mathbb{1}\{N(s,\pi(s))\geq 1\}\mathbb{1}\{\mathcal{E}\}\right] =: T_{2,2} \\ &+ \sum_{s\in\mathcal{S}_3} \rho(s)\mathbb{E}_{\mathcal{D}}\left[[r(s,\pi(s)) - r(s,\hat{\pi}(s))]\,\mathbb{1}\{N(s,\pi(s))\geq 1\}\mathbb{1}\{\mathcal{E}\}\right] =: T_{2,3}. \end{aligned}$$

The proof is completed by observing the following three upper bounds:

$$T_{2,1} \leq \frac{2SC^\pi L}{N}; \qquad T_{2,2} \lesssim \frac{1}{N^9}; \qquad T_{2,3} \lesssim \sqrt{\frac{C^\pi SL}{N}\min\{1, 10(C^\pi - 1)\}} \lesssim \sqrt{\frac{(C^\pi - 1)SL}{N}}.$$

Proof of the bound on $T_{2,1}$. We again use the basic fact that

$$[r(s,\pi(s)) - r(s,\hat{\pi}(s))]\,\mathbb{1}\{N(s,\pi(s))\geq 1\}\mathbb{1}\{\mathcal{E}\} \leq 1$$

to reach

$$T_{2,1} \leq \sum_{s\in\mathcal{S}_1}\rho(s) \leq \frac{2SC^\pi L}{N},$$

where the last inequality hinges on the definition (3.36a) of $\mathcal{S}_1$, namely for any $s\in\mathcal{S}_1$, one has $\rho(s) < \frac{2C^\pi L}{N}$.

Proof of the bound on $T_{2,2}$. Fix a state $s\in\mathcal{S}_2$, we define the following two sets of actions:

$$\begin{aligned} \mathcal{A}_1(s) &:= \{a \mid r(s,a) < r(s,\pi(s)), \mu(s,a) \leq L/(200N)\}, \\ \mathcal{A}_2(s) &:= \{a \mid r(s,a) < r(s,\pi(s)), \mu(s,a) > L/(200N)\}. \end{aligned}$$

Further define $A(s,a)$ to be the event that $\hat{r}(s,\pi(s)) - b(s,\pi(s)) < \hat{r}(s,a) - b(s,a)$. Clearly one has $r(s,\pi(s)) - r(s,\hat{\pi}(s)) \leq \mathbb{1}\{\cup_{a\in\mathcal{A}_1(s)\cup\mathcal{A}_2(s)}A(s,a)\}$. Consequently, we can write the following decomposition:

$$\begin{aligned} &\mathbb{E}_{\mathcal{D}}\left[[r(s,\pi(s)) - r(s,\hat{\pi}(s))]\,\mathbb{1}\{N(s,\pi(s))\geq 1\}\,\mathbb{1}\{\mathcal{E}\}\right] \\ &\quad\leq \mathbb{P}(\exists a, r(s,a) < r(s,\pi(s)), A(s,a), N(s,\pi(s))\geq 1) \\ &\quad\leq \mathbb{P}(\exists a\in\mathcal{A}_1(s), A(s,a), N(s,\pi(s))\geq 1) =: p_1(s) \\ &\qquad + \mathbb{P}(\exists a\in\mathcal{A}_2(s), A(s,a), N(s,\pi(s))\geq 1) =: p_2(s). \end{aligned}$$

As a result, $T_{2,2}$ obeys

$$T_{2,2} \leq \sum_{s\in\mathcal{S}_2}\rho(s)p_1(s) + \sum_{s\in\mathcal{S}_2}\rho(s)p_2(s), \tag{3.37}$$

which satisfy the bounds

$$\sum_{s\in\mathcal{S}_2}\rho(s)p_1(s)\lesssim\frac{1}{N^{10}},\quad\text{and}\quad\sum_{s\in\mathcal{S}_2}\rho(s)p_2(s)\lesssim\frac{1}{N^9}.$$

Taking these two bounds collectively leads us to the desired conclusion. In what follows, we focus on the proving the aforementioned two bounds.

Proof of the bound on $\sum_{s\in\mathcal{S}_2}\rho(s)p_1(s)$. Fix a state $s\in\mathcal{S}_2$. In view of the data coverage assumption, one has

$$\mu(s,\pi(s))\geq\frac{\rho(s)}{C^\pi}\geq\frac{2L}{N}. \tag{3.38}$$

In contrast, for any $a\in\mathcal{A}_1(s)$, we have

$$\mu(s,a)\leq\frac{L}{200N}. \tag{3.39}$$

Therefore one has $\mu(s,\pi(s))\gg\mu(s,a)$ for any non-expert action a. As a result, the optimal action is selected more frequently than the sub-optimal ones. It turns out that under such circumstances, the LCB algorithm picks the right action with high probability. We make this intuition precise below.

The bounds (3.38) and (3.39) together with Chernoff's bound give

$$\mathbb{P}\left(N(s,a)\leq\frac{5L}{200}\right)\geq1-\exp\left(-\frac{L}{200}\right);$$
$$\mathbb{P}\left(N(s,\pi(s))>L\right)\geq1-\exp\left(-\frac{L}{4}\right).$$

These allow us to obtain an upper bound for the function $\hat{r}-b$ evaluated at sub-optimal actions and a lower bound on $\hat{r}(s,\pi(s))-b(s,\pi(s))$. More precisely, if $N(s,a)=0$, we know that $\hat{r}(s,a)=-1$; when $1\leq N(s,a)\leq\frac{5L}{200}$, we have

$$\hat{r}(s,a)-b(s,a)\leq1-\sqrt{\frac{L}{5L/200}}\leq-5.$$

Now we turn to lower bounding the function $\hat{r}-b$ evaluated at the optimal action. When $N(s,\pi(s))>L$, one has

$$\hat{r}(s,\pi(s))-b(s,\pi(s))>-\sqrt{\frac{L}{N(s,\pi(s))}}=-1.$$

To conclude, if both $N(s,a) \leq \frac{5L}{200}$ and $N(s,\pi(s)) \geq L$ hold, we must have $\hat{r}(s,a) - b(s,a) < \hat{r}(s,\pi(s)) - b(s,\pi(s))$. Therefore we can deduce that

$$\begin{aligned}\sum_{s\in\mathcal{S}_2}\rho(s)p_1(s) &= \sum_{s\in\mathcal{S}_2}\rho(s)\mathbb{P}(\exists a\in\mathcal{A}_1(s), A(s,a), N(s,\pi(s))\geq 1)\\ &\leq (|\mathcal{A}|-1)\exp\left(-\frac{L}{200}\right)+\exp\left(-\frac{1}{4}L\right)\\ &\leq |\mathcal{A}|\exp\left(-\frac{L}{200}\right)\\ &\lesssim \frac{1}{N^{10}}.\end{aligned}$$

The last inequality comes from the choice of $L = 2000\log(2S|\mathcal{A}|N)$.

Proof of the bound on $\sum_{s\in\mathcal{S}_2}\rho(s)p_2(s)$. Before embarking on the proof of $\sum_{s\in\mathcal{S}_2}\rho(s)p_2(s) \lesssim\lesssim \frac{1}{N^9}$, it is helpful to pause and gather a few useful properties of (s,a) with $s\in\mathcal{S}_2$, $a\in\mathcal{A}_2(s)$:

1. $\rho(s) \geq \frac{2C^\pi L}{N}$ and hence $\mu(s,\pi(s)) \geq \frac{2L}{N}$ by the definition of C^π;
2. $\frac{L}{200N} \leq \mu(s,a) \leq \frac{1}{10}\mu(s,\pi(s))$;
3. $\sum_{a\in\mathcal{A}_2}\mu(s,a) \leq \frac{1}{10}\mu(s,\pi(s))$;
4. $|\mathcal{A}_2(s)| \leq 200N/L$.

In addition, we define a high probability event on which the sample sizes $N(s,a)$ concentrate around their respective means $N\mu(s,a)$:

$$\begin{aligned}\mathcal{E}_2(s) :=& \Bigg\{\frac{1}{2}N\mu(s,\pi(s)) \leq N(s,\pi(s)) \leq 2N\mu(s,\pi(s)),\\ &\forall a\in\mathcal{A}_2(s), \frac{1}{2}N\mu(s,a) \leq N(s,a) \leq 2N\mu(s,a)\Bigg\},\end{aligned}$$

which—in view of the Chernoff bound and the union bound—obeys

$$\mathbb{P}(\mathcal{E}_2(s)) \geq 1 - 1/N^9. \tag{3.40}$$

With these preparations in place, we can derive

$$\begin{aligned}p_2(s) &= \mathbb{P}(\exists a\in\mathcal{A}_2, A(s,a), N(s,\pi(s))\geq 1)\\ &\leq \mathbb{P}(\mathcal{E}_2^c(s)) + \mathbb{P}(\exists a\in\mathcal{A}_2, A(s,a), N(s,\pi(s))\geq 1, \mathcal{E}_2(s))\\ &\leq \mathbb{P}(\mathcal{E}_2^c(s)) + \sum_{a\in\mathcal{A}_2}\mathbb{P}(A(s,a), N(s,\pi(s))\geq 1, \mathcal{E}_2(s))\\ &\lesssim \frac{1}{N^9} + \frac{|\mathcal{A}_2|}{N^{10}} \lesssim \frac{1}{N^9},\end{aligned}$$

where the last line arises from the bound

$$\mathbb{P}(A(s,a), N(s,\pi(s)) \geq 1, \mathcal{E}_2(s)) \lesssim \frac{1}{N^{10}}, \tag{3.41}$$

and the cardinality upper bound $|\mathcal{A}_2(s)| \lesssim N$. This completes the bound on $\sum_{s\in\mathcal{S}_2} p(s)$.

Proof of the bound (3.41). On the event $\mathcal{E}_2(s)$, one must have $N(s,a) \geq 1$ and $N(s,\pi(s)) \geq 1$. Therefore, we can define

$$\epsilon := \sqrt{\frac{L}{N(s,a)}} - \sqrt{\frac{L}{N(s,\pi(s))}} \quad \text{and} \quad \Delta = r(s,\pi(s)) - r(s,a),$$

and obtain the following bound on the conditional probability

$$\begin{aligned}
&\mathbb{P}\left(\hat{r}(s,a) - \sqrt{\frac{L}{N(s,a)}} \geq \hat{r}(s,\pi(s)) - \sqrt{\frac{L}{N(s,\pi(s))}} \;\middle|\; N(s,\pi(s)), N(s,a), \mathcal{E}_2\right) \\
&\leq \exp\left(-2\frac{N(s,a)N(s,\pi(s))(\epsilon+\Delta)^2}{N(s,a)+N(s,\pi(s))} \;\middle|\; N(s,\pi(s)), N(s,a), \mathcal{E}_2\right),
\end{aligned}$$

where the inequality arises from Lemma 3.13. Note that under event $\mathcal{E}_2(s)$ and the property $\mu(s,a) \leq \frac{1}{10}\mu(s,\pi(s))$, we have $N(s,\pi(s)) \geq 4N(s,a)$ and thus $\epsilon \geq \frac{1}{2}\sqrt{\frac{L}{N(s,a)}}$. This allows us to further upper bound the probability as

$$\begin{aligned}
&\mathbb{P}\left(\hat{r}(s,a) - \sqrt{\frac{L}{N(s,a)}} \geq \hat{r}(s,\pi(s)) - \sqrt{\frac{L}{N(s,\pi(s))}} \;\middle|\; N(s,\pi(s)), N(s,a), \mathcal{E}_2\right) \\
&\leq \exp\left(-N(s,a)(\epsilon+\Delta)^2\right) \\
&\leq \exp\left(-\left(\frac{1}{2}\sqrt{L} + \sqrt{N(s,a)}\Delta\right)^2\right) \\
&\leq \exp\left(-\frac{1}{4}L\right) \lesssim \frac{1}{N^{10}},
\end{aligned}$$

under the choice of $L = 2000\log(2S|\mathcal{A}|N)$. Since this upper bound holds for any configuration of $N(s,a)$ and $N(s,\pi(s))$, one has the desired claim. □

Proof of the bound on $T_{2,3}$. On the good event $\mathcal{E}$, we know that

$$\begin{aligned}
r(s,\pi(s)) - r(s,\hat{\pi}(s)) &\leq r(s,\pi(s)) - [\hat{r}(s,\hat{\pi}(s)) - b(s,\hat{\pi}(s))] \\
&\leq r(s,\pi(s)) - [\hat{r}(s,\pi(s)) - b(s,\pi(s))] \\
&\leq 2b(s,\pi(s)).
\end{aligned}$$

Here the middle line arises from the definition of the LCB algorithm, i.e., $\hat{\pi}(s) \in \arg\max_a \hat{r}(s,a) - b(s,a)$ for each s. Substitute this upper bound into the definition of T_2 to obtain

$$\begin{aligned} T_{2,3} &\leq 2\sum_{s\in\mathcal{S}_3} \rho(s)\mathbb{E}_{\mathcal{D}}\left[b(s,\pi(s))\mathbb{1}\{N(s,\pi(s)) \geq 1\}\mathbb{1}\{\mathcal{E}\}\right] \\ &= 2\sum_{s\in\mathcal{S}_3} \rho(s)\mathbb{E}_{\mathcal{D}}\left[\sqrt{\frac{L}{N(s,\pi(s))}}\mathbb{1}\{N(s,\pi(s)) \geq 1\}\mathbb{1}\{\mathcal{E}\}\right] \\ &\leq 2\sqrt{L}\sum_{s\in\mathcal{S}_3} \rho(s)\mathbb{E}_{\mathcal{D}}\left[\sqrt{\frac{1}{N(s,\pi(s))\vee 1}}\mathbb{1}\{N(s,\pi(s)) \geq 1\}\right], \end{aligned}$$

where we have used the definition of $b(s,a)$. Lemma 3.14 tells us that there exists a universal constant $c > 0$ such that

$$\mathbb{E}_{\mathcal{D}}\left[\sqrt{\frac{1}{N(s,\pi(s))\vee 1}}\mathbb{1}\{N(s,\pi(s)) \geq 1\}\right] \leq \frac{c}{\sqrt{N\mu(s,\pi(s))}}.$$

As a result, we reach the conclusion that

$$T_{2,3} \leq 2\sqrt{L}\sum_{s\in\mathcal{S}_3} \rho(s)\frac{c}{\sqrt{N\mu(s,\pi(s))}}.$$

In view of the assumption $\max_s \rho(s)/\mu(s,\pi(s)) \leq C^\pi$, one further has

$$T_{2,3} \leq 2c\sqrt{\frac{C^\pi L}{N}}\sum_{s\in\mathcal{S}_3}\sqrt{\rho(s)} \leq 2c\sqrt{\frac{C^\pi L}{N}}\sqrt{S}\sqrt{\sum_{s\in\mathcal{S}_3}\rho(s)},$$

with the last inequality arising from Cauchy-Schwarz's inequality. The desired bound on $T_{2,3}$ follows from the following simple fact regarding $\sum_{s\in\mathcal{S}_3}\rho(s)$:

$$\sum_{s\in\mathcal{S}_3}\rho(s) \leq \min\left\{1, 10(C^\pi - 1)\right\}. \tag{3.42}$$

Proof of the inequality (3.42). The upper bound 1 is trivial to see. To achieve the other upper bound, we first use the assumption $\max_s \rho(s)/\mu(s,\pi(s)) \leq C^\pi$ to see

$$\sum_{s\in\mathcal{S}_3}\rho(s) \leq \sum_{s\in\mathcal{S}_3} C^\pi\mu(s,\pi(s)) \leq 10C^\pi\sum_{s\in\mathcal{S}_3}\bar{\mu}(s).$$

Here the last relation follows from the definition of $\mathcal{S}_3$. Note that

$$\sum_{s\in\mathcal{S}_3}\bar{\mu}(s) \leq \sum_s \bar{\mu}(s) = 1 - \sum_s \mu(s,\pi(s)) \leq 1 - \frac{1}{C^\pi},$$

where we have reused the assumption $\max_s \rho(s)/\mu(s,\pi(s)) \leq C^\pi$. Taking the previous two inequalities collectively yields the final claim. □

Proof of the bound (3.35c) on T_3

It is not hard to see that

$$\sum_s \rho(s) \left[r(s, \pi(s)) - r(s, \hat{\pi}(s)) \right] \mathbb{1}\{N(s, \pi(s)) \geq 1\} \leq 1,$$

which further implies

$$T_3 \leq \mathbb{E}_{\mathcal{D}} \left[\mathbb{1}\{\mathcal{E}^c\} \right] = \mathbb{P}(\mathcal{E}^c).$$

It then boils down to upper bounding the probability $\mathbb{P}(\mathcal{E}^c)$. The proof is similar in spirit to that of Lemma 3.3.

Fix a state-action pair (s, a). If $N(s, a) = 0$, one clearly has $-1 = \hat{r}(s, a) - b(s, a) \leq r(s, a) \leq \hat{r}(s, a) + b(s, a) = 1$. Therefore we concentrate on the case when $N(s, a) \geq 1$. Apply the Hoeffding's inequality to see that for any $\delta_1 \in (0, 1)$, one has

$$\mathbb{P}\left(|\hat{r}(s, a) - r(s, a)| \geq \sqrt{\frac{\log(2/\delta_1)}{2N(s, a)}} \mid N(s, a) \right) \leq \delta_1.$$

In particular, setting $\delta_1 = \delta/(S|\mathcal{A}|)$ yields

$$\mathbb{P}\left(|\hat{r}(s, a) - r(s, a)| \geq \sqrt{\frac{\log(2S|\mathcal{A}|/\delta)}{2N(s, a)}} \mid N(s, a) \right) \leq \frac{\delta}{S|\mathcal{A}|}, \tag{3.43}$$

Recall that $b(s, a)$ is defined such that when $N(s, a) \geq 1$,

$$b(s, a) = \sqrt{\frac{2000 \log(2S|\mathcal{A}|/\delta)}{N(s, a)}}.$$

Since the inequality (3.43) holds for any $N(s, a)$, we have for any fixed (s, a),

$$\mathbb{P}\left(|\hat{r}(s, a) - r(s, a)| \geq b(s, a) \right) \leq \frac{\delta}{S|\mathcal{A}|}.$$

Taking a union bound over $\mathcal{S} \times \mathcal{A}$ leads to the conclusion that $\mathbb{P}(\mathcal{E}^c) \leq \delta$, and hence $T_3 \leq \delta$. Taking $\delta = 1/N$ gives the advertised result.

3.8.2 Proof of Theorem 3.5

We prove the lower bound differently for the following regimes: $C^\star = 1$, $C^\star \geq 2$, and $C^\star \in (1, 2)$. When $C^\star = 1$, the offline RL problem reduces to the imitation learning problem in contextual bandits, whose lower bound has been shown in the paper Rajaraman et al. (2020). When $C^\star \in (1, 2)$ or $C^\star \geq 2$, we generalize the lower bound given for the multi-armed bandits with different choices of initial distributions. In what follows, we detail the proofs for each regime.

The case when $C^\star = 1$. When $C^\star = 1$, one has $d^\star(s,a) = \mu(s,a)$ for any (s,a) pair. This recovers the imitation learning problem, where the rewards are also included in the dataset. Thus the lower bound proved in Lemma 3.6 is applicable, which comes from a modified version of Theorem 6 in the paper Rajaraman et al. (2020):

$$\inf_{\hat{\pi}} \sup_{(\rho,\mu,R)\in\mathsf{CB}(1)} \mathbb{E}_{\mathcal{D}}[J(\pi^\star) - J(\hat{\pi})] \gtrsim \min\left(1, \frac{S}{N}\right). \tag{3.44}$$

The case when $C^\star \geq 2$. Fix a contextual bandit instance (ρ, μ, R), define the loss/sub-optimality of an estimated policy π to be

$$\mathcal{L}(\pi; (\rho,\mu,R)) \coloneqq J(\pi^\star) - J(\hat{\pi}).$$

We intend to show that when $C^\star \geq 2$,

$$\inf_{\hat{\pi}} \sup_{(\rho,\mu,R)\in\mathsf{CB}(C^\star)} \mathbb{E}_{\mu\otimes R}[\mathcal{L}(\pi; (\rho,\mu,R))] \gtrsim \min\left(1, \sqrt{\frac{SC^\star}{N}}\right). \tag{3.45}$$

Our proof follows the standard recipe of proving minimax lower bounds, namely, we first construct a family of hard contextual bandit instances, and then apply Fano's inequality to obtain the desired lower bound.

Construction of hard instances. Consider a CB with state space $\mathcal{S} \coloneqq \{1, 2, \ldots, S\}$. Set the initial distribution $\rho_0(s) = 1/S$ for any $s \in \mathcal{S}$. Each state $s \in \mathcal{S}$ is associated with two actions a_1 and a_2. The behavior distribution for each s, a is specified below

$$\mu_0(s,a_1) = \frac{1}{S} - \frac{1}{SC^\star} \qquad \text{and} \qquad \mu_0(s,a_2) = \frac{1}{SC^\star}.$$

It is easy to check that for any reward distribution R, one has $(\rho_0, \mu_0, R) \in \mathsf{CB}(C^\star)$. It remains to construct a set of reward distributions that are nearly indistinguishable from the data. To achieve this goal, we leverage the Gilbert-Varshamov lemma (cf. Lemma 3.15) to obtain a set $\mathcal{V} \subseteq \{-1,1\}^S$ that obeys (1) $|\mathcal{V}| \geq \exp(S/8)$ and (2) $\|\boldsymbol{v}_1 - \boldsymbol{v}_2\|_1 \geq S/2$ for any $\boldsymbol{v}_1 \boldsymbol{v}_2 \in \mathcal{V}$ with $\boldsymbol{v}_1 \neq \boldsymbol{v}_2$. With this set $\mathcal{V}$ in place, we can continue to construct the following set of Bernoulli reward distributions

$$\mathcal{R} \coloneqq \left\{ \left\{ \mathsf{Bern}\left(\frac{1}{2}\right), \mathsf{Bern}\left(\frac{1}{2} + v_s\delta\right) \right\}_{s\in\mathcal{S}} \mid \boldsymbol{v} \in \mathcal{V} \right\}.$$

Here $\delta \in (0, 1/3)$ is a parameter that will be specified later. Each element $\boldsymbol{v} \in \mathcal{V}$ is mapped to a reward distribution such that for the state s, the reward distribution associated with (s, a_2) is $\mathsf{Bern}(\frac{1}{2} + v_s\delta)$. In view of the second property of the set $\mathcal{V}$, one has for any policy π and any two different reward distributions $R_1, R_2 \in \mathcal{R}$,

$$\mathcal{L}(\pi; (\rho_0,\mu_0,R_1)) + \mathcal{L}(\pi; (\rho_0,\mu_0,R_2)) \geq \frac{\delta}{4}.$$

Application of Fano's inequality. Now we are ready to apply Fano's inequality, that is

$$\inf_{\hat{\pi}} \sup_{(\rho_0,\mu_0,R)|R\in\mathcal{R}} \mathbb{E}_{\mu_0\otimes R}[\mathcal{L}(\pi;(\rho_0,\mu_0,R))] \geq \frac{\delta}{8}\left(1-\frac{N\max_{i\neq j}\mathsf{KL}\left(\mu\otimes R_i\|\mu\otimes R_j\right)+\log 2}{\log|\mathcal{R}|}\right).$$

It then remains to control $\max_{i\neq j}\mathsf{KL}\left(\mu\otimes R_i\|\mu\otimes R_j\right)$ and $\log|\mathcal{R}|$. For the latter quantity, we have

$$\log|\mathcal{R}| = \log|\mathcal{V}| \geq S/8,$$

where the inequality comes from the first property of the set $\mathcal{V}$. With regards to the KL divergence, one has

$$\max_{i\neq j}\mathsf{KL}\left(\mu\otimes R_i\|\mu\otimes R_j\right) \leq S\cdot\frac{1}{SC^\star}\cdot 16\delta^2 = \frac{16\delta^2}{C^\star}.$$

As a result, we conclude that as long as

$$\frac{200N\delta^2}{SC^\star}\leq 1,$$

one has

$$\inf_{\hat{\pi}} \sup_{(\rho_0,\mu_0,R)|R\in\mathcal{R}} \mathcal{L}(\pi;(\rho_0,\mu_0,R)) \gtrsim \delta.$$

To finish the proof, we can set $\delta = \sqrt{\frac{SC^\star}{200N}}$ when $\sqrt{\frac{SC^\star}{200N}} < \frac{1}{3}$, and $\delta = \frac{1}{3}$ otherwise. This yields the desired lower bound (3.45).

The case when $C^\star \in (1,2)$. We intend to show that

$$\inf_{\hat{\pi}} \sup_{(\rho,\mu,R)\in\mathsf{CB}(C^\star)} \mathbb{E}[\mathcal{L}(\pi;(\rho,\mu,R))] \gtrsim \min\left(C^\star-1,\sqrt{\frac{S(C^\star-1)}{N}}\right). \tag{3.46}$$

The proof is similar to that of the previous case, with the difference lying in the construction of ρ_0 and μ_0.

Construction of hard instances. Consider a CB with state space $\mathcal{S} := \{0,1,2,\ldots,S\}$ and action space $\mathcal{A} := \{a_1,a_2\}$. Set the initial distribution $\rho_0(s) = (C^\star-1)/S$ for any $1\leq s\leq S$ and $\rho_0(0) = 2-C^\star$. Each state $1\leq s\leq S$ is associated with two actions a_1 and a_2 such that

$$\mu_0(s,a_1) = \mu_0(s,a_2) = \frac{C^\star-1}{SC^\star}.$$

In contrast, for $s=0$, one has a single action a_1 with $\mu_0(0,a_1) = \frac{2-C^\star}{C^\star}$. Similar to the above case, we have for any reward distribution R, that $(\rho_0,\mu_0,R)\in\mathsf{CB}(C^\star)$.

We deploy essentially the same family $\mathcal{R}$ of reward distributions as before with an additional reward of $R(0, a_1) \equiv 0$ on state $s = 0$. As a result, one can show that for any policy π and any two different reward distributions $R_1, R_2 \in \mathcal{R}$,

$$\mathcal{L}(\pi; (\rho_0, \mu_0, R_1)) + \mathcal{L}(\pi; (\rho_0, \mu_0, R_2)) \geq \frac{\delta}{4}(C^\star - 1).$$

Application of Fano's inequality. Fano's inequality tells us that

$$\inf_{\hat{\pi}} \sup_{(\rho_0,\mu_0,R)|R\in\mathcal{R}} \mathbb{E}[\mathcal{L}(\pi; (\rho_0, \mu_0, R))] \geq \frac{\delta}{8}\left(1 - \frac{N \max_{i\neq j} \mathsf{KL}\left(\mu \otimes R_i \| \mu \otimes R_j\right) + \log 2}{S/8}\right).$$

In the current case, we have

$$\max_{i\neq j} \mathsf{KL}\left(\mu \otimes R_i \| \mu \otimes R_j\right) \leq S \cdot \frac{C^\star - 1}{SC^\star} \cdot 16\delta^2 = \frac{16(C^\star - 1)}{C^\star}\delta^2.$$

As before, setting

$$\delta = \min\left(\sqrt{\frac{SC^\star}{200(C^\star - 1)N}}, \frac{1}{3}\right)$$

yields the lower bound

$$\inf_{\hat{\pi}} \sup_{(\rho_0,\mu_0,R)|R\in\mathcal{R}} \mathbb{E}[\mathcal{L}(\pi; (\rho_0, \mu_0, R))] \gtrsim \min\left(C^\star - 1, \sqrt{\frac{SC^\star(C^\star - 1)}{N}}\right) \gtrsim \min\left(C^\star - 1, \sqrt{\frac{S(C^\star - 1)}{N}}\right).$$

Putting the pieces together. We are now in position to summarize and simplify the three established lower bounds (3.44), (3.45), and (3.46).

When $C^\star = 1$, the claim in Theorem 3.5 is identical to the bound (3.44).

When $C^\star \geq 2$, we have from the bound (3.45) that

$$\inf_{\hat{\pi}} \sup_{(\rho,\mu,R)\in\mathsf{CB}(C^\star)} \mathbb{E}[\mathcal{L}(\pi; (\rho, \mu, R))] \gtrsim \min\left(1, \sqrt{\frac{SC^\star}{N}}\right) \asymp \min\left(1, \sqrt{\frac{S(C^\star - 1)}{N}}\right).$$

Further notice that

$$\sqrt{\frac{S(C^\star - 1)}{N}} \geq \sqrt{\frac{S}{N}} \geq \min\left(1, \frac{S}{N}\right).$$

The claimed lower bound in Theorem 3.5 arises.

In the end, when $C^\star \in (1, 2)$, we know from the bounds (3.44) and (3.46) that

$$\inf_{\hat{\pi}} \sup_{(\rho,\mu,R)\in\mathsf{CB}(C^\star)} \mathbb{E}[\mathcal{L}(\pi; (\rho, \mu, R))] \gtrsim \max\left\{\min\left(1, \frac{S}{N}\right), \min\left(C^\star - 1, \sqrt{\frac{S(C^\star - 1)}{N}}\right)\right\}.$$

Elementary calculations reveal that

$$\max\left\{\min\left(1, \frac{S}{N}\right), \min\left(C^\star - 1, \sqrt{\frac{S(C^\star - 1)}{N}}\right)\right\} \asymp \min\left(1, \sqrt{\frac{S(C^\star - 1)}{N}} + \frac{S}{N}\right),$$

which completes the proof.

3.8.3 Proof of Proposition 3.3

We design the hard instance with state space $\{s_0, s_1\}$ and action space $\{a_0, a_1\}$. Only under state (s_0, a_0) we can possibly get non-zero reward, and all other state-action pairs give 0 rewards. We set $d^\star(s_0) = d^\star(s_0, a_0) = C^\star - 1 - \epsilon$, $d^\star(s_1) = 2 - C^\star + \epsilon$ for some small $\epsilon > 0$. The constraints introduced by concentrability are $\mu(s_0, a_0) \geq (C^\star - 1 - \epsilon)/C^\star$, $\mu(s_1) \geq (2 - C^\star + \epsilon)/C^\star$.

We set $\mu(s_0, a_0) = (C^\star - 1 - \epsilon)/C^\star$, $\mu(s_0, a_1) = (C^\star - 1)/C^\star$, $\mu(s_1) = (2 - C^\star + \epsilon)/C^\star$. One can verify that $d^\star, \mu$ are valid probability distributions and the concentrability assumption still holds.

In this case, since $\mu(s_0, a_0) < \mu(s_0, a_1)$, the algorithm fails to identify the optimal arm a_0 as $N \to \infty$. This incurs the following expected sub-optimality

$$\lim_{N \to \infty} \mathbb{E}_{\mathcal{D}}[J(\pi^\star) - J(\hat{\pi})] = d^\star(s_0) \geq C^\star - 1 - \epsilon.$$

Setting $\epsilon \to 0$ gives us the conclusion.

3.9 Proofs for MDPs

We begin by presenting several Bellman equations for discounted MDPs, which is followed by the proof of Lemma 3.1. We then prove general properties of Algorithm 4 under the clean event (3.22). These include the contraction properties given in Proposition 3.4 as well as the value difference lemma (cf. Lemma 3.2). Next, we prove the LCB sub-optimality Theorem 3.6. In the end, we prove the minimax lower bound followed by an analysis of imitation learning with an alternative data coverage assumption.

3.9.1 Bellman and Bellman-like equations

Given a discounted MDP, the Bellman value operator $\mathcal{T}_\pi$ associated with a policy π is defined as

$$\mathcal{T}_\pi V \coloneqq r_\pi + \gamma P_\pi V. \tag{3.47}$$

It is well-known that V^π is the unique solution to $\mathcal{T}_\pi V = V$, which is known as the Bellman equation.

In addition to V^π, other quantities in an MDP also follow a Bellman-like equation, which we briefly review here. For discounted occupancy measures, simple algebra gives

$$d_\pi = (1-\gamma)\rho + \gamma d_\pi P_\pi \quad \Rightarrow \quad d_\pi = (1-\gamma)\rho(I - \gamma P_\pi)^{-1}, \tag{3.48}$$

$$d^\pi = (1-\gamma)\rho^\pi + \gamma d^\pi P^\pi \quad \Rightarrow \quad d^\pi = (1-\gamma)\rho^\pi(I - \gamma P^\pi)^{-1}. \tag{3.49}$$

3.9.2 Proof of Lemma 3.1

The proof is similar to that of Lemma 3.3. For completeness, we include it here.

From the algorithmic design, it is clear (in particular the Q update and the monotonic improvement step) that

$$V_t(s) \in [0, V_{\max}], \qquad \text{for all } s \in \mathcal{S} \text{ and } t \geq 0.$$

As a result, for a fixed tuple (s, a, t), if $m_t(s, a) = 0$, one has

$$\left|r(s,a) + \gamma P_{s,a} \cdot V_t - r_t(s,a) - \gamma P^t_{s,a} \cdot V_{t-1}\right| \leq 1 + \gamma V_{\max} = V_{\max} \leq b_t(s,a).$$

When $m_t(s, a) \geq 1$, exploiting the independence between V_t and $P^t_{s,a}$ and using Hoeffding's inequality to obtain

$$\mathbb{P}\left(\left|r(s,a) + \gamma P_{s,a} \cdot V_t - r_t(s,a) - \gamma P^t_{s,a} \cdot V_{t-1}\right| \geq V_{\max}\sqrt{L/m_t(s,a)} \mid m_t(s,a)\right) \leq 2\exp\left(-2L\right).$$

Since the above inequality holds for any $m_t(s, a)$, one necessarily has

$$\mathbb{P}\left(\left|r(s,a) + \gamma P_{s,a} \cdot V_t - r_t(s,a) - \gamma P^t_{s,a} \cdot V_{t-1}\right| \geq b_t(s,a)\right) \leq 2\exp\left(-2L\right).$$

Taking a union bound over s, a and $t \in \{0, \ldots, T\}$ and setting $\delta_1 = \frac{\delta}{2S|\mathcal{A}|(T+1)}$ finishes the proof.

3.9.3 Proof of Proposition 3.4

We prove the claims one by one.

Proof of $V_{t-1} \leq V_t$. The first claim $V_{t-1} \leq V_t$ is directly implied by line 15 of Algorithm 4.

Proof of $V_t \leq V^{\pi_t}$. For the second claim $V_t \leq V^{\pi_t}$, it suffices to prove that $V_t \leq \mathcal{T}_{\pi_t} V_t$. Indeed, $V_t \leq \mathcal{T}_{\pi_t} V_t$ together with the monotonicity of the Bellman's operator yield the conclusion $V_t \leq V^{\pi_t}$. In what follows, we prove $V_t \leq \mathcal{T}_{\pi_t} V_t$ via induction.

The base case $V_0 \leq \mathcal{T}_{\pi_0} V_0$ holds due to zero initialization. Hence from now on, we assume $V_k \leq \mathcal{T}_{\pi_k} V_k$ for $0 \leq k \leq t-1$ and intend to prove $V_t \leq \mathcal{T}_{\pi_t} V_t$. We split the proof into two cases.

- If $V_{t-1}(s) \geq \max_a\{r_{t-1}(s,a) - b_{t-1}(s,a) + \gamma P^{t-1}_{s,a} \cdot V_{t-1}\}$, the algorithm sets $V_t(s) = V_{t-1}(s)$ and $\pi_t(s) = \pi_{t-1}(s)$. Consequently, we have

  $$V_t(s) = V_{t-1}(s) \leq (\mathcal{T}_{\pi_{t-1}} V_{t-1})(s) \leq (\mathcal{T}_{\pi_t} V_t)(s),$$

  where the first inequality arises from the induction hypothesis and the last one holds since $V_{t-1} \leq V_t$ and $\pi_t(s) = \pi_{t-1}(s)$.

- If instead, the algorithm sets $Q_t(s,a) = r_t(s,a) - b_t(s,a) + \gamma P^t_{s,a} \cdot V_{t-1}$ with $\pi_t(s) = \arg\max_a Q_t(s,a)$ and $V_t(s) = Q_t(s,\pi_t(s))$, then we have

$$\begin{aligned}(\mathcal{T}_{\pi_t} V_t)(s) =& r(s,\pi_t(s)) + \gamma P_{s,\pi_t(s)} \cdot V_t \\ \geq& r(s,\pi_t(s)) + \gamma P_{s,\pi_t(s)} \cdot V_{t-1} \\ =& r_t(s,\pi_t(s)) - b_t(s,\pi_t(s)) + \gamma P^t_{s,\pi_t(s)} \cdot V_{t-1} \\ & + b_t(s,\pi_t(s)) + r(s,\pi_t(s)) - r_t(s,\pi_t(s)) + \gamma(P_{s,\pi_t(s)} - P^t_{s,\pi_t(s)}) \cdot V_{t-1} \\ =& V_t(s) + b_t(s,\pi_t(s)) + r(s,\pi_t(s)) - r_t(s,\pi_t(s)) + \gamma(P_{s,\pi_t(s)} - P^t_{s,\pi_t(s)}) \cdot V_{t-1} \\ \geq& V_t(s),\end{aligned}$$

  where the first inequality is due to $V_{t-1} \leq V_t$ and the last inequality holds under the clean event $\mathcal{E}_{\text{MDP}}$.

This finishes the proof of $V_t \leq \mathcal{T}_{\pi_t} V_t$ and hence $V_t \leq V^{\pi_t}$. The claim $V^{\pi_t} \leq V^\star$ is trivial to see.

Proof of $Q_t \leq r + \gamma P V_{t-1} \leq r + \gamma P V_t$. Since $V_t \geq V_{t-1}$, we have

$$\begin{aligned}r(s,a) + \gamma P_{s,a} \cdot V_t \geq& r(s,a) + \gamma P_{s,a} \cdot V_{t-1} \\ =& r_t(s,a) - b_t(s,a) + \gamma P^t_{s,a} \cdot V_{t-1} \\ & + b_t(s,a) + r(s,a) - r_t(s,a) + \gamma(P_{s,a} - P^t_{s,a}) \cdot V_{t-1} \\ \geq& Q_t(s,a),\end{aligned}$$

where the last inequality holds under $\mathcal{E}_{\text{MDP}}$.

Proof of $Q^\pi - Q_t \leq \gamma P^\pi (Q^\pi - Q_{t-1}) + 2b_t$. Let $Q(:,\pi) \in \mathbb{R}^S$ be a vector with elements $Q^\pi(s,\pi(s))$. By definition, one has

$$\begin{aligned}& Q^\pi(s,a) - Q_t(s,a) \\ &= r(s,a) + \gamma P_{s,a} \cdot V^\pi - r_t(s,a) + b_t(s,a) - \gamma P^t_{s,a} \cdot V_{t-1} \\ &= \gamma P_{s,a} \cdot V^\pi - \gamma P_{s,a} \cdot V_{t-1} + b_t(s,a) + r(s,a) - r_t(s,a) + \gamma(P_{s,a} - P^t_{s,a}) \cdot V_{t-1} \\ &\leq \gamma P_{s,a} \cdot (Q^\pi(:,\pi) - Q_{t-1}(:,\pi)) + b_t(s,a) + r(s,a) - r_t(s,a) + \gamma(P_{s,a} - P^t_{s,a}) \cdot V_{t-1} \\ &\leq \gamma P_{s,a} \cdot (Q^\pi(:,\pi) - Q_{t-1}(:,\pi)) + 2b_t(s,a).\end{aligned}$$

Here, the first inequality comes from the fact that $V_{t-1} \geq \max_a Q_{t-1}(:,a) \geq Q_t(:,\pi)$ and the last inequality again holds under $\mathcal{E}_{\text{MDP}}$.

3.9.4 Proof of Lemma 3.2

In view of Proposition 3.4, one has $V_t \leq V^{\pi_t}$. Therefore we obtain

$$\mathbb{E}_\rho \left[V^\pi(s) - V^{\pi_t}(s)\right] \leq \mathbb{E}_\rho \left[V^\pi(s) - V_t(s)\right] \leq \mathbb{E}_\rho \left[V^\pi(s) - V_t^{\text{mid}}(s)\}\right],$$

where the last inequality arises from the monotonicity imposed by Algorithm 4. Note that $V_t^{\text{mid}}(s) = Q_t(s, \pi_t^{\text{mid}})$ and that π_t^{mid} is greedy with respect to Q_t. We can continue the upper bound as

$$\mathbb{E}_\rho \left[V^\pi(s) - V^{\pi_t}(s)\right] \leq \mathbb{E}_\rho \left[Q^\pi(s, \pi(s)) - Q_t(s, \pi_t^{\text{mid}})\}\right] \leq \mathbb{E}_\rho \left[Q^\pi(s, \pi(s)) - Q_t(s, \pi(s))\right].$$

Rewriting using the matrix notation gives

$$\mathbb{E}_\rho \left[V^\pi(s) - V^{\pi_t}(s)\right] \leq \mathbb{E}_\rho \left[Q^\pi(s, \pi(s)) - Q_t(s, \pi(s))\right] = \rho^\pi(Q^\pi - Q_t). \tag{3.50}$$

Now we are ready to apply the third claim in Proposition 3.4 to deduce that on the event $\mathcal{E}_{\text{MDP}}$:

$$\begin{aligned}
Q^\pi - Q_t &\leq \gamma P^\pi (Q^\pi - Q_{t-1}) + 2b_t \leq \gamma P^\pi \left[\gamma P^\pi (Q^\pi - Q_{t-2}) + 2b_{t-1}\right] + 2b_t \\
&\leq \cdots \\
&\leq \gamma^t (P^\pi)^t (Q^\pi - Q_0) + 2\sum_{j=1}^{t} (\gamma P^\pi)^{t-j} b_j \\
&\leq \frac{\gamma^t}{1-\gamma}\mathbf{1} + 2\sum_{j=1}^{t} (\gamma P^\pi)^{t-j} b_j.
\end{aligned}$$

Here $\mathbf{1}$ denotes the all-one vector with dimension $S|\mathcal{A}|$, and the last inequality arises from the fact that $Q^\pi - Q_0 = Q^\pi \leq (1-\gamma)^{-1}\mathbf{1}$. Multiplying both sides the of the equation above by ρ^π, we conclude that

$$\rho^\pi(Q^\pi - Q_t) \leq \frac{\gamma^t}{1-\gamma} + 2\sum_{j=1}^{t} \rho^\pi (\gamma P^\pi)^{t-j} b_j = \frac{\gamma^t}{1-\gamma} + 2\sum_{j=1}^{t} v_{t-j}^\pi b_j, \tag{3.51}$$

where we use the definition of $v_k^\pi = \rho^\pi (\gamma P^\pi)^k$. Combine the inequalities (3.50) and (3.51) to reach the desired result.

3.9.5 Proof of Theorem 3.6

Similar to the proof given for contextual bandits, we prove a stronger result than Theorem 3.6. Fix any deterministic expert policy π. Assume that the data coverage assumption holds, that is

$$\max_{s,a} \frac{d^\pi(s,a)}{\mu(s,a)} \leq C^\pi.$$

Then for all $C^\pi \geq 1$, Algorithm 4 with $\delta = 1/N$ achieves

$$\mathbb{E}_{\mathcal{D}}\left[J(\pi) - J(\hat{\pi})\right] \lesssim \min\left(\frac{1}{1-\gamma}, \sqrt{\frac{SC^\pi}{(1-\gamma)^5 N}}\right). \tag{3.52}$$

In addition, if $1 \leq C^\pi \leq 1 + \frac{L\log(N)}{200(1-\gamma)N}$, then we have a tighter performance upper bound

$$\mathbb{E}_{\mathcal{D}}\left[J(\pi) - J(\hat{\pi})\right] \lesssim \min\left(\frac{1}{1-\gamma}, \frac{S}{(1-\gamma)^4 N}\right). \tag{3.53}$$

The result in Theorem 3.6 can be recovered by taking $\pi = \pi^\star$.

We split the proof into two cases: (1) the general case when $C^\pi \geq 1$ and (2) the regime where $C^\pi \leq 1 + L/(200m)$.

The general case when $C^\pi \geq 1$. The proof of the general case follows similar steps as those in the proof of Theorem 3.4. We first decompose the expected sub-optimality into three terms:

$$\begin{aligned}
&\mathbb{E}_{\mathcal{D}}\left[\sum_s \rho(s)[V^\pi(s) - V^{\pi_T}(s)]\right] \\
&= \mathbb{E}_{\mathcal{D}}\left[\sum_s \rho(s)[V^\pi(s) - V^{\pi_T}(s)]\, \mathbb{1}\{\exists t \leq T, m_t(s,\pi(s)) = 0\}\right] =: T_1 \\
&\quad + \mathbb{E}_{\mathcal{D}}\left[\sum_s \rho(s)[V^\pi(s) - V^{\pi_T}(s)]\, \mathbb{1}\{\forall t \leq T, m_t(s,\pi(s)) \geq 1\}\, \mathbb{1}\{\mathcal{E}_{\text{MDP}}\}\right] =: T_2 \\
&\quad + \mathbb{E}_{\mathcal{D}}\left[\sum_s \rho(s)[V^\pi(s) - V^{\pi_T}(s)]\, \mathbb{1}\{\forall t \leq T, m_t(s,\pi(s)) \geq 1\}\, \mathbb{1}\{\mathcal{E}^c_{\text{MDP}}\}\right] =: T_3.
\end{aligned}$$

Similar to before, the first term T_1 captures the sub-optimality incurred by the missing mass on the expert action $\pi(s)$. The second term T_2 is the sub-optimality under the clean event $\mathcal{E}_{\text{MDP}}$, while the last one T_3 denotes the sub-optimality suffered under the complement event $\mathcal{E}^c_{\text{MDP}}$, on which the empirical average of Q-function falls outside the constructed confidence interval.

As we will show in subsequent sections, these error terms satisfy the following upper bounds:

$$T_1 \leq \frac{4SC^\pi(T+1)^2}{9(1-\gamma)^2 N}; \tag{3.54a}$$

$$T_2 \leq \frac{\gamma^T}{1-\gamma} + 32\frac{1}{(1-\gamma)^2}\sqrt{\frac{LSC^\pi(T+1)}{N}}; \tag{3.54b}$$

$$T_3 \leq V_{\max}\delta. \tag{3.54c}$$

Setting $\delta = 1/N$, $T = \log N/(1-\gamma)$ and noting that $\gamma^T \le 1/N$ yield that

$$\mathbb{E}_{\mathcal{D}}\left[J(\pi) - J(\hat{\pi})\right] \lesssim \left(\sqrt{\frac{SC^\pi}{(1-\gamma)^5 N}} + \frac{SC^\pi}{(1-\gamma)^4 N}\right).$$

Note that we always have $\mathbb{E}_{\mathcal{D}}\left[J(\pi) - J(\hat{\pi})\right] \le \frac{1}{1-\gamma}$. In the interesting regime of $\frac{SC^\pi}{(1-\gamma)^3 N} \le 1$, the first term above always dominates. This gives the desired claim (3.52).

The case when $C^\pi \le 1 + L/(200m)$. Under this circumstance, the following lemma proves useful.

Lemma 3.5. *For any deterministic policy $\hat{\pi}$, one has*

$$J(\pi) - J(\hat{\pi}) \le V_{\max}^2 \mathbb{E}_{s\sim d_\pi}\left[\mathbb{1}\left\{\hat{\pi}(s) \ne \pi(s)\right\}\right]. \tag{3.55}$$

Proof. In view of the performance difference lemma in Kakade and Langford (2002, Lemma 6.1), one has

$$\begin{aligned} J(\pi) - J(\hat{\pi}) &= \frac{1}{1-\gamma}\mathbb{E}_{s\sim d_\pi}\left[Q^{\hat{\pi}}(s,\pi(s)) - Q^{\hat{\pi}}(s,\hat{\pi}(s))\right] \\ &= \frac{1}{1-\gamma}\mathbb{E}_{s\sim d_\pi}\left[\left[Q^{\hat{\pi}}(s,\pi(s)) - Q^{\hat{\pi}}(s,\hat{\pi}(s))\right]\mathbb{1}\left\{\hat{\pi}(s) \ne \pi(s)\right\}\right] \\ &\le V_{\max}^2 \mathbb{E}_{s\sim d_\pi}\left[\mathbb{1}\left\{\hat{\pi}(s) \ne \pi(s)\right\}\right]. \end{aligned}$$

Here the last line uses the fact that $Q^{\hat{\pi}}(s,\pi(s)) - Q^{\hat{\pi}}(s,\hat{\pi}(s)) \le V_{\max}$. □

Lemma 3.5 links the sub-optimality of a policy to its disagreement with the optimal policy. With Lemma 3.5 at hand, we can continue to decompose the expected sub-optimality into:

$$\begin{aligned} &\mathbb{E}_{\mathcal{D}}\left[\sum_s \rho(s)[V^\pi(s) - V^{\pi_T}(s)]\right] \\ &\le V_{\max}^2\, \mathbb{E}_{\mathcal{D}}[\mathbb{E}_{s\sim d_\pi}[\mathbb{1}\{\pi_T(s) \ne \pi(s)\}]] \\ &= V_{\max}^2\, \mathbb{E}_{\mathcal{D}}[\mathbb{E}_{s\sim d_\pi}[[\mathbb{1}\{\pi_T(s) \ne \pi(s)\}\,\mathbb{1}\{\exists t \le T, m_t(s,\pi(s)) = 0\}]] =: T_1' \\ &\quad + V_{\max}^2\, \mathbb{E}_{\mathcal{D}}[\mathbb{E}_{s\sim d_\pi}[[\mathbb{1}\{\pi_T(s) \ne \pi(s)\}\,\mathbb{1}\{\forall t \le T, m_t(s,\pi(s)) \ge 1\}]] =: T_2' \end{aligned}$$

We bound each term according to

$$T_1' \le \frac{4SC^\pi(T+1)^2}{9(1-\gamma)^2 N}; \tag{3.56a}$$

$$T_2' \lesssim \frac{SC^\pi LT}{(1-\gamma)^2 N} + \frac{ST^{10}}{(1-\gamma)^2 N^9}. \tag{3.56b}$$

The claimed bound (3.53) follows by taking $\delta = 1/N$ and $T = \log N/(1-\gamma)$.

Proof of the bound (3.54a) on T_1 and the bound (3.56a) on T_1'

Since for any $s \in \mathcal{S}$, $V^{\pi}(s) - V^{\pi_T}(s) \leq V_{\max}$ one has

$$T_1 \leq V_{\max}\, \mathbb{E}_{\mathcal{D}} \left[\sum_s \rho(s)\, \mathbb{1}\{\exists t \leq T, m_t(s, \pi(s)) = 0\} \right] = V_{\max} \sum_s \rho(s) \mathbb{P}\left(\exists t \leq T, m_t(s, \pi(s)) = 0\right).$$

The definition of the normalized occupancy measure (3.5b) entails $\rho(s) \leq d^{\pi}(s, \pi(s))$ and thus

$$\frac{\rho(s)}{\mu(s, \pi(s))} \leq \frac{1}{1-\gamma} \cdot \frac{d^{\pi}(s, \pi(s))}{\mu(s, \pi(s))} \leq \frac{C^{\pi}}{1-\gamma}.$$

Here the last relation follows from the data coverage assumption. Combine the above two inequalities to see that

$$\begin{aligned} T_1 &\leq V_{\max} \sum_s \frac{C^{\pi}}{1-\gamma} \mu(s, \pi(s)) \mathbb{P}\left(\exists t \leq T, m_t(s, \pi(s)) = 0\right) \\ &= \frac{C^{\pi}}{(1-\gamma)^2} \sum_s \mu(s, \pi(s))\, \mathbb{P}\left(\exists t \leq T, m_t(s, \pi(s)) = 0\right) \\ &\leq \frac{C^{\pi}}{(1-\gamma)^2} \sum_{t=0}^{T} \sum_s \mu(s, \pi(s))\, \mathbb{P}\left(m_t(s, \pi(s)) = 0\right), \end{aligned}$$

where in the penultimate line, we identify $V_{\max}$ with $1/(1-\gamma)$, and the last relation is by the union bound. Direct calculations yield

$$\mathbb{P}\left(m_t(s, \pi(s)) = 0\right) = (1 - \mu(s, \pi(s)))^m,$$

which further implies

$$T_1 \leq \frac{C^{\pi}(T+1)}{(1-\gamma)^2} \sum_s \mu(s, \pi(s))(1 - \mu(s, \pi(s)))^m \leq \frac{4C^{\pi}S(T+1)}{9(1-\gamma)^2 m} = \frac{4C^{\pi}S(T+1)^2}{9(1-\gamma)^2 N}.$$

Here, we have used $\max_{x \in [0,1]} x(1-x)^m \leq 4/(9m)$ and the fact that $m = N/(T+1)$.

The bound (3.56a) on T_1' follows from exactly the same argument as above, except that we replace ρ with d^{π}.

Proof of the bound (3.54b) on T_2

Lemma 3.2 asserts that on the clean event $\mathcal{E}_{\mathrm{MDP}}$, one has

$$\begin{aligned} T_2 &\leq \frac{\gamma^T}{1-\gamma} + 2\sum_{t=1}^{T} \mathbb{E}_{\mathcal{D},\nu^{\pi}_{T-t}}\left[b_t(s,\pi(s))\,\mathbb{1}\{m_t(s,\pi(s)) \geq 1\}\right] \\ &= \frac{\gamma^T}{1-\gamma} + 2\sum_{t=1}^{T} \mathbb{E}_{\mathcal{D},\nu^{\pi}_{T-t}}\left[V_{\max}\sqrt{\frac{L}{m_t(s,\pi(s))}}\,\mathbb{1}\{m_t(s,\pi(s)) \geq 1\}\right] \\ &\leq \frac{\gamma^T}{1-\gamma} + 2\sum_{t=1}^{T} \mathbb{E}_{\nu^{\pi}_{T-t}}\left[16V_{\max}\sqrt{\frac{L}{m\mu(s,\pi(s))}}\right]. \end{aligned} \tag{3.57}$$

Here, we substitute in the definition of $b_t(s,a)$ in the middle line and the last inequality arises from Lemma 3.14 with $c_{1/2} \leq 16$.

By definition of $\nu^{\pi}_k = \rho^{\pi}(\gamma P^{\pi})^k$, we have $\sum_{k=0}^{\infty} \nu^{\pi}_k = d^{\pi}/(1-\gamma)$. Therefore, one has

$$\begin{aligned} \sum_{t=1}^{T} \mathbb{E}_{\nu^{\pi}_{T-t}}\left[\frac{1}{\sqrt{\mu(s,\pi(s))}}\right] &= \sum_{t=1}^{T}\sum_{s} \nu^{\pi}_{T-t}(s,\pi(s))\frac{1}{\sqrt{\mu(s,\pi(s))}} \\ &= \sum_{s}\left[\sum_{t=1}^{T} \nu^{\pi}_{T-t}(s,\pi(s))\right]\frac{1}{\sqrt{\mu(s,\pi(s))}} \\ &\leq \sum_{s}\frac{d^{\pi}(s,\pi(s))}{1-\gamma}\frac{1}{\sqrt{\mu(s,\pi(s))}}. \end{aligned}$$

We then apply the concentrability assumption and the Cauchy–Schwarz inequality to deduce that

$$\begin{aligned} \sum_{t=1}^{T} \mathbb{E}_{\nu^{\pi}_{T-t}}\left[\frac{1}{\sqrt{\mu(s,\pi(s))}}\right] &\leq \sqrt{\frac{C^{\pi}}{(1-\gamma)^2}}\sum_{s}\sqrt{d^{\pi}(s,\pi(s))} \\ &\leq \sqrt{\frac{C^{\pi}}{(1-\gamma)^2}}\sqrt{S}\sqrt{\sum_{s} d^{\pi}(s,\pi(s))} \\ &= \frac{\sqrt{SC^{\pi}}}{1-\gamma}. \end{aligned}$$

Substitute the above bound into the inequality (3.57) to arrive at the conclusion

$$T_2 \leq \frac{\gamma^T}{1-\gamma} + 32\frac{1}{(1-\gamma)^2}\sqrt{\frac{LSC^{\pi}}{m}}.$$

The proof is completed by noting that $m = N/(T+1)$.

Proof of the bound (3.54c) on T_3

It is easy to see that

$$\sum_s \rho(s)[V^{\pi}(s) - V^{\pi_T}(s)]\, \mathbb{1}\{\forall s, t, m_t(s, \pi(s)) \geq 1\} \leq V_{\max},$$

which further implies

$$T_3 \leq V_{\max}\, \mathbb{E}_{\mathcal{D}}[\mathbb{1}\{\mathcal{E}^c_{\mathrm{MDP}}\}] = V_{\max}\, \mathbb{P}(\mathcal{E}^c_{\mathrm{MDP}}) \leq V_{\max}\delta.$$

Here, the last bound relies on Lemma 3.1.

Proof of the bound (3.56b) on T_2'

Partition the state space into the following two disjoint sets:

$$\mathcal{S}_1 := \left\{ s \mid d_\pi(s) < \frac{2C^\pi L}{m} \right\}, \tag{3.58a}$$

$$\mathcal{S}_2 := \left\{ s \mid d_\pi(s) \geq \frac{2C^\pi L}{m} \right\}, \tag{3.58b}$$

In words, the set $\mathcal{S}_1$ includes the states that are less important in evaluating the performance of LCB. We can then decompose the term T_2' accordingly:

$$\begin{aligned} T_2' = V_{\max}^2 \sum_{s\in\mathcal{S}_1} d_\pi(s)\, \mathbb{E}_{\mathcal{D}}[\mathbb{1}\{\pi_T(s) \neq \pi(s)\}\, \mathbb{1}\{\forall t, m_t(s, \pi(s)) \geq 1\}] =: T_{2,1} \\ + V_{\max}^2 \sum_{s\in\mathcal{S}_2} d_\pi(s)\, \mathbb{E}_{\mathcal{D}}[\mathbb{1}\{\pi_T(s) \neq \pi(s)\}\, \mathbb{1}\{\forall t, m_t(s, \pi(s)) \geq 1\}] =: T_{2,2}. \end{aligned}$$

The proof is completed by observing the following two upper bounds:

$$T_{2,1} \leq \frac{2SC^\pi LT}{(1-\gamma)^2 N}, \qquad \text{and} \qquad T_{2,2} \lesssim \frac{S}{(1-\gamma)^2}\left(\frac{T}{N}\right)^9.$$

Proof of the bound on $T_{2,1}$. We again use the basic fact that

$$\mathbb{E}_{\mathcal{D}}[\mathbb{1}\{\pi_T(s) \neq \pi(s)\}\, \mathbb{1}\{\forall s, t, m_t(s, \pi(s)) \geq 1\}] \leq 1$$

to reach

$$T_{2,1} \leq V_{\max}^2 \sum_{s\in\mathcal{S}_1} d_\pi(s) \leq \frac{2SC^\pi L}{(1-\gamma)^2 m},$$

where the last inequality hinges on the definition of $\mathcal{S}_1$ given in (3.58a), namely for any $s \in \mathcal{S}_1$, one has $d_\pi(s) < \frac{2C^\pi L}{m}$. Identifying m with $N/(T+1)$ concludes the proof.

Proof of the bound on $T_{2,2}$. Equivalently, we can write $T_{2,2}$ as

$$T_{2,2} = V_{\max}^2 \sum_{s \in \mathcal{S}_2} d_\pi(s) \mathbb{P}\left(\pi_T(s) \neq \pi(s),\ m_t(s, \pi(s)) \geq 1\ \forall t\right).$$

By inspecting Algorithm 4, one can realize the following inclusion

$$\{\pi_T(s) \neq \pi(s)\} \subseteq \{\pi_0(s) \neq \pi(s)\} \cup \{\exists 0 \leq t \leq T-1 \text{ and } \exists a \neq \pi(s), Q_{t+1}(s,a) \geq Q_{t+1}(s, \pi(s))\}.$$

Indeed, if $\pi_0(s) = \pi(s)$ and for all t, $Q_{t+1}(s, \pi(s)) > \max_{a \neq \pi(s)} Q_{t+1}(s,a)$, LCB would select the expert action in the end, i.e., $\pi_T(s) = \pi(s)$. Therefore, we can upper bound $T_{2,2}$ as

$$\begin{aligned} T_{2,2} \leq\ & V_{\max}^2 \sum_{s \in \mathcal{S}_2} d_\pi(s) \mathbb{P}\left(\pi_0(s) \neq \pi(s), m_t(s, \pi(s)) \geq 1\ \forall t\right) =: \beta_1 \\ & + V_{\max}^2 \sum_{s \in \mathcal{S}_2} d_\pi(s) \mathbb{P}\left(\exists t \leq T-1, \exists a \neq \pi(s), Q_{t+1}(s,a) \geq Q_{t+1}(s, \pi(s)), m_t(s, \pi(s)) \geq 1\ \forall t\right) =: \beta_2. \end{aligned}$$

In the sequel, we bound β_1 and β_2 in the reverse order.

Bounding β_2. Fix a state $s \in \mathcal{S}_2$. In view of the data coverage assumption, one has

$$\mu(s, \pi(s)) \geq \frac{1}{C^\pi} d_\pi(s) \geq \frac{1}{C^\pi} \frac{2C^\pi L}{m} = \frac{2L}{m}. \tag{3.59}$$

In contrast, for any $a \neq \pi(s)$, since $C^\pi \leq 1 + \frac{L}{200m}$, we have

$$\mu(s,a) \leq \sum_{a \neq \pi(s)} \mu(s,a) \leq 1 - \frac{1}{C^\pi} \leq \frac{L}{200m}, \tag{3.60}$$

where the middle inequality reuses the concentrability assumption. One has $\mu(s, \pi(s)) \gg \mu(s,a)$ for any non-expert action a. As a result, the expert action is pulled more frequently than the others. It turns out that under such circumstances, the LCB algorithm picks the expert action with high probability. We shall make this intuition precise below.

The bounds (3.59) and (3.60) together with Chernoff's bound give

$$\begin{aligned} \mathbb{P}\left(m_t(s,a) \leq \frac{5L}{200}\right) &\geq 1 - \exp\left(-\frac{L}{200}\right); \\ \mathbb{P}\left(m_t(s, \pi(s)) \geq L\right) &\geq 1 - \exp\left(-\frac{L}{4}\right). \end{aligned}$$

These allow us to obtain an upper bound for the function Q_{t+1} evaluated at non-expert actions and a lower bound on $Q_{t+1}(s, \pi(s))$. More precisely, when $m_t(s,a) \leq \frac{5L}{200}$, we have

$$\begin{aligned} Q_t(s,a) &= r_t(s,a) - b_t(s,a) + \gamma P_{s,a}^t \cdot V_{t-1} \\ &= r_t(s,a) - V_{\max}\sqrt{\frac{L}{m_t(s,a) \vee 1}} + \gamma P_{s,a}^t \cdot V_{t-1} \\ &\leq 1 - V_{\max}\sqrt{\frac{L}{5L/200}} + \gamma V_{\max} \\ &\leq -5V_{\max}. \end{aligned}$$

Here we used the fact that $L \geq 70$. Now we turn to lower bounding the function Q_t evaluated at the optimal action. When $m_t(s, \pi(s)) \geq L$, one has

$$Q_t(s,\pi(s)) = r_t(s,\pi(s)) - V_{\max}\sqrt{\frac{L}{m_t(s,\pi(s))}} + \gamma P_{s,\pi(s)}^t \cdot V_{t-1} \geq -V_{\max}.$$

To conclude, if both $m_t(s,a) \leq \frac{5L}{200}$ and $m_t(s,\pi(s)) \geq L$ hold, we must have $Q_t(s,a) < Q_t(s,\pi(s))$. Therefore we can deduce that

$$\begin{aligned} &\mathbb{P}\left(\exists 0 \leq t \leq T \text{ and } \exists a \neq \pi(s), Q_t(s,a) \geq Q_t(s,\pi(s)), m_t(s,\pi(s)) \geq 1 \ \forall t\right) \\ &\quad \leq \sum_{0 \leq t \leq T} \mathbb{P}\left(\exists a \neq \pi(s), Q_t(s,a) \geq Q_t(s,\pi(s)), m_t(s,\pi(s)) \geq 1 \ \forall t\right) \\ &\quad \leq \sum_{0 \leq t \leq T-1} \left\{ (|\mathcal{A}|-1)\exp\left(-\frac{L}{200}\right) + \exp\left(-\frac{1}{4}L\right) \right\} \\ &\quad \leq T|\mathcal{A}|\exp\left(-\frac{L}{200}\right), \end{aligned}$$

which further implies

$$\begin{aligned} \beta_2 &\leq V_{\max}^2 \sum_{s \in \mathcal{S}_2} d_\pi(s) T|\mathcal{A}|\exp\left(-\frac{L}{200}\right) \\ &\leq TV_{\max}|\mathcal{A}| \cdot \frac{1}{1-\gamma}\exp\left(-\frac{L}{200}\right) \\ &\lesssim Tm^{-9}. \end{aligned}$$

Bounding β_1. In fact, the analysis of β_2 has revealed that with high probability, $\pi(s)$ is the most played arm among all actions. More precisely, we have

$$\begin{aligned}\beta_1 &\leq V_{\max}^2 \sum_{s\in\mathcal{S}_2} d_\pi(s)\mathbb{P}\left(\pi_0(s)\neq\pi(s)\right)\\ &\leq V_{\max}^2 \sum_{s\in\mathcal{S}_2} d_\pi(s)\left\{\mathbb{P}\left(\max_a m_0(s,a)\geq\frac{5L}{200}\right)+\mathbb{P}\left(m_0(s,\pi(s))\leq L\right)\right\}\\ &\leq V_{\max}^2|\mathcal{A}|\exp\left(-\frac{L}{200}\right)\lesssim\frac{1}{(1-\gamma)^2 m^{-9}}.\end{aligned}$$

Combine the bounds on β_1 and β_2 to arrive at the claim on $T_{2,2}$.

3.9.6 Proof of Theorem 3.7

Similar to the proof of the lower bound for contextual bandits, we split the proof into three cases: (1) $C^\star = 1$, (2) $C^\star \geq 2$, and (3) $C^\star \in (1,2)$. For $C^\star = 1$, we adapt the lower bound from episodic imitation learning (Rajaraman et al., 2020) to the discounted case. For both $C^\star \in (1,2)$ and $C^\star \geq 2$, we rely on the construction of the MDP in the paper Lattimore and Hutter (2012), which reduces the policy learning problem in MDP to a bandit problem. The key difference is that in our construction, we need to carefully design the initial distribution ρ to incorporate the effect of $C^\star$ in the lower bound.

The case when $C^\star = 1$. In this case we have $\mu(s,a) = d^\star(s,a)$ for all (s,a) pairs, which is the imitation learning setting. We adapt the lower bound given in Rajaraman et al. (2020) for episodic imitation learning to the discounted case and obtain the following lemma:

Lemma 3.6. *When $C^\star = 1$, one has*

$$\inf_{\hat{\pi}}\sup_{(\rho,\mu,P,R)\in\mathsf{MDP}(1)}\mathbb{E}_{\mathcal{D}}[J(\pi^\star)-J(\hat{\pi})]\gtrsim\min\left\{\frac{1}{1-\gamma},\frac{S}{(1-\gamma)^2 N}\right\}. \tag{3.61}$$

We defer the proof to Section 3.9.6, which follows exactly the analysis by Rajaraman et al. (2020) except for changing the setting from episodic to discounted.

The case when $C^\star \geq 2$. When $C^\star \geq 2$, we intend to show that

$$\inf_{\hat{\pi}}\sup_{(\rho,\mu,P,R)\in\mathsf{MDP}(C^\star)}\mathbb{E}_{\mathcal{D}}[J(\pi^\star)-J(\hat{\pi})]\gtrsim\min\left(\frac{1}{1-\gamma},\sqrt{\frac{SC^\star}{(1-\gamma)^3 N}}\right). \tag{3.62}$$

We adopt the following construction of the hard MDP instance from the work Lattimore and Hutter (2012).

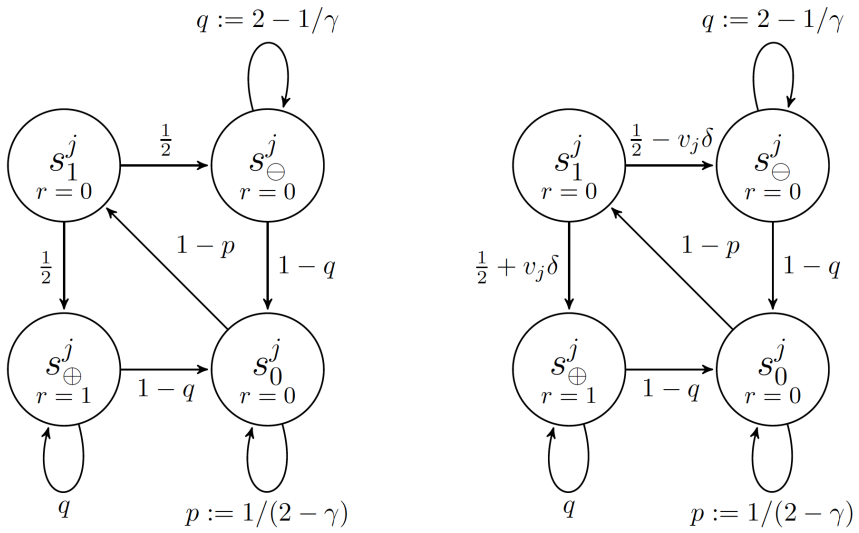

Figure 3.5: Illustration of one replica in the hard MDP_h. The left plot shows the transition probabilities from (s_1^j, a_1) and the right plot shows them from (s_1^j, a_2).

Construction of hard instances. Consider the MDP which consists of $S/4$ replicas of MDPs in Figure 3.5 and an extra state s_{-1}. The total number of states is $S+1$. For each replica, we have four states $s_0, s_1, s_\oplus, s_\ominus$. There is only one action, say a_1, in all the states except s_1, which has two actions a_1, a_2. The rewards are all deterministic. In addition, the transitions for states $s_0, s_\oplus, s_\ominus$ are shown in the diagram. More specifically, we have $\mathbb{P}(s_\oplus^j \mid s_1^j, a_1) = \mathbb{P}(s_\ominus^j \mid s_1^j, a_1) = 1/2$ and $\mathbb{P}(s_\oplus^j \mid s_1^j, a_2) = 1/2 + v_j\delta$, and $\mathbb{P}(s_\ominus^j \mid s_1^j, a_2) = 1/2 - v_j\delta$. Here $v_j \in \{-1, +1\}$ is the design choice associated with the j-th replica and $\delta \in [0, 1/4]$ will be specified later. Clearly, if $v_j = 1$, the optimal action at s_1^j is a_2, otherwise, the optimal one is a_1. Under the extra state s_{-1}, there is only one action with reward 0 which transits to itself with probability 1. We use s_i^j to denote state i in j-th replica, where $j \in [S/4]$. Based on the description above, the only parameter in this MDP is the transition dynamics associated with the state s_1^j. We will later specify how to set these for each s_1^j. The single replica has the following important properties:

1. The probabilities p, q are designed such that the three states $s_0, s_\ominus, s_\oplus$ are mostly absorbing, while any action in s_1 will lead to immediate transition to $s_\oplus$ or $s_\ominus$.

2. The state $s_\oplus$ is the only state that gives reward 1, which helps reduce the MDP problem to a bandit one: the MDP only depends on the choice of transition probabilities at state s_1^j; once a policy reaches state s_1 it should choose the action most likely to lead to state $\oplus$ whereupon it will either be rewarded or punished (visit state $\oplus$ or $\ominus$). Eventually, it will return to state 1 where the whole process repeats.

We also need to specify the initial distribution ρ_0 and the behavior distribution μ_0. When

$C^\star \geq 2$, we set the initial distribution ρ_0 to be uniformly distributed on the state s_0 in all the $S/4$ replicas, i.e., $\forall j \in [S/4], \rho_0(s_0^j) = 4/S$. From $d^\star = (1-\gamma)\rho(I - \gamma P^{\pi^\star})^{-1}$ we can derive $d^\star$ as follows:

$$
\begin{aligned}
d^\star(s_0^j) &= \frac{8}{(2+\gamma)S}, \qquad d^\star(s_1^j) = \frac{8\gamma(1-\gamma)}{(2-\gamma)(2+\gamma)S} \in \left[\frac{1-\gamma}{S}, \frac{4(1-\gamma)}{S}\right], \\
d^\star(s_\oplus^j) &= \frac{\gamma(\frac{1}{2}\mathbb{1}\{v_j = -1\} + (\frac{1}{2}+\delta)\mathbb{1}\{v_j = 1\})}{2(1-\gamma)} \cdot d^\star(s_1^j), \\
d^\star(s_\ominus^j) &= \frac{\gamma(\frac{1}{2}\mathbb{1}\{v_j = 1\} + (\frac{1}{2}-\delta)\mathbb{1}\{v_j = -1\})}{2(1-\gamma)} \cdot d^\star(s_1^j), \qquad d^\star(s_{-1}) = 0.
\end{aligned}
$$

This allows us to construct the behavior distribution μ_0 as follows:

$$
\begin{aligned}
\mu_0(s_0^j) &= \frac{d^\star(s_0^j)}{C^\star}, \qquad \mu_0(s_1^j, a_2) = \frac{d^\star(s_1^j)}{C^\star}, \qquad \mu_0(s_1^j, a_1) = d^\star(s_1^j) \cdot \left(1 - \frac{1}{C^\star}\right) \\
\mu_0(s_\oplus^j) &= \frac{3}{4} \cdot \frac{\gamma}{2(1-\gamma)C^\star} \cdot d^\star(s_1^j), \qquad \mu_0(s_\ominus^j) = \frac{1}{2} \cdot \frac{\gamma}{2(1-\gamma)c^\star} \cdot d^\star(s_1^j), \\
\mu_0(s_{-1}) &= 1 - \sum_j (\mu_0(s_0^j) + \mu_0(s_1^j) + \mu_0(s_\oplus^j) + \mu_0(s_\ominus^j))
\end{aligned}
$$

It is easy to check that for any $v_j \in \{-1, 1\}$, $\delta \in [0, 1/4]$, one has $\mu_0(s_{-1}) > 0$, and more importantly

$$(\rho_0, \mu_0, P, R) \in \mathsf{MDP}(C^\star).$$

Since in this construction of MDP, the reward distribution is deterministic and fixed, and we only need to change the transition dynamics P, which is governed by the choice of δ and $v_j{}_{1 \leq k \leq S/4}$. Hence we write the loss/sub-optimality of a policy π w.r.t. a particular design of P as

$$\mathcal{L}(\pi; P) = J_P(\pi^\star) - J_P(\pi).$$

Our target then becomes

$$\inf_{\hat{\pi}} \sup_{(\rho_0, \mu_0, P, R) \in \mathsf{MDP}(C^\star)} \mathbb{E}[\mathcal{L}(\hat{\pi}; P)] \gtrsim \min\left(\frac{1}{1-\gamma}, \sqrt{\frac{SC^\star}{(1-\gamma)^3 N}}\right).$$

It remains to construct a set of transition probabilities (determined by δ and $\boldsymbol{v}$) that are nearly indistinguishable given the data. Similar to the construction in the lower bound for contextual bandits, we leverage the Gilbert-Varshamov lemma (cf. Lemma 3.15) to obtain a set $\mathcal{V} \subseteq \{-1, 1\}^{S/4}$ that obeys (1) $|\mathcal{V}| \geq \exp(S/32)$ and (2) $\|\boldsymbol{v}_1 - \boldsymbol{v}_2\|_1 \geq S/8$ for any $\boldsymbol{v}_1, \boldsymbol{v}_2 \in \mathcal{V}$ with $\boldsymbol{v}_1 \neq \boldsymbol{v}_2$. Each element $\boldsymbol{v} \in \mathcal{V}$ is mapped to a transition probability at s_1^j such that the probability of transiting to $s_\oplus^j$ associated with (s_1^j, a_2) is $\frac{1}{2} + v_j\delta$. We denote the resulting set of transition probabilities as $\mathcal{P}$. We record a useful characteristic of this family $\mathcal{P}$ of transition dynamics below, which results from the second property of the set $\mathcal{V}$.

Lemma 3.7. *For any policy π and any two different transition probabilities $P_1, P_2 \in \mathcal{P}$, the following holds:*

$$\mathcal{L}(\pi; P_1) + \mathcal{L}(\pi; P_2) \geq \frac{\delta}{32(1-\gamma)}.$$

Application of Fano's inequality. We are now ready to apply Fano's inequality, that is

$$\inf_{\hat{\pi}} \sup_{P \in \mathcal{P}} \mathbb{E}[\mathcal{L}(\hat{\pi}; P)] \geq \frac{\delta}{64(1-\gamma)} \left(1 - \frac{N \max_{i \neq j} \mathsf{KL}\left(\mu_0 \otimes P_i \| \mu_0 \otimes P_j\right) + \log 2}{\log|\mathcal{P}|}\right).$$

It remains to controlling $\max_{i \neq j} \mathsf{KL}\left(\mu_0 \otimes P_i \| \mu_0 \otimes P_j\right)$ and $\log|\mathcal{P}|$. For the latter quantity, we have

$$\log|\mathcal{P}| = \log|\mathcal{V}| \geq S/32,$$

where the inequality comes from the first property of the set $\mathcal{V}$. With regards to the KL divergence, one has

$$\max_{i \neq j} \mathsf{KL}\left(\mu_0 \otimes P_i \| \mu_0 \otimes P_j\right) \leq \frac{4(1-\gamma)}{SC^\star} \cdot \frac{S}{4} \cdot 16\delta^2 = \frac{16(1-\gamma)\delta^2}{C^\star},$$

since $\mu_0(s_1^j, a_2) \in [\frac{1-\gamma}{SC^\star}, \frac{4(1-\gamma)}{SC^\star}]$. As a result, we conclude that as long as

$$\frac{c_3(1-\gamma)N\delta^2}{SC^\star} \leq 1$$

for some universal constant c_3, one has

$$\inf_{\hat{\pi}} \sup_{P} \mathbb{E}[[\mathcal{L}(\hat{\pi}; P)] \gtrsim \frac{\delta}{1-\gamma}.$$

To finish the proof, we can set $\delta = \sqrt{\frac{SC^\star}{c_3(1-\gamma)N}}$ when $\sqrt{\frac{SC^\star}{c_3(1-\gamma)N}} < \frac{1}{4}$ and $\delta = \frac{1}{4}$ otherwise. This yields the desired lower bound (3.62).

The case when $C^\star \in (1,2)$. We intend to show that when $C^\star \in (1,2)$,

$$\inf_{\hat{\pi}} \sup_{(\rho,\mu,P,R) \in \mathsf{MDP}(C^\star)} \mathbb{E}_{\mathcal{D}}[J(\pi^\star) - J(\hat{\pi})] \gtrsim \min\left(\frac{C^\star - 1}{1-\gamma}, \sqrt{\frac{S(C^\star-1)}{(1-\gamma)^3 N}}\right). \tag{3.63}$$

The proof is similar to that of the previous case but with a different construction for ρ_0 and μ_0.

Construction of the hard instance. Let $\rho_0(s_0^j) = 4(C^\star - 1)/S$, $\rho_0(s_{-1}) = 2 - C^\star$. From $d^\star = (1-\gamma)\rho(I - \gamma P^{\pi^\star})^{-1}$ we can derive $d^\star$ as follows.

$$d^\star(s_0^j) = \frac{8(C^\star - 1)}{(2+\gamma)S}, \qquad d^\star(s_1^j) = \frac{8\gamma(1-\gamma)(C^\star - 1)}{(2-\gamma)(2+\gamma)S} \in \left[\frac{(1-\gamma)(C^\star - 1)}{S}, \frac{4(1-\gamma)(C^\star - 1)}{S}\right],$$
$$d^\star(s_\oplus^j) = \frac{\gamma(\frac{1}{2}\mathbb{1}\{v_j = -1\} + (\frac{1}{2}+\delta)\mathbb{1}\{v_j = 1\})}{2(1-\gamma)} \cdot d^\star(s_1^j),$$
$$d^\star(s_\ominus^j) = \frac{\gamma(\frac{1}{2}\mathbb{1}\{v_j = 1\} + (\frac{1}{2}-\delta)\mathbb{1}\{v_j = -1\})}{2(1-\gamma)} \cdot d^\star(s_1^j), \qquad d^\star(s_{-1}) = 2 - C^\star.$$

This allows us to construct the behavior distribution μ_0 as follows

$$\mu_0(s_0^j) = \frac{d^\star(s_0^j)}{C^\star}, \qquad \mu_0(s_1^j, a_1) = \mu_0(s_1^j, a_2) = \frac{d^\star(s_1^j)}{C^\star}$$
$$\mu_0(s_\oplus^j) = \frac{3}{4} \cdot \frac{\gamma}{2(1-\gamma)} \cdot d^\star(s_1^j), \qquad \mu_0(s_\ominus^j) = \frac{1}{2} \cdot \frac{\gamma}{2(1-\gamma)} \cdot d^\star(s_1^j),$$
$$\mu_0(s_{-1}) = 1 - \sum_j (\mu_0(s_0^j) + \mu_0(s_1^j) + \mu_0(s_\oplus^j) + \mu_0(s_\ominus^j))$$

Again, one can check that for any $v_j \in \{-1, 1\}$ and $\delta \in [0, 1/4]$, we have $\mu_0(s_{-1}) > 0$ and

$$(\rho_0, \mu_0, P, R) \in \mathsf{MDP}(C^\star).$$

We use the same family $\mathcal{P}$ of transition probabilities as before. Following the same proof as Lemma 3.7 and noting that the initial distribution is multiplied by an extra $C^\star - 1$ factor, we know that for any policy π, and any two different distributions $P_1, P_2 \in \mathcal{P}$,

$$\mathcal{L}(\pi; P_1) + \mathcal{L}(\pi; P_2) \geq \frac{(C^\star - 1)\delta}{32(1-\gamma)}.$$

Application of Fano's inequality. Now we are ready to apply Fano's inequality, that is

$$\inf_{\hat{\pi}} \sup_{P \in \mathcal{P}} \mathbb{E}[\mathcal{L}(\hat{\pi}; P)] \geq \frac{\delta}{64(1-\gamma)} \left(1 - \frac{N \max_{i \neq j} \mathsf{KL}\left(\mu_0 \otimes P_i \| \mu_0 \otimes P_j\right) + \log 2}{\log|\mathcal{P}|}\right).$$

Now the KL divergence satisfies

$$\mathsf{KL}(\mu_0 \otimes P_i \| \mu_0 \otimes P_j) \leq \frac{4(1-\gamma)(C^\star - 1)}{SC^\star} \cdot \frac{S}{4} \cdot 16\delta^2 = \frac{16(1-\gamma)(C^\star - 1)\delta^2}{C^\star}.$$

Here the first inequality comes from that $\mu_0(s_1^j) = \frac{c_2(1-\gamma)(C^\star - 1)}{SC^\star}$ for some constant $c_2 \in [1, 4]$. As a result, we conclude that as long as

$$\frac{c_3(1-\gamma)(C^\star - 1)N\delta^2}{SC^\star} \leq 1$$

for some universal constant c_3, one has

$$\inf_{\hat{\pi}} \sup_{P \in \mathcal{P}} \mathbb{E}[\mathcal{L}(\pi; P)] \gtrsim \frac{(C^\star - 1)\delta}{1-\gamma}.$$

To finish the proof, we can set $\delta = \sqrt{\frac{SC^\star}{c_3(1-\gamma)(C^\star-1)N}}$ when $\sqrt{\frac{SC^\star}{c_3(1-\gamma)(C^\star-1)N}} < \frac{1}{4}$, and $\delta = \frac{1}{4}$ otherwise. This yields the desired lower bound (3.63).

Putting the pieces together. Now we are in position to summarize and simplify the three established lower bounds (3.61), (3.62), and (3.63).

When $C^\star = 1$, the claim in Theorem 3.7 is identical to the bound (3.61).

When $C^\star \geq 2$, we have from the bound (3.62) that

$$\inf_{\hat{\pi}} \sup_{P} \mathbb{E}[\mathcal{L}(\hat{\pi}; P)] \gtrsim \min\left(\frac{1}{1-\gamma}, \sqrt{\frac{SC^\star}{(1-\gamma)^3 N}}\right) \asymp \min\left(\frac{1}{1-\gamma}, \sqrt{\frac{S(C^\star-1)}{(1-\gamma)^3 N}}\right).$$

Further notice that

$$\sqrt{\frac{S(C^\star-1)}{(1-\gamma)^3 N}} \geq \sqrt{\frac{S}{(1-\gamma)^4 N}} \geq \min\left(\frac{1}{1-\gamma}, \frac{S}{(1-\gamma)^2 N}\right).$$

The claimed lower bound in Theorem 3.7 arises.

In the end, when $C^\star \in (1, 2)$, we know from the bounds (3.61) and (3.63) that

$$\begin{aligned}\inf_{\hat{\pi}} \sup_{P} \mathbb{E}[\mathcal{L}(\hat{\pi}; P)] &\gtrsim \max\left\{\min\left(\frac{1}{1-\gamma}, \frac{S}{(1-\gamma)^2 N}\right), \min\left(\frac{C^\star - 1}{1-\gamma}, \sqrt{\frac{S(C^\star-1)}{(1-\gamma)^3 N}}\right)\right\} \\ &\asymp \min\left(\frac{1}{1-\gamma}, \frac{S}{(1-\gamma)^2 N} + \sqrt{\frac{S(C^\star-1)}{(1-\gamma)^3 N}}\right),\end{aligned}$$

which completes the proof.

Proof of Lemma 3.7

By definition, one has

$$\begin{aligned}\mathcal{L}(\pi; P_1) + \mathcal{L}(\pi; P_2) &= J_{P_1}(\pi^\star) - J_{P_1}(\pi) + J_{P_2}(\pi^\star) - J_{P_2}(\pi) \\ &= \sum_{j=1}^{S/4} \rho_0(s_0^j) \left(V_{P_1}^\star(s_0^j) - V_{P_1}^\pi(s_0^j) + V_{P_2}^\star(s_0^j) - V_{P_2}^\pi(s_0^j)\right),\end{aligned}$$

where we have ignored the state s_{-1} since it has zero rewards. Our proof consists of three steps. We first connect the value difference $V_{P_1}^\star(s_0^j) - V_{P_1}^\pi(s_0^j)$ at s_0^j to that $V_{P_1}^\star(s_1^j) - V_{P_1}^\pi(s_1^j)$ at s_1^j. Then, we further link the value difference at s_1^j to the difference in transition probabilities, i.e., δ in our design. In the end, we use the property of the set $\mathcal{V}$ to conclude the lower bound.

Step 1. Since at state s_0^j, we only have one action a_1 with $r(s_0^j, a_1) = 0$, from the definition of value function one has

$$V_{P_1}^{\pi}(s_0^j) = \sum_{i=0}^{\infty} \gamma^{i+1}(1-p)p^i V_{P_1}^{\pi}(s_1^j),$$

for any policy π. Thus we have

$$V_{P_1}^{\star}(s_0^j) - V_{P_1}^{\pi}(s_0^j) = \sum_{i=0}^{\infty} \gamma^{i+1}(1-p)p^i \left(V_{P_1}^{\star}(s_1^j) - V_{P_1}^{\pi}(s_1^j)\right) > \frac{1}{4}\left(V_{P_1}^{\star}(s_1^j) - V_{P_1}^{\pi}(s_1^j)\right),$$

where we have used the fact that (assuming $\gamma \geq 1/2$)

$$\sum_{i=0}^{\infty} \gamma^{i+1}(1-p)p^i = \frac{1}{2}\gamma \geq \frac{1}{4}.$$

The same conclusion holds for P_2. Therefore we can obtain the following lower bound

$$\mathcal{L}(\pi; P_1) + \mathcal{L}(\pi; P_2) \geq \frac{1}{S}\sum_{j=1}^{S/4}\left(V_{P_1}^{\star}(s_1^j) - V_{P_1}^{\pi}(s_1^j) + V_{P_2}^{\star}(s_1^j) - V_{P_2}^{\pi}(s_1^j)\right).$$

Step 2. Without loss of generality, we assume that under P_1, $\mathbb{P}(s_{\oplus}^j \mid s_1^j, a_2) = \frac{1}{2} + \delta$, i.e., $v_j = +1$. Clearly, in this case, a_2 is the optimal action at s_1^j. If the policy π chooses the sub-optimal action (i.e., a_1) at s_1^j, then we have

$$\begin{aligned} V_{P_1}^{\star}(s_1^j) - V_{P_1}^{\pi}(s_1^j) &= \gamma\left(\left(\frac{1}{2}+\delta\right)V_{P_1}^{\star}\left(s_{\oplus}^j\right) + \left(\frac{1}{2}-\delta\right)V_{P_1}^{\star}\left(s_{\ominus}^j\right) - \frac{1}{2}V_{P_1}^{\pi}\left(s_{\oplus}^j\right) - \frac{1}{2}V_{P_1}^{\pi}\left(s_{\ominus}^j\right)\right) \\ &\geq \gamma\delta\left(V_{P_1}^{\star}\left(s_{\oplus}^j\right) - V_{P_1}^{\star}\left(s_{\ominus}^j\right)\right) \\ &\geq \gamma\delta\sum_{i=0}^{\infty}\gamma^i q^i = \frac{\gamma\delta}{1-\gamma q} = \frac{\gamma\delta}{2(1-\gamma)}. \end{aligned}$$

On the other hand, if $\pi(s_1^j)$ is not the optimal action (a_1 in this case), we have the trivial lower bound $V_{P_1}^{\star}(s_1^j) - V_{P_1}^{\pi}(s_1^j) \geq 0$. As a result, we obtain

$$V_{P_1}^{\star}(s_1^j) - V_{P_1}^{\pi}(s_1^j) \geq \frac{\gamma\delta}{2(1-\gamma)} 1\left\{\pi(s_1^j) \neq \pi_{P_1}^{\star}(s_1^j)\right\},$$

which implies

$$\begin{aligned} \mathcal{L}(\pi; P_1) + \mathcal{L}(\pi; P_2) &\geq \frac{1}{S}\cdot\frac{\gamma\delta}{2(1-\gamma)}\sum_{j=1}^{S/4}\left(1\left\{\pi(s_1^j) \neq \pi_{P_1}^{\star}(s_1^j)\right\} + 1\left\{\pi(s_1^j) \neq \pi_{P_2}^{\star}(s_1^j)\right\}\right) \\ &\geq \frac{1}{S}\cdot\frac{\gamma\delta}{2(1-\gamma)}\sum_{j=1}^{S/4} 1\left\{\pi_{P_1}^{\star}(s_1^j) \neq \pi_{P_2}^{\star}(s_1^j)\right\}. \end{aligned}$$

Step 3. In the end, we use the second property of the set $\mathcal{V}$, namely for any $\boldsymbol{v}_i \neq \boldsymbol{v}_j$ in $\mathcal{V}$, one has $\|\boldsymbol{v}_i - \boldsymbol{v}_j\|_1 \geq S/8$. An immediate consequence is that

$$\sum_{j=1}^{S/4} 1\left\{\pi^\star_{P_1}(s_1^j) \neq \pi^\star_{P_2}(s_1^j)\right\} = \|\boldsymbol{v}_{P_1} - \boldsymbol{v}_{P_2}\|_1 \geq \frac{S}{8}.$$

Taking the previous three steps collectively completes the proof.

Proof of Lemma 3.6

In the case of $C^\star = 1$, we have $d^\star = \mu$ which is the imitation learning setting. We adapt the information-theoretic lower bound for the episodic MDPs given in the work Rajaraman et al. (2020, Theorem 6) to the discounted setting.

Notations and Setup: Let $\mathcal{S}(\mathcal{D})$ be the set of all states that are observed in dataset $\mathcal{D}$. When $C^\star = 1$, we know the optimal policy $\pi^\star(s)$ at all states $s \in \mathcal{S}(\mathcal{D})$ visited in the dataset $\mathcal{D}$. We define $\Pi_{\text{mimic}}(\mathcal{D})$ as the family of deterministic policies which always take the optimal action on each state visited in $\mathcal{D}$, namely,

$$\Pi_{\text{mimic}}(\mathcal{D}) := \Big\{\forall s \in \mathcal{S}(\mathcal{D}),\ \pi(s) = \pi^\star(s)\Big\}, \tag{3.64}$$

Informally, $\Pi_{\text{mimic}}(\mathcal{D})$ is the family of policies which are "compatible" with the dataset collected by the learner.

Define $\mathbb{M}_{\mathcal{S},\mathcal{A}}$ as the family of MDPs over state space $\mathcal{S}$ and action space $\mathcal{A}$. We proceed by by lower bounding the Bayes expected suboptimality. That is, we aim at finding a distribution $\mathcal{P}$ over MDPs supported on $\mathbb{M}_{\mathcal{S},\mathcal{A}}$ such that,

$$\mathbb{E}_{\mathsf{MDP}\sim\mathcal{P}}\Big[J(\pi^\star) - \mathbb{E}_{\mathcal{D}}\left[J(\hat{\pi})\right]\Big] \gtrsim \min\left\{\frac{1}{1-\gamma}, \frac{S}{(1-\gamma)^2 N}\right\},$$

where $\hat{\pi}$ is a function of dataset $\mathcal{D}$.

Construction of the distribution $\mathcal{P}$: We first determine the distribution of the optimal policy, and then we design $\mathcal{P}$ such that conditioned on the optimal policy, the distribution is deterministic. We let the distribution of the optimal policy be uniform over all deterministic policies. That is, for each $s \in \mathcal{S}$, $\pi^\star(s) \sim \text{Unif}(\mathcal{A})$. For every $\pi^\star$, we construct an MDP instance in in Figure 3.6. Hence the distribution over MDPs comes from the randomness in π.

For a fixed optimal policy $\pi^\star$, the MDP instance $\mathsf{MDP}[\pi^\star]$ is determined as follows: we initialize with a fixed initial distribution over states $\rho = \{\zeta, \cdots, \zeta, 1-(S-2)\zeta, 0\}$ where $\zeta = \frac{1}{N+1}$. Let the last state be a special state b which we refer to as the "bad state". At each state $s \in \mathcal{S}\backslash\{b\}$, choosing the optimal action renews the state in the initial distribution ρ and

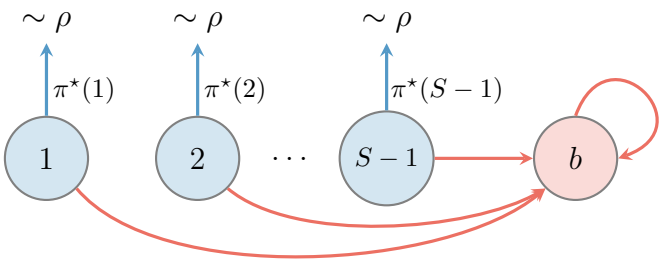

Figure 3.6: The hard MDP instance for the case $C^\star = 1$. Upon playing the optimal (blue) action at any state except b, the learner returns to a new state according to initial distribution $\rho = \{\zeta, \cdots, \zeta, 1-(S-2)\zeta, 0\}$ where $\zeta = \frac{1}{N+1}$. Any other choice of action (red) deterministically transitions the state to b.

gives a reward of 1, while any other choice of action deterministically induces a transition to the bad state b and offers zero reward. In addition, the bad state is absorbing and dispenses no reward regardless of the choice of action. That is,

$$P(\cdot \mid s, a) = \begin{cases} \rho, & s \in \mathcal{S} \setminus \{b\},\ a = \pi^\star(s) \\ \delta_b, & \text{otherwise,} \end{cases} \tag{3.65}$$

and the reward function of the MDP is given by

$$r(s, a) = \begin{cases} 1, & s \in \mathcal{S} \setminus \{b\},\ a = \pi^\star(s), \\ 0, & \text{otherwise.} \end{cases} \tag{3.66}$$

Under this construction, it is easy to see that $J_{\mathsf{MDP}}(\pi^\star(\mathsf{MDP})) = 1/(1-\gamma)$ since the optimal action always acquires reward 1 throughout the trajectory. Thus the Bayes risk can be written as

$$\mathbb{E}_{\mathsf{MDP}\sim\mathcal{P}}\left[\frac{1}{1-\gamma} - \mathbb{E}\left[J_{\mathsf{MDP}}(\widehat{\pi}(\mathcal{D}))\right]\right]. \tag{3.67}$$

Understanding the conditional distribution. Now we study the conditional distribution of the MDP given the observed dataset $\mathcal{D}$. We start from the conditional distribution of the optimal policy. We present the following lemma without proof.

Lemma 3.8 (Rajaraman et al. (2020, Lemma A.14))**.** *Conditioned on the dataset $\mathcal{D}$ collected by the learner, the optimal policy $\pi^\star$ is distributed $\sim \mathrm{Unif}(\Pi_{\mathrm{mimic}}(\mathcal{D}))$. In other words, at each state visited in the dataset, the optimal action is fixed. At the remaining states, the optimal action is sampled uniformly from $\mathcal{A}$.*

Now we define the conditional distribution of the MDPs given the dataset $\mathcal{D}$ collected by the learner as below.

Definition 3.2. *Define* $\mathcal{P}(\mathcal{D})$ *as the distribution of* MDP *conditioned on the observed dataset* $\mathcal{D}$*. In particular,* $\pi^\star \sim \mathrm{Unif}(\Pi_{\mathrm{mimic}}(\mathcal{D}))$ *and* $\mathsf{MDP} = \mathsf{MDP}[\pi^\star]$.

From Lemma 3.8 and the definition of $\mathcal{P}(\mathcal{D})$ in Definition 3.2, applying Fubini's theorem gives

$$\mathbb{E}_{\mathsf{MDP}\sim\mathcal{P}}\left[\frac{1}{1-\gamma} - \mathbb{E}_{\mathcal{D}}\left[J(\widehat{\pi})\right]\right] = \mathbb{E}_{\mathcal{D}}\left[\mathbb{E}_{\mathsf{MDP}\sim\mathcal{P}}\left[\frac{1}{1-\gamma} - J(\widehat{\pi})\right]\right]. \tag{3.68}$$

Lower bounding the Bayes Risk. Next we relate the Bayes risk to the first time the learner visits a state unobserved in $\mathcal{D}$.

Lemma 3.9. *In the trajectory induced by the infinite-horizon MDP and policy, define the stopping time* τ *as the first time that the learner encounters a state* $s \neq b$ *that has not been visited in* $\mathcal{D}$ *at time* t*. That is,*

$$\tau = \begin{cases} \inf\{t : s_t \notin \mathcal{S}(\mathcal{D}) \cup \{b\}\} & \exists t : s_t \notin \mathcal{S}(\mathcal{D}) \cup \{b\} \\ +\infty & \textit{otherwise.} \end{cases} \tag{3.69}$$

Then, conditioned on the dataset $\mathcal{D}$ *collected by the learner,*

$$\mathbb{E}_{\mathsf{MDP}\sim\mathcal{P}(\mathcal{D})}\left[J(\pi^\star) - \mathbb{E}\left[J(\widehat{\pi})\right]\right] \geq \left(1 - \frac{1}{|\mathcal{A}|}\right)\mathbb{E}_{\mathsf{MDP}\sim\mathcal{P}(\mathcal{D})}\left[\mathbb{E}_{\widehat{\pi}(\mathcal{D})}\left[\frac{\gamma^\tau}{1-\gamma}\right]\right] \tag{3.70}$$

We defer the proof to the end of this section.

Plugging the result of Lemma 3.9 into equality (3.68), we obtain

$$\begin{aligned}
\mathbb{E}_{\mathsf{MDP}\sim\mathcal{P}}\left[J(\pi^\star) - \mathbb{E}\left[J(\widehat{\pi})\right]\right] &\geq \left(1 - \frac{1}{|\mathcal{A}|}\right)\mathbb{E}_{\mathcal{D}}\left[\mathbb{E}_{\mathsf{MDP}\sim\mathcal{P}(\mathcal{D})}\left[\mathbb{E}_{\widehat{\pi}(\mathcal{D})}\left[\frac{\gamma^\tau}{1-\gamma}\right]\right]\right], \\
&\overset{(i)}{\geq} \left(1 - \frac{1}{|\mathcal{A}|}\right)\frac{1}{2(1-\gamma)}\mathbb{E}_{\mathcal{D}}\left[\mathbb{E}_{\mathsf{MDP}\sim\mathcal{P}(\mathcal{D})}\left[\Pr_{\widehat{\pi}(\mathcal{D})}\left[\tau \leq \lfloor\frac{1}{\log(1/\gamma)}\rfloor\right]\right]\right], \\
&= \left(1 - \frac{1}{|\mathcal{A}|}\right)\frac{1}{2(1-\gamma)}\mathbb{E}_{\mathsf{MDP}\sim\mathcal{P}}\left[\mathbb{E}_{\mathcal{D}}\left[\Pr_{\widehat{\pi}(\mathcal{D})}\left[\tau \leq \lfloor\frac{1}{\log(1/\gamma)}\rfloor\right]\right]\right],
\end{aligned}$$

where (i) uses Markov's inequality. Lastly we bound the probability that we visit a state unobserved in the dataset before time $\lfloor\frac{1}{\log(1/\gamma)}\rfloor$. For any policy $\widehat{\pi}$, from a similar proof as Rajaraman et al. (2020, Lemma A.16) we have

$$\mathbb{E}_{\mathsf{MDP}\sim\mathcal{P}}\left[\mathbb{E}_{\mathcal{D}}\left[\Pr_{\widehat{\pi}}\left[\tau \leq \lfloor\frac{1}{\log(1/\gamma)}\rfloor\right]\right]\right] \gtrsim \min\left\{1, \frac{S}{\log(1/\gamma)N}\right\}. \tag{3.71}$$

Therefore,

$$\begin{aligned}
\mathbb{E}_{\mathsf{MDP}\sim\mathcal{P}}\left[J(\pi^\star) - \mathbb{E}\left[J(\widehat{\pi})\right]\right] &\gtrsim \left(1 - \frac{1}{|\mathcal{A}|}\right)\frac{1}{\log(1/\gamma)}\min\left\{1, \frac{S}{(1-\gamma)N}\right\} \\
&\geq \left(1 - \frac{1}{|\mathcal{A}|}\right)\frac{\gamma}{1-\gamma}\min\left\{1, \frac{S}{(1-\gamma)N}\right\}
\end{aligned}$$

Here we use the fact that $\log(x) \leq x - 1$. Since $1 - \frac{1}{|\mathcal{A}|} \geq 1/2$ for $|\mathcal{A}| \geq 2$, the final result follows.

Proof of Lemma 3.9. To facilitate the analysis, we define an auxiliary random variable τ_b to be the first time the learner encounters the state b. If no such state is encountered, τ_b is defined as $+\infty$. Formally,

$$\tau_b = \begin{cases} \inf\{t : s_t = b\}, & \exists t : s_t = b, \\ +\infty, & \text{otherwise.} \end{cases}$$

Conditioned on the observed dataset $\mathcal{D}$, we have

$$\frac{1}{1-\gamma} - \mathbb{E}_{\mathsf{MDP} \sim \mathcal{P}(\mathcal{D})}\left[J(\widehat{\pi})\right] = \frac{1}{1-\gamma} - \mathbb{E}_{\mathsf{MDP} \sim \mathcal{P}(\mathcal{D})}\left[\mathbb{E}_{\widehat{\pi}}\left[\sum\nolimits_{t=0}^{\infty} \gamma^t r(s_t, a_t)\right]\right] \tag{3.72}$$

$$\geq \mathbb{E}_{\mathsf{MDP} \sim \mathcal{P}(\mathcal{D})}\left[\mathbb{E}_{\widehat{\pi}}\left[\frac{\gamma^{\tau_b - 1}}{1-\gamma}\right]\right] \tag{3.73}$$

where the last inequality follows from the fact that r is bounded in $[0, 1]$, and the state b is absorbing and always offers 0 reward. Fixing the dataset $\mathcal{D}$ and the optimal policy $\pi^\star$ (which determines the MDP $\mathsf{MDP}[\pi^\star]$), we study $\mathbb{E}_{\widehat{\pi}(\mathcal{D})}\left[\frac{\gamma^{\tau_b - 1}}{1-\gamma}\right]$ and try to relate it to $\mathbb{E}_{\widehat{\pi}(\mathcal{D})}\left[\frac{\gamma^{\tau}}{1-\gamma}\right]$. Note that for any t and state $s \in \mathcal{S}$,

$$\begin{aligned} \Pr_{\widehat{\pi}}\left[\tau_b = t+1, \tau = t, s_t = s\right] &= \Pr_{\widehat{\pi}}\left[\tau_b = t+1 \mid \tau = t, s_t = s\right] \Pr_{\widehat{\pi}}\left[\tau = t, s_t = s\right] \\ &= \Big(1 - \mathbb{1}\{\widehat{\pi}(s) = \pi^\star(s)\}\Big) \Pr_{\widehat{\pi}}\left[\tau = t, s_t = s\right]. \end{aligned}$$

In the last equation, we use the fact that the learner must play an action other than $\pi^\star(s_t)$ to visit b at time $t+1$. Next we take an expectation with respect to the randomness of $\pi^\star$ which conditioned on $\mathcal{D}$ is drawn from $\mathrm{Unif}(\Pi_{\mathrm{mimic}}(\mathcal{D}))$. Note that $\mathsf{MDP}[\pi^\star]$ is also determined conditioning on $\pi^\star$. Observe that the dependence of the second term $\Pr_{\widehat{\pi}}\left[\tau = t, s_t = s\right]$ on $\pi^\star$ comes from the probability computed with the underlying MDP chosen as $\mathsf{MDP}[\pi^\star]$. However it only depends on the characteristics of $\mathsf{MDP}[\pi^\star]$ on the observed states in $\mathcal{D}$. On the other hand, the first term $(1 - \mathbb{1}\{\widehat{\pi}(s) = \pi^\star(s)\})$ depends only on $\pi^\star(s)$, where s is an unobserved state. Thus the two terms are independent. By taking expectation with respect to the randomness of $\pi^\star \sim \mathrm{Unif}(\Pi_{\mathrm{mimic}}(\mathcal{D}))$ and $\mathsf{MDP} = \mathsf{MDP}[\pi^\star]$, we have

$$\begin{aligned} &\mathbb{E}_{\mathsf{MDP} \sim \mathcal{P}(\mathcal{D})}\Big[\Pr_{\widehat{\pi}(\mathcal{D})}\left[\tau_b = t+1, \tau = t, s_t = s\right]\Big] \\ &= \mathbb{E}_{\mathsf{MDP} \sim \mathcal{P}(\mathcal{D})}\Big[1 - \mathbb{1}\{\widehat{\pi}(s) = \pi^\star(s)\}\Big]\ \mathbb{E}_{\mathsf{MDP} \sim \mathcal{P}(\mathcal{D})}\Big[\Pr_{\widehat{\pi}}\left[\tau = t, s_t = s\right]\Big] \\ &= \left(1 - \frac{1}{|\mathcal{A}|}\right) \mathbb{E}_{\mathsf{MDP} \sim \mathcal{P}(\mathcal{D})}\Big[\Pr_{\widehat{\pi}}\left[\tau = t, s_t = s\right]\Big] \end{aligned}$$

where in the last equation, we use the fact that conditioned on $\mathcal{D}$ either (i) $s = b$, in which case $\tau \neq t$ and both sides are 0, or (ii) if $s \neq b$, then $\tau = t$ implies that the state s visited at time t must not be observed in $\mathcal{D}$, so $\pi^\star(s) \sim \text{Unif}(\mathcal{A})$. Using the fact that $\Pr_{\widehat{\pi}}[\tau_b = t+1, \tau = t, s_t = s] \leq \Pr_{\widehat{\pi}}[\tau_b = t+1, s_t = s]$ and summing over $s \in \mathcal{S}$ results in the inequality,

$$\mathbb{E}_{\mathsf{MDP}\sim\mathcal{P}(\mathcal{D})}\Big[\Pr_{\widehat{\pi}}[\tau_b = t+1]\Big] \geq \left(1 - \frac{1}{|\mathcal{A}|}\right) \mathbb{E}_{\mathsf{MDP}\sim\mathcal{P}(\mathcal{D})}\Big[\Pr_{\widehat{\pi}}[\tau = t]\Big].$$

Multiplying both sides by $\frac{\gamma^t}{1-\gamma}$ and summing over $t = 1, \cdots, \infty$,

$$\mathbb{E}_{\mathsf{MDP}\sim\mathcal{P}(\mathcal{D})}\left[\mathbb{E}_{\widehat{\pi}}\left[\frac{\gamma^{\tau_b - 1}}{1-\gamma}\right]\right] \geq \left(1 - \frac{1}{|\mathcal{A}|}\right) \mathbb{E}_{\mathsf{MDP}\sim\mathcal{P}(\mathcal{D})}\left[\mathbb{E}_{\widehat{\pi}}\left[\frac{\gamma^{\tau}}{1-\gamma}\right]\right].$$

here we use the fact that the initial distribution ρ places no mass on the bad state b. Therefore, $\Pr_{\widehat{\pi}(D)}[\tau_b = 1] = \rho(b) = 0$. This equation in conjunction with (3.73) completes the proof.

3.9.7 Imitation learning in discounted MDPs

In Theorem 3.3, we have shown that imitation learning has a worse rate than LCB even in the contextual bandit case when $C^\star \in (1, 2)$. In this section, we show that if we change the concentrability assumption from density ratio to conditional density ratio, behavior cloning continues to work in certain regime. This also shows that behavior cloning works when $C^\star = 1$ in the discounted MDP case.

Theorem 3.8. *Assume the expert policy $\pi^\star$ is deterministic and that $\max \frac{(1-\gamma)d^*(a|s)}{\mu(a|s)} \leq C^\star$ for some $C^\star \in [1, 2)$. We consider a variant of behavior cloning policy:*

$$\Pi_{mimic} = \{\pi \in \Pi_{det} : \forall s \in \mathcal{D}, \pi(\cdot \mid s) = \arg\max_a N(s, a)\}. \tag{3.74}$$

Here $\pi \in \Pi_{det}$ refers to the set of all deterministic policies. Then for any $\hat{\pi} \in \Pi_{mimic}$, we have

$$\mathbb{E}_{\mathcal{D}}[J(\pi^*) - J(\hat{\pi})] \lesssim \frac{S}{C_0 N(1-\gamma)^2},$$

where $C_0 = 1 - \exp\left(-\mathsf{KL}\left(\frac{1}{2} \| \frac{1}{C^\star}\right)\right)$.

Proof. Define the following population loss:

$$\mathcal{L}(\hat{\pi}, \pi^*) = \mathbb{E}_{\mathcal{D}}[\mathbb{E}_{s\sim d_\star}[1\{\hat{\pi}(s) \neq \pi^*(s)\}]]. \tag{3.75}$$

From Lemma 3.5, we know that it suffices to control the population loss $\mathcal{L}(\hat{\pi}, \pi^\star)$. From a similar argument as in Rajaraman et al. (2020), we know that when $C^\star = 1$, the expected suboptimality of $\hat{\pi}$ is upper bounded by $\min(\frac{1}{1-\gamma}, \frac{S}{(1-\gamma)^2 N})$.

When $C^\star \in (1,2)$, the contribution to the indicator loss can be decomposed into two parts: (1) the loss incurred due to the states not included in $\mathcal{D}$ whose expected value is upper bounded by S/N; (2) the loss incurred due to states the states for which the optimal action is not the most frequent in $\mathcal{D}$. Conditioned on $N(s)$ and from $\mu(\pi^\star(s)|s) \geq d^\star(\pi^\star(s)|s)/C^\star = 1/C^\star$ the probability of not picking the optimal action is upper bounded by $\exp(-N(s) \cdot \mathsf{KL}\left(\text{Bern}\left(\frac{1}{2}\right) \| \text{Bern}\left(\frac{1}{C^\star}\right)\right))$ using Chernoff's inequality. We have

$$
\begin{aligned}
&\mathbb{E}[\mathcal{L}(\hat{\pi}, \pi^\star)] && (3.76)\\
&= \mathbb{E}_{s\sim d_\star, \mathcal{D}}[1\{\hat{\pi}(s) \neq \pi^\star(s)\}] \\
&\leq \mathbb{E}_{s\sim d_\star, \mathcal{D}}[\mathbb{P}(N(s) = 0)] + \mathbb{E}_{s\sim d_\star}\, \mathbb{E}_{\mathcal{D}}[\mathbb{P}(\hat{\pi}(s) \neq \pi^\star(s)) \mid N(s) \geq 1] \\
&\lesssim \frac{S}{N} + \mathbb{E}_{s\sim d_\star}\, \mathbb{E}_{\mathcal{D}}\left[\exp\left(-N(s) \cdot \mathsf{KL}\left(\text{Bern}\left(\frac{1}{2}\right) \| \text{Bern}\left(\frac{1}{C^\star}\right)\right)\right) \mid N(s) \geq 1\right] \\
&\lesssim \frac{S}{N} + \sum_s p(s) \sum_{n=1}^{N} \binom{N}{n} \exp\left(-n \cdot \mathsf{KL}\left(\text{Bern}\left(\frac{1}{2}\right) \| \text{Bern}\left(\frac{1}{C^\star}\right)\right)\right) p(s)^n (1-p(s))^{N-n} \\
&\leq \frac{S}{N} + \sum_s p(s) \left(1 - p(s)\left(1 - \exp\left(-\mathsf{KL}\left(\text{Bern}\left(\frac{1}{2}\right) \| \text{Bern}\left(\frac{1}{C^\star}\right)\right)\right)\right)\right)^N. && (3.77)
\end{aligned}
$$

Denote $C_0 = 1 - \exp\left(-\mathsf{KL}\left(\text{Bern}\left(\frac{1}{2}\right) \| \text{Bern}\left(\frac{1}{C^\star}\right)\right)\right)$. Note that $\max_{x\in[0,1]} x(1-C_0 x)^N \leq \frac{1}{C_0(N+1)}(1-\frac{1}{N+1})^N \leq \frac{4}{9C_0 N}$. Thus we have $\mathbb{E}[\mathcal{L}(\hat{\pi}, \pi^\star)] \leq \frac{4S}{9C_0 N}$. We then use Lemma 3.5 to conclude that the final sub-optimality is upper bounded by $\frac{S}{C_0 N(1-\gamma)^2}$. □

3.10 LCB in episodic Markov decision processes

The aim of this section is to illustrate the validity of Conjecture 1 in episodic MDPs. In Section 3.10.1, we give a brief review of episodic MDPs, describing the batch dataset and offline RL objective in this setting, and introducing additional notation. We then present a variant of the VI-LCB algorithm (Algorithm 5) for episodic MDPs and state its sub-optimality guarantees in Section 3.10.2. In Section 3.10.3, we show that the proposed penalty captures a confidence interval and prove a value difference lemma for Algorithm 5. Section 3.10.4 is devoted to the proof of the sub-optimality upper bound. In Section 3.10.5, we give an alternative sub-optimality decomposition as an attempt to obtain a tight dependency on $C^\star$ in the regime $C^\star \in [1,2)$. We analyze the sub-optimality in this regime in a special example provided in Section 3.10.6.

3.10.1 Model and notation

Episodic MDP. We consider an episodic MDP described by a tuple $(\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \rho, H)$, where $\mathcal{S} = \{\mathcal{S}_h\}_{h=1}^H$ is the state space, $\mathcal{A}$ is the action space, $\mathcal{P} = \{P_h\}_{h=1}^H$ is the set of transition kernels with $P_h : \mathcal{S}_h \times \mathcal{A} \mapsto \Delta(\mathcal{S}_{h+1})$, $\mathcal{R} = \{R_h\}_{h=1}^H$ is the set of reward

distributions $R_h : \mathcal{S}_h \times \mathcal{A} \to \Delta([0,1])$ with $r : \mathcal{S} \times \mathcal{A} \mapsto [0,1]$ as the expected reward function, $\rho : \mathcal{S}_1 \to \Delta(\mathcal{S}_1)$ is the initial distribution, and H is the horizon. To streamline our analysis, we assume that $\{\mathcal{S}_h\}_{h=1}^H$ partition the state space $\mathcal{S}$ and are disjoint.

Policy and value functions. Similar to the discounted case, we consider deterministic policies $\pi : \mathcal{S} \mapsto \mathcal{A}$ that map each state to an action. For any $h \in \{1, \ldots, H\}$, $s \in \mathcal{S}_h$, and $a \in \mathcal{A}_h$, the value function $V_h^\pi : \mathcal{S} \mapsto \mathbb{R}$ and Q-function $Q_h^\pi : \mathcal{S} \times \mathcal{A} \mapsto \mathbb{R}$ are respectively defined as

$$V_h^\pi(s) \coloneqq \mathbb{E}\left[\sum_{i=h}^H r_i \middle| s_h = s, a_i = \pi(s_i) \text{ for } i \geq h\right],$$
$$Q_h^\pi(s,a) \coloneqq \mathbb{E}\left[\sum_{i=h}^H r_i \middle| s_h = s, a_h = a, a_i = \pi(s_i) \text{ for } i \geq h+1\right].$$

Since we assume that the set of state in different levels are disjoint, we drop the subscript h when it is it clear from the context. The expected value of a policy π is defined analogously to the discounted case:

$$J(\pi) \coloneqq \mathbb{E}_{s\sim\rho}[V_1^\pi(s)].$$

It is well-known that a deterministic policy $\pi^\star$ exists that maximizes the value function from any state.

Episodic occupancy measures. We define the (normalized) state occupancy measure $d_\pi : \mathcal{S} \mapsto [0, H]$ and state-action occupancy measure $d^\pi : \mathcal{S} \times \mathcal{A} \mapsto [0, H]$ as

$$d_\pi(s) \coloneqq \frac{1}{H}\sum_{h=1}^H \mathbb{P}_h(s_h = s; \pi), \quad \text{and} \quad d^\pi(s,a) \coloneqq \frac{1}{H}\sum_{h=1}^H \mathbb{P}_h(s_h = s, a_h = a; \pi), \tag{3.78}$$

where we overload notation and write $\mathbb{P}_h(s_h = s; \pi)$ to denote the probability of visiting state $s_h = s$ (and similarly $s_h = s, a_h = a$) at level h after executing policy π and starting from $s_1 \sim \rho(\cdot)$.

Batch dataset. The batch dataset $\mathcal{D}$ consists of tuples (s, a, r, s'), where $r = r(s,a)$ and $s' \sim P(\cdot \mid s, a)$. As in the discounted case, we assume that (s,a) pairs are generated i.i.d. according to a data distribution μ, unknown to the agent. We denote by $N(s,a) \geq 0$ the number of times a pair (s,a) is observed in $\mathcal{D}$ and by $N = |\mathcal{D}|$ the total number of samples.

The learning objective. Fix a deterministic policy π. The expected sub-optimality of policy $\hat{\pi}$ computed based on dataset $\mathcal{D}$ competing with policy π is defined as

$$\mathbb{E}_{\mathcal{D}}\left[J(\pi) - J(\hat{\pi})\right]. \tag{3.79}$$

Algorithm 5 Episodic value iteration with LCB

Inputs: Batch dataset $\mathcal{D}$.
$\hat{V}_{H+1} \leftarrow\!\shortmid 0$.
for $h = H-1, \ldots, 1$ **do**
  for $s \in \mathcal{S}_h, a \in \mathcal{A}$ **do**
    if $N(s,a) = 0$ **then**
      Set $r(s,a) = 0$.
      Set the empirical transition vector $\hat{P}_{s,a}$ randomly.
      Set the penalty $b(s,a) = H\sqrt{L}$.
    else
      Set $r(s,a)$ according to dataset.
      Compute the empirical transition vector $\hat{P}_{s,a}$ according to dataset.
      Set the penalty $b(s,a) = H\sqrt{L/N(s,a)}$, where $L = 2000\log(2S|\mathcal{A}|/\delta)$.
    Compute $\hat{Q}_h(s,a) \leftarrow\!\shortmid r(s,a) - b(s,a) + \hat{P}_{s,a} \cdot \hat{V}_{h+1}$.
    Compute $\hat{V}_h(s) \leftarrow\!\shortmid \max_a \hat{Q}_h(s,a)$ and $\hat{\pi}(s) \in \arg\max_a \hat{Q}_h(s,a)$.
Return: $\hat{\pi}$.

Assumption on dataset coverage. Equipped with the definitions for occupancy densities in episodic MDPs, we define the concentrability coefficient in the episodic case analogously: given a deterministic policy π, C^π is the smallest constant satisfying

$$\frac{d^\pi(s,a)}{\mu(s,a)} \leq C^\pi \qquad \forall s \in \mathcal{S}, a \in \mathcal{A}. \tag{3.80}$$

Matrix notation. We adopt a matrix notation similar to the one described in Section 3.4.

Bellman equations. Given any value function $V : \mathcal{S}_{h+1} \mapsto \mathbb{R}$, the Bellman value operator at each level $h \in \{1, \ldots, H\}$

$$\mathcal{T}_h V = r_h + P_h V. \tag{3.81}$$

We write $(\mathcal{T}_h V)(s,a) = r_h(s,a) + (P_h V)(s,a)$ for $\mathcal{S} \in \mathcal{S}_h, a \in \mathcal{A}$.

3.10.2 Episodic value iteration with LCB

Algorithm 5 presents a pseudocode for value iteration with LCB in the episodic setting. As in the classic value iteration in episodic MDPs, this algorithm computes values and policy through a backward recursion starting at $h = H$ with the distinction of subtracting penalties when computing the Q-function. This algorithm can be viewed as an instance of Algorithm 5 of Jin et al. (2020c).

In the following theorem, we provide an upper bound on the expected sub-optimality of the policy returned by Algorithm 5. The proof is presented in Section 3.10.4.

Theorem 3.9 (LCB sub-optimality, episodic MDP)**.** *Consider an episodic MDP and assume that*

$$\frac{d^{\pi}(s,a)}{\mu(s,a)} \leq C^{\pi} \qquad \forall s \in \mathcal{S}, a \in \mathcal{A}$$

holds for an arbitrary deterministic policy π. Set $\delta = 1/N$ in Algorithm 5. Then, for all $C^{\pi} \geq 1$, one has

$$\mathbb{E}_{\mathcal{D}}[J(\pi) - J(\hat{\pi})] \lesssim \min\left\{H, \widetilde{O}\left(H^2\sqrt{\frac{SC^{\pi}}{N}}\right)\right\}.$$

In addition, if $1 \leq C^{\pi} \leq 1 + L/(200N)$, then we have a tighter performance guarantee

$$\mathbb{E}_{\mathcal{D}}[J(\pi) - J(\hat{\pi})] \lesssim \min\left\{H, \widetilde{O}\left(H^2\frac{S}{N}\right)\right\}.$$

We make the following conjecture that the sub-optimality rate smoothly transitions from $1/N$ to $1/\sqrt{N}$ as C^{π} increases from 1 to 2.

Conjecture 2. *Assume as in Theorem 3.9. If $1 \leq C^{\pi} \leq 2$, then policy $\hat{\pi}$ returned by Algorithm 5 obeys*

$$\mathbb{E}_{\mathcal{D}}[J(\pi) - J(\hat{\pi})] \lesssim \min\left\{H, \widetilde{O}\left(H^2\sqrt{\frac{S(C^{\pi}-1)}{N}}\right)\right\}.$$

We present our attempt in proving the above conjecture in part in Section 3.10.5 followed by an example in Section 3.10.6.

3.10.3 Properties of Algorithm 5

In this section, we prove two properties of Algorithm 5. We first prove that the penalty captures the Q-function lower confidence bound. Then, we prove a value difference lemma.

Clean event in episodic MDPs. Define the following clean event

$$\mathcal{E}_{\text{EMDP}} := \left\{\forall h, \forall s \in \mathcal{S}_h, \forall a : \left|r(s,a) + P_{s,a} \cdot \hat{V}_{h+1} - \hat{r}(s,a) - \hat{P}_{s,a} \cdot \hat{V}_{h+1}\right| \leq b_h(s,a)\right\}, \quad (3.82)$$

where $\hat{V}_{H+1} = 0$. In the following lemma, we show that the penalty used in Algorithm 5 captures the confidence interval of the empirical expectation of the Q-function.

Lemma 3.10 (Clean event probability, episodic MDP)**.** *One has $\mathbb{P}(\mathcal{E}_{EMDP}) \geq 1 - \delta$.*

Proof. The proof is analogous to the proof of Lemma 3.1. Fix a tuple (s, a, h). If $N(s, a) = 0$, it is immediate that

$$|r(s,a) + P_{s,a} \cdot \hat{V}_{h+1} - \hat{r}(s,a) - \hat{P}_{s,a} \cdot \hat{V}_{h+1}| \leq H\sqrt{L}.$$

When $N(s, a) \geq 1$, we exploit the independence of $\hat{V}_{h+1}$ and $\hat{P}_{s,a}$ (thanks to the disjoint state space at each step h) and conclude by Hoeffding's inequality that for any $\delta_1 \in (0, 1)$

$$\mathbb{P}\left(|r(s,a) + P_{s,a} \cdot \hat{V}_{h+1} - \hat{r}(s,a) + P_{s,a} \cdot \hat{V}_{h+1}| \geq H\sqrt{\frac{2\log(2/\delta_1)}{N(s,a)}}\right) \leq \delta_1.$$

The claim follows by taking a union bound over $s \in \mathcal{S}_h, a \in \mathcal{A}, h \in [H]$ and setting $\delta_1 = \delta/(S|\mathcal{A}|)$. □

Value difference lemma. The following lemma bounds the sub-optimality of Algorithm 5 by expected bonus. This result is similar to Theorem 4.2 in Jin et al. (2020c). We present the proof for completeness.

Lemma 3.11 (Value difference for Algorithm 5)**.** *Let π be an arbitrary policy. On the event $\mathcal{E}_{EMDP}$, the policy $\hat{\pi}$ returned by Algorithm 5 satisfies*

$$J(\pi) - J(\hat{\pi}) \leq 2H\,\mathbb{E}_{d^\pi}\left[b(s,a)\right].$$

Proof. Define the following self-consistency error

$$\iota_h(s,a) = \mathcal{T}_h\hat{V}_{h+1}(s,a) - \hat{Q}_h(s,a),$$

where $\mathcal{T}_h$ is the Bellman value operator defined in (3.81). Let π' be an arbitrary policy. By Jin et al. (2020c, Lemma A.1), one has

$$\begin{aligned}\hat{V}_1(s) - V_1^{\pi'}(s) = &\sum_{h=1}^{H} \mathbb{E}[\hat{Q}_h(s_h, \hat{\pi}(s_h)) - \hat{Q}_h(s_h, \pi'(s_h)) \mid s_1 = s] \\ &- \sum_{h=1}^{H} \mathbb{E}[\iota_h(s_h, \pi'(s_h)) \mid s_1 = s]\end{aligned} \tag{3.83}$$

Setting $\pi' \leftarrow\!\shortmid \pi$ in (3.83) gives

$$\begin{aligned}V_1^{\pi}(s) - \hat{V}_1(s) &= \sum_{h=1}^{H} \mathbb{E}[\iota_h(s_h, \pi(s_h)) \mid s_1 = s] - \sum_{h=1}^{H} \mathbb{E}[\hat{Q}_h(s_h, \hat{\pi}(s_h)) - \hat{Q}_h(s_h, \pi(s_h)) \mid s_1 = s] \\ &\leq \sum_{h=1}^{H} \mathbb{E}[\iota_h(s_h, \pi(s_h)) \mid s_1 = s],\end{aligned} \tag{3.84}$$

where the last line uses the fact that $\hat{\pi}(s)$ maximizes $\hat{Q}_h(s,a)$.

We apply (3.83) once more, this time setting $\pi' \leftarrow\!\shortmid \hat{\pi}$:

$$\begin{aligned}\hat{V}_1(s) - V_1^{\hat{\pi}}(s) &= \sum_{h=1}^{H} \mathbb{E}[\hat{Q}_h(s_h, \hat{\pi}(s_h)) - \hat{Q}_h(s_h, \hat{\pi}(s_h)) \mid s_1 = s] - \sum_{h=1}^{H} \mathbb{E}[\iota_h(s_h, \hat{\pi}(s_h)) \mid s_1 = s] \\ &\leq -\sum_{h=1}^{H} \mathbb{E}[\iota_h(s_h, \pi'(s_h)) \mid s_1 = s]. \end{aligned} \tag{3.85}$$

Adding (3.84) and (3.85), we have

$$\begin{aligned} V_1^{\pi}(s) - V_1^{\hat{\pi}}(s) &= V_1^{\pi}(s) - \hat{V}_1(s) + \hat{V}_1(s) - V_1^{\hat{\pi}}(s) \\ &\leq \sum_{h=1}^{H} \mathbb{E}[\iota_h(s_h, \pi(s_h)) \mid s_1 = s] - \sum_{h=1}^{H} \mathbb{E}[\iota_h(s_h, \pi'(s_h)) \mid s_1 = s]. \end{aligned} \tag{3.86}$$

By Jin et al. (2020c, Lemma 5.1), conditioned on $\mathcal{E}_{\text{EMDP}}$, we have

$$0 \leq \iota_h(s,a) \leq 2b_h(s,a) \quad \forall s, a, h.$$

The proof is completed by applying the above bound in (3.86) and taking an expectation with respect to ρ

$$\begin{aligned} \mathbb{E}_{\rho}[V_1^{\pi}(s) - V_1^{\hat{\pi}}(s)] &\leq 2\sum_{h=1}^{H} \mathbb{E}[b_h(s_h, \pi(s_h))] \\ &= 2\sum_{h=1}^{H} P_h(s_h; \pi) b_h(s_h, \pi(s_h)) = 2H\, \mathbb{E}_{d^{\pi}}[b(s,a)], \end{aligned}$$

where the last equation hinges on the definition of occupancy measure for episodic MDPs given in (3.78). □

3.10.4 Proof of Theorem 3.9

The proof follows a similar decomposition argument as in Theorem 3.6. Nonetheless, we present a complete proof for the reader's convenience.

We divide the proof into two parts and separately analyze the general case $C^{\pi} \geq 1$ and $C^{\star} \leq 1 + L/(200N)$ since the techniques used in the proof of these two claims are rather distinct.

The general case when $C^\pi \geq 1$. We decompose the expected sub-optimality into two terms

$$\begin{aligned}\mathbb{E}_{\mathcal{D}}\left[\sum_s \rho(s)[V_1^\pi(s) - V_1^{\hat{\pi}}(s)]\right] &= \mathbb{E}_{\mathcal{D}}\left[\sum_s \rho(s)[V_1^\pi(s) - V_1^{\hat{\pi}}(s)]\,\mathbb{1}\{\mathcal{E}_{\text{EMDP}}\}\right] =: T_1 \\ &\quad + \mathbb{E}_{\mathcal{D}}\left[\sum_s \rho(s)[V_1^\pi(s) - V_1^{\hat{\pi}}(s)]\,\mathbb{1}\{\mathcal{E}_{\text{EMDP}}^c\}\right] =: T_2.\end{aligned} \tag{3.87}$$

The first term T_1 captures the sub-optimality under the clean event $\mathcal{E}_{\text{EMDP}}$ whereas T_2 represents the sub-optimality suffered when the constructed confidence interval via the penalty function falls short of containing the empirical Q-function estimate. We will prove in subsequent sections that T_1 and T_2 are bounded according to:

$$T_1 \leq 32H^2\sqrt{\frac{SC^\pi L}{N}} \tag{3.88a}$$

$$T_2 \leq H\delta. \tag{3.88b}$$

Taking the above bounds as given for the moment and setting $\delta = 1/N$, we conclude that

$$\mathbb{E}_{\mathcal{D}}[J(\pi) - J(\hat{\pi})] \lesssim \min\left(H, 32H^2\sqrt{\frac{SC^\pi L}{N}}\right).$$

The case when $C^\pi \leq 1 + L/(200N)$. To obtain faster rates in this regime, we resort to directly analyzing the policy sub-optimality instead of bounding the value sub-optimality (such as by Lemma 3.11). It is useful to connect the sub-optimality of a policy to whether it disagrees with the optimal policy at each state. The following lemma due to Ross and Bagnell (2010, Theorem 2.1) provides such a connection.

Lemma 3.12. *For any deterministic policies $\pi, \hat{\pi}$, one has*

$$J(\pi) - J(\hat{\pi}) \leq H^2\,\mathbb{E}_{s\sim d_\pi}[\mathbb{1}\{\pi(s) \neq \hat{\pi}(s)\}].$$

We apply Lemma 3.12 to bound the sub-optimality and further decompose it based on whether any samples are observed on each state s.

$$\begin{aligned}&\mathbb{E}_{\mathcal{D}}[\rho(s)[V_1^\pi(s) - V_1^{\hat{\pi}}(s)]] \\ &\quad\leq H^2\,\mathbb{E}_{\mathcal{D}}\,\mathbb{E}_{d_\pi}[\mathbb{1}\{\pi(s) \neq \hat{\pi}(s)\}] \\ &\quad= H^2\,\mathbb{E}_{\mathcal{D}}\,\mathbb{E}_{d_\pi}[\mathbb{1}\{\pi(s) \neq \hat{\pi}(s)\}\,\mathbb{1}\{N(s,\pi(s)) = 0\}] =: T_1' \\ &\qquad+ H^2\,\mathbb{E}_{\mathcal{D}}\,\mathbb{E}_{d_\pi}[\mathbb{1}\{\pi(s) \neq \hat{\pi}(s)\}\,\mathbb{1}\{N(s,\pi(s)) \geq 1\}] =: T_2'.\end{aligned}$$

In a similar manner to the proof of Theorem 3.6, we prove the following bounds on T_1' and T_2':

$$T_1' \leq H^2 \frac{4C^\pi}{N}; \tag{3.89a}$$

$$T_2' \lesssim \frac{2SC^\pi H^2 L}{N} + H^2 \frac{|\mathcal{A}|}{N^9}. \tag{3.89b}$$

Proof the bound (3.88a) on T_1

By the value difference Lemma 3.11, one has

$$\begin{aligned}
\mathbb{E}_{\mathcal{D}}[\sum_s \rho(s)[V^\pi(s) - V^{\hat{\pi}}(s)]\, \mathbb{1}\{\mathcal{E}_{\text{EMDP}}\}] &\leq 2H \sum_{s,a} d^\pi(s,a)\, \mathbb{E}_{\mathcal{D}}[b(s,a)] \\
&\leq 2H \sum_{s,a} d^\pi(s,a) H\, \mathbb{E}_{\mathcal{D}} \left[\sqrt{\frac{L}{N(s,a) \vee 1}}\right] \\
&\leq 32H^2 \sum_{s,a} d^\pi(s,a) \left[\sqrt{\frac{L}{N\mu(s,a)}}\right],
\end{aligned}$$

where the last inequality uses the bound on inverse moments of binomial random variables given in 3.14 with $c_{1/2} \leq 16$. We then apply the concentrability assumption and the Cauchy-Schwarz inequality to conclude that

$$\begin{aligned}
T_1 &\leq 32H^2 \sum_{s,a} \sqrt{d^\pi(s,a)} \sqrt{HC^\pi \mu(s,a)} \left[\sqrt{\frac{L}{N\mu(s,a)}}\right] \\
&\leq 32H^2 \sqrt{\frac{C^\pi LH}{N}} \sum_s \sqrt{d^\pi(s,\pi(s))} \leq 32H^2 \sqrt{\frac{SC^\pi L}{N}}.
\end{aligned}$$

Proof of the bound (3.88b) on T_2

We use a argument similar to that in the proof of (3.54c). First, observe that $\sum_s \rho(s)[V_1^\pi(s) - V^{\hat{\pi}}(s)] \leq H$. Consequently, in light of Lemma 3.10 one can conclude

$$T_3 \leq H\, \mathbb{E}_{\mathcal{D}}[\mathbb{1}\{\mathcal{E}^c_{\text{EMDP}}\}] = H\, \mathbb{P}(\mathcal{E}^c_{\text{EMDP}}) \leq H\delta.$$

Proof of the bound (3.89a) on T_1'

We have

$$T_1' \leq H^2\, \mathbb{E}_{d_\pi}\, \mathbb{E}_{\mathcal{D}}[\mathbb{1}\{N(s,\pi(s)) = 0\}] \leq H^2\, \mathbb{E}_{d_\pi}\, \mathbb{P}(N(s,\pi(s)) = 0).$$

It follows from the concentrability assumption $d^\pi(s,\pi(s))/\mu(s,\pi(s)) \le C^\pi$ that

$$T_1 \le H^2 \sum_s C^\pi \mu(s,\pi(s))\, \mathbb{P}(N(s,\pi(s))=0) = H^2 C^\pi \sum_s \mu(s,\pi(s))(1-\mu(s,\pi(s)))^N.$$

Note that $\max_{x\in[0,1]} x(1-x)^N \le 4/(9N)$. We thus conclude that

$$T_1 \le H^2 C^\pi \sum_s \mu(s,\pi(s))(1-\mu(s,\pi(s)))^N \le H^2 \frac{4C^\pi}{9N}.$$

Proof of the bound (3.89b) on T_2'

We prove the bound on T_2' by partitioning the states based on how much they are occupied under the target policy. Define the following set:

$$\mathcal{O}_1 := \left\{ s \mid d_\pi(s) < \frac{2C^\pi L}{N} \right\}. \tag{3.90}$$

We can then decompose T_2' according to whether state s belongs to $\mathcal{O}_1$:

$$\begin{aligned} T_2' = H^2 \sum_{s\in\mathcal{O}_1} d_\pi(s)\, \mathbb{E}_{\mathcal{D}}[\mathbb{1}\{\hat\pi(s)\neq\pi(s)\}\, \mathbb{1}\{N(s,\pi(s))\ge 1\}] =: T_{2,1} \\ + H^2 \sum_{s\notin\mathcal{O}_1} d_\pi(s)\, \mathbb{E}_{\mathcal{D}}[\mathbb{1}\{\hat\pi(s)\neq\pi(s)\}\, \mathbb{1}\{N(s,\pi(s))\ge 1\}] =: T_{2,2}. \end{aligned}$$

Here, $T_{2,1}$ captures the sub-optimality due to the less important states under the target policy. We will shortly prove the following bounds on these two terms:

$$T_{2,1} \le \frac{2SC^\pi H^2 L}{N} \qquad \text{and} \qquad T_{2,2} \lesssim H^2 \frac{|\mathcal{A}|}{N^9}.$$

Proof of the bound on $T_{2,1}$. Since $\mathbb{E}_{\mathcal{D}}[\mathbb{1}\{\hat\pi(s)\neq\pi(s)\}\, \mathbb{1}\{N(s,\pi(s))\ge 1\}] \le 1$, it follows immediately that

$$T_{2,1} \le H^2 \sum_{s\in\mathcal{S}_1} d_\pi(s) \le \frac{2SC^\pi H^2 L}{N},$$

where the last inequality relies on the definition of $\mathcal{O}_1$ provided in (3.90).

Proof of the bound on $T_{2,2}$. The term $T_{2,2}$ is equal to

$$T_{2,2} = H^2 \sum_{s\notin\mathcal{O}_1} d_\pi(s) \mathbb{P}\left(\hat\pi(s)\neq\pi(s),\ N(s,\pi(s))\ge 1\right).$$

We subsequently show that the probability $\mathbb{P}\left(\hat{\pi}(s) \neq \pi(s),\ N(s, \pi(s)) \geq 1\right)$ is small. Fix a state $s \notin \mathcal{O}_1$ and let h be the level to which s belongs. The concentrability assumption along with the constraint on $d_\pi(s)$ implies the following lower bound on $\mu(s, \pi(s))$:

$$\mu(s, \pi(s)) \geq \frac{1}{C^\pi} d_\pi(s) \geq \frac{1}{C^\pi} \frac{2C^\pi L}{N} = \frac{2L}{N}. \tag{3.91}$$

On the other hand, by the concentrability assumption and using $C^\pi \leq 1 + \frac{L}{200N}$, the following upper bound holds for $\mu(s, a \neq \pi(s))$:

$$\mu(s, a) \leq \sum_{a \neq \pi(s)} \mu(s, a) \leq 1 - \frac{1}{C^\pi} \leq \frac{L}{200N}, \tag{3.92}$$

The above bounds suggest that the target action is likely to be included in the dataset more frequently than the rest of the actions for $s \notin \mathcal{O}_1$. We will see shortly that in this scenario, the LCB algorithm picks the target action with high probability. The bounds (3.91) and (3.92) together with Chernoff's bound give

$$\mathbb{P}\left(N(s, a \neq \pi(s)) \leq \frac{5L}{200}\right) \geq 1 - \exp\left(-\frac{L}{200}\right);$$
$$\mathbb{P}\left(N(s, \pi(s)) \geq L\right) \geq 1 - \exp\left(-\frac{L}{4}\right).$$

We can thereby write an upper bound $\hat{Q}_h(s, a \neq \pi(s))$ and a lower bound on $\hat{Q}_h(s, \pi(s))$. In particular, when $N(s, a) \leq \frac{5L}{200}$, one has

$$\begin{aligned}
\hat{Q}_h(s, a) &= r_h(s, a) - b_h(s, a) + \hat{P}_{s,a} \cdot \hat{V}_{h+1} \\
&= r_h(s, a) - H\sqrt{\frac{L}{N(s, a) \vee 1}} + \hat{P}_{s,a} \cdot \hat{V}_{h+1} \\
&\leq 1 - H\sqrt{\frac{L}{5L/200}} + H \leq -4H,
\end{aligned}$$

where we used the fact that $L \geq 70$. When $N(s, \pi(s)) \geq L$, one has

$$\hat{Q}_h(s, \pi(s)) = r_h(s, \pi(s)) - H\sqrt{\frac{L}{N(s, \pi(s))}} + \hat{P}_{s,\pi(s)} \cdot V_{h+1} \geq -H.$$

Note that if both $N(s, a \neq \pi(s)) \leq \frac{5L}{200}$ and $N(s, \pi(s)) \geq L$ hold, we must have $\hat{Q}_h(s, a \neq \pi(s)) < \hat{Q}_h(s, \pi(s))$. Therefore, we deduce that

$$\mathbb{P}\left(\hat{\pi}(s) \neq \pi(s),\ N(s, \pi(s)) \geq 1\right) \leq (|\mathcal{A}|-1)\exp\left(-\frac{L}{200}\right) + \exp\left(-\frac{1}{4}L\right) \leq |\mathcal{A}|\exp\left(-\frac{L}{200}\right),$$

which further implies

$$T_{2,2} \leq H^2 \sum_{s \notin \mathcal{O}_1} d_\pi(s)|\mathcal{A}|\exp\left(-\frac{L}{200}\right) \leq H^2|\mathcal{A}|\exp\left(-\frac{L}{200}\right) \lesssim H^2|\mathcal{A}|N^{-9}.$$

3.10.5 The case of $C^\pi \in [1, 2)$

In this section, we present an attempt in obtaining tight bounds on the LCB algorithm for episodic MDPs in the regime $C^\pi \in [1, 2)$. We start with a decomposition similar to the one given in (3.87).

$$\mathbb{E}_{\mathcal{D}}\left[\sum_s \rho(s)[V_1^\pi(s) - V^{\hat{\pi}}(s)]\right] = \mathbb{E}_{\mathcal{D}}\left[\sum_s \rho(s)[V_1^\pi(s) - V^{\hat{\pi}}(s)]\,\mathbb{1}\{\mathcal{E}_{\text{EMDP}}\}\right] =: T_1$$
$$+ \mathbb{E}_{\mathcal{D}}\left[\sum_s \rho(s)[V_1^\pi(s) - V^{\hat{\pi}}(s)]\,\mathbb{1}\{\mathcal{E}^c_{\text{EMDP}}\}\right] =: T_2.$$

An upper bound on the term T_2 is already proven in (3.88b). We follow a different route for bounding the term T_1. For any state $s \in \mathcal{S}$, define

$$\bar{\mu}(s) \coloneqq \sum_{a \neq \pi(s)} \mu(s, a) \tag{3.93}$$

to be the total mass on actions not equal to the target policy $\pi(s)$. Consider the following set:

$$\mathcal{B} \coloneqq \{s \mid \mu(s, \pi(s)) \leq 9\bar{\mu}(s)\}. \tag{3.94}$$

The set $\mathcal{B}$ includes the states for which the expert action is drawn more frequently under the data distribution. We then decompose T_1 based on whether state s belongs to $\mathcal{B}$

$$T_2 = \mathbb{E}_{\mathcal{D}}\left[\sum_{s\in\mathcal{B}} \rho(s)[V_1^\pi(s) - V^{\hat{\pi}}(s)]\,\mathbb{1}\{\mathcal{E}_{\text{EMDP}}\}\right] =: \beta_1 \tag{3.95}$$
$$+ \mathbb{E}_{\mathcal{D}}\left[\sum_{s\notin\mathcal{B}} \rho(s)[V_1^\pi(s) - V^{\hat{\pi}}(s)]\,\mathbb{1}\{\mathcal{E}_{\text{EMDP}}\}\right] =: \beta_2. \tag{3.96}$$

We prove the following bound on β_1:

$$\beta_1 \leq 136H^2\sqrt{\frac{S(C^\pi - 1)L}{N}}. \tag{3.97}$$

We *conjecture* that β_2 is bounded similarly:

$$\beta_2 \lesssim H^2\sqrt{\frac{S(C^\pi - 1)L}{N}}. \tag{3.98}$$

We demonstrate our conjecture on β_2 in a special episodic MDP case with $H = 3, |\mathcal{S}_h| = 2$, and $|\mathcal{A}| = 2$ in Section 3.10.6.

Proof of the bound (3.97) on β_1. By Lemma 3.10, it follows that

$$\begin{aligned}
\beta_1 &= \mathbb{E}_{\mathcal{D}}[\sum_{s\in\mathcal{B}} \rho(s)[V^{\pi}(s) - V^{\hat{\pi}}(s)]\, \mathbb{1}\{\mathcal{E}_{\text{EMDP}}\}] \\
&\leq 2H \sum_{s\in\mathcal{B}} d^{\pi}(s,\pi(s))\, \mathbb{E}_{\mathcal{D}}[b(s,\pi(s))] \\
&\leq 2H \sum_{s\in\mathcal{B}} d^{\pi}(s,\pi(s)) H\, \mathbb{E}_{\mathcal{D}}\left[\sqrt{\frac{L}{N(s,\pi(s))\vee 1}}\right] \\
&\leq 32H^2 \sum_{s\in\mathcal{B}} d^{\pi}(s,\pi(s)) \left[\sqrt{\frac{L}{N\mu(s,\pi(s))}}\right]
\end{aligned}$$

In the first inequality, we substituted the definition of penalty and the second inequality arises from Lemma 3.14 with $c_{1/2} \leq 16$. We then apply the concentrability assumption to bound $d^{\pi}(s,\pi(s)) \leq C^{\pi}\mu(s,\pi(s))$ and thereby conclude

$$\begin{aligned}
\beta_1 &\leq 32H^2 \sum_{s\in\mathcal{B}} C^{\pi}\mu(s,\pi(s)) \left[\sqrt{\frac{L}{N\mu(s,\pi(s))}}\right] \\
&= 32C^{\pi}H^2\sqrt{\frac{L}{N}} \sum_{s\in\mathcal{B}} \sqrt{\mu(s,\pi(s))} \\
&\leq 32C^{\pi}H^2\sqrt{\frac{LS}{N}} \sqrt{\sum_{s\in\mathcal{B}} \mu(s,\pi(s))},
\end{aligned}$$

where the last line is due to Cauchy-Schwarz inequality. We continue the bound relying on the definition of $\mathcal{B}$

$$\beta_1 \leq 32C^{\pi}H^2\sqrt{\frac{LS}{N}} \sqrt{\sum_{s} \mu(s,\pi(s))\, \mathbb{1}\{\mu(s,\pi(s)) \leq 9\bar{\mu}(s)\}} \leq 32C^{\pi}H^2\sqrt{\frac{LS}{N}} \sqrt{\sum_{s} 9\bar{\mu}(s)}. \tag{3.99}$$

It is easy to check that the concentrability assumption implies the following bound on the total mass over the actions not equal to $\pi(s)$

$$\sum_{s} \bar{\mu} \leq \frac{C^{\pi}-1}{C^{\pi}}.$$

Substituting the above bound to (3.99) and bounding $C^{\pi} \leq 2$ yields

$$\beta_1 \leq 136H^2\sqrt{\frac{S(C^{\pi}-1)L}{N}}.$$

3.10.6 Analysis of LCB for a simple episodic MDP

We consider an episodic MDP with $H = 3$, $\mathcal{S}_1 = \{1, 2\}$, $\mathcal{S}_2 = \{3, 4\}$, $\mathcal{S}_3 = \{5, 6\}$, and $\mathcal{A} = \{1, 2\}$, where we assume without loss of generality that action 1 is optimal in all states. We are interested in bounding the β_2 term defined in (3.96) when $C^\pi \in [1, 2)$:

$$\beta_2 = \mathbb{E}_{\mathcal{D}}\left[\sum_{s:\mu(s,\pi^\star(s))\geq 9\bar{\mu}(s)} \rho(s)[V_1^\pi(s) - V_1^{\hat{\pi}}(s)]\, \mathbb{1}\{\mathcal{E}_{\text{EMDP}}\}\right]. \tag{3.100}$$

Note that β_2 captures sub-optimality in states for which $\mu(s, \pi(s)) > 9\bar{\mu}(s)$. To illustrate the key ideas and avoid clutter, we consider the following setting:

1. Competing with the optimal policy $\pi(s) = \pi^\star(s) = 1$ and thus the concentrability assumption $d^\star(s, a) \leq C^\star \mu(s, a)$ for all $s \in \mathcal{S}, a \in \mathcal{A}$;
2. $\mu(s, 1) \geq 9\mu(s, 2)$ for all $s \in \mathcal{S}$;
3. $N(s, a) = N\mu(s, a) \geq 1$ for all $s \in \mathcal{S}, a \in \mathcal{A}$.
4. We assume that the rewards are deterministic and consider an implementation of Algorithm 5 with deterministic rewards. In particular, at level H this implementation of VI-LCB sets $\hat{Q}_H$ according to

$$\hat{Q}_H(s, a) = \begin{cases} 0 & N(s, a) = 0; \\ r(s, a) & N(s, a) \geq 1. \end{cases}$$

Outline of the proof. Let us first give an outline for the sub-optimality analysis of the episodic VI-LCB Algorithm 5 in this example. We begin by showing that the concentrability assumption in conjunction with $\mu(s, 1) \geq 9\mu(s, 2)$ dictates certain bounds on the penalties. Afterward, we argue that the episodic VI-LCB algorithm finds the optimal policy at levels 2 and 3 with high probability. This result allows writing the sub-optimality as an expectation over the product of the gap $g_1(s) = Q_1^\star(s, 1) - Q_1^\star(s, 2)$ and the probability that the agent chooses the wrong action, i.e., $\mathbb{P}(\hat{\pi}(s) \neq 1)$. Consequently, if for state s the gap $g_1(s)$ is small, the sub-optimality incurred by that state is also small. On the other hand, when the gap is large, we prove via Hoeffding's inequality that $\mathbb{P}(\hat{\pi}(s) \neq 1)$ is negligible.

Bounds on penalties. The setting introduced above dictates the following bounds on penalties

$$b_h(s, 2) - b_h(s, 1) \geq \frac{1}{3} b_h(s, 2) + b_h(s, 1), \tag{3.101a}$$

$$3\sqrt{\frac{LC^\star}{N(\bar{d}(s, 1) + C^\star - 1)}} \leq b_h(s, 1) \leq 3\sqrt{\frac{LC^\star}{N\bar{d}^\star(s, 1)}}, \tag{3.101b}$$

whose proofs can be found at the end of this subsection.

VI-LCB policy in each level. The main idea for a tight sub-optimality bound is to directly compare $\hat{Q}_h(s,1)$ to $\hat{Q}_h(s,2)$ at every level. Specifically, we first determine the conditions under which $\mathbb{E}[\hat{Q}_h(s,1) - \hat{Q}_h(s,2)] > 0$ and then show $\hat{Q}_h(s,1) > \hat{Q}_h(s,2)$ with high probability via a concentration argument. It turns out that these conditions depend on the value of the sub-optimality gap associated with a state defined as

$$g_h(s) \coloneqq Q_h^\star(s,1) - Q_h^\star(s,2) \geq 0 \quad \forall s \in \mathcal{S}, \forall h \in \{1,2,3\}. \tag{3.102}$$

We start the analysis at level 3 going backwards to level 1.

- **Level 3.** Since $N(s,a) \geq 1$ and the rewards are deterministic, the value function computed by VI-LCB algorithm is equal to $V_3^\star$ and action 1 is selected for both states 5 and 6, i.e.,

$$\hat{V}_3 = V_3^\star. \tag{3.103}$$

- **Level 2.** We first show that $\hat{Q}_2(s,1)$ is greater than $\hat{Q}_2(s,2)$ in expectation

$$\begin{aligned}\mathbb{E}[\hat{Q}_2(s,1) - \hat{Q}_2(s,2)] =& \mathbb{E}[r(s,1) - b_2(s,1) + \hat{P}_{s,1} \cdot V_3^\star - r(s,2) + b_2(s,2) - \hat{P}_{s,2} \cdot V_3^\star] \\ =& b_2(s,2) - b_2(s,1) + g_2(s) \\ \geq& \frac{1}{3} b_2(s,2) + b_2(s,1) + g_2(s) \geq \frac{1}{3} b_2(s,2) \geq 0,\end{aligned} \tag{3.104}$$

  where we used the bound on $b_2(s,2) - b_2(s,1)$ given in (3.101a). By the concentration inequality in Lemma 3.13 we then show $\hat{Q}_2(s,1) \geq \hat{Q}_2(s,2)$ with high probability:

$$\begin{aligned}\mathbb{P}(\hat{Q}_2(s,2) - \hat{Q}_2(s,1) \geq 0) &\leq \exp\left(-6 \frac{N(s,1)N(s,2)\,\mathbb{E}^2[\hat{Q}_2(s,1) - \hat{Q}_2(s,2)]}{N(s,1) + N(s,2)}\right) \\ &\leq \exp\left(-1.8 N(s,2) \left(\frac{1}{3}\right)^2 b_2^2(s,2)\right) \\ &= \exp\left(-0.8 N(s,2) \frac{L}{N(s,2)}\right) \lesssim \frac{1}{N^{160}},\end{aligned} \tag{3.105}$$

  where in the second inequality we used $N(s,2) \leq 1/9 N(s,1)$ as well as the bound given in (3.104) and the last inequality holds for $c_1 \geq 1$ and $\delta = 1/N$.

- **Level 1.** Define the following event

$$\mathcal{E}_o = \{\hat{\pi}(s) = 1,\ \forall s \in \mathcal{S}_2\}, \tag{3.106}$$

  which refers to the event that action 1 is chosen for all states at level 2. Conditioned on $\mathcal{E}_o$, the Q-function computed by VI-LCB in level 1 is given by

$$\begin{aligned}\hat{Q}_1(s,a) = r(s,a) - b_1(s,a) &+ \hat{P}(3 \mid s,a)[r(3,1) - b_2(3,1) + \hat{P}_{3,1} V_3^\star] \\ &+ \hat{P}(4 \mid s,a)[r(4,1) - b_2(4,1) + \hat{P}_{4,1} V_3^\star].\end{aligned} \qquad \forall s \in \mathcal{S}_1, a \in \mathcal{A}.$$

Taking the expectation with respect to the data randomness, one has for any $s \in \mathcal{B}$ that

$$\begin{aligned}
&\mathbb{E}[\hat{Q}_1(s,1) - \hat{Q}_1(s,2)] \\
&\quad = [b_1(s,2) - b_1(s,1)] + [P(3|s,2) - P(3|s,1)]b_2(3,1) \\
&\qquad + [P(4|s,2) - P(4|s,1)]b_2(4,1) + g_1(s) \\
&\quad = [b_1(s,2) - b_1(s,1)] + [P(3|s,1) - P(3|s,2)][b_2(4,1) - b_2(3,1)] + g_1(s),
\end{aligned}$$

where the last equation uses $P(3 \mid s,a) = 1 - P(4 \mid s,a)$. We continue the analysis assuming that $p \coloneqq P(3 \mid s,1) - P(3 \mid s,2) \geq 0$; the other case can be shown similarly. Using $p \geq 0$ and $b_2(4,1) \geq 0$ together with the penalty bound of (3.101a), we see that

$$\mathbb{E}[\hat{Q}_1(s,1) - \hat{Q}_1(s,2)] \geq \frac{1}{3}b_1(s,2) + b_1(s,1) - pb_2(3,1) + g_1(s).$$

We proceed by applying (3.101b) on $b_1(s,1)$ and $b_1(3,1)$

$$\mathbb{E}[\hat{Q}_1(s,1) - \hat{Q}_1(s,2)] \geq \frac{1}{3}b_1(s,2) + 3\sqrt{\frac{LC^\star}{N(d^\star(s,1) + C^\star - 1)}} - 3p\sqrt{\frac{LC^\star}{Nd^\star(3,1)}} + g_1(s). \tag{3.107}$$

Note that $d^\star(s,1) = \rho(s)/3$ and $3d^\star(3,1) = \rho(s)P(3|s,1) + \rho(2)P(3|s,2) \geq \rho(s)P(3|s,1) \geq \rho(s)p$. Substituting these quantities into (3.107), we obtain

$$\begin{aligned}
\mathbb{E}[\hat{Q}_1(s,1) - \hat{Q}_1(s,2)] &\geq \frac{1}{3}b_1(s,2) + 3\sqrt{\frac{LC^\star}{N(\rho(s)/3 + C^\star - 1)}} - 3p\sqrt{\frac{LC^\star}{N\rho(s)p/3}} + g_1(s) \\
&\geq \frac{1}{3}b_1(s,2) + 3\sqrt{\frac{LC^\star}{N(\rho(s)/3 + C^\star - 1)}} - 3\sqrt{\frac{LC^\star}{N\rho(s)/3}} + g_1(s),
\end{aligned}$$

where the last inequality uses $p \leq 1$. Observe that

$$\frac{1}{\sqrt{\rho(s)/3}} - \frac{1}{\sqrt{\rho(s)/3 + C^\star - 1}} = \frac{\sqrt{\rho/3 + C^\star - 1} - \sqrt{\rho/3}}{\sqrt{\rho(s)/3(\rho(s)/3 + C^\star - 1)}} \leq 3\frac{\sqrt{C^\star - 1}}{\rho(s)}.$$

This implies

$$\rho(s)g_1(s) \geq 9\sqrt{\frac{2(C^\star - 1)L}{N}} \quad \Rightarrow \quad \mathbb{E}[\hat{Q}_1(s,1) - \hat{Q}_1(s,2)] \geq \frac{1}{3}b_1(s,2). \tag{3.108}$$

Then, a similar argument to (3.105) proves that $\hat{Q}(s,1) > \hat{Q}(s,2)$ with high probability:

$$\mathbb{P}(\hat{Q}_1(s,2) - \hat{Q}_1(s,1) \geq 0) \lesssim \frac{1}{N^{160}}. \tag{3.109}$$

Sub-optimality bound. We are now ready to compute the sub-optimality. Decompose the sub-optimality based on whether event $\mathcal{E}_o$ defined in (3.106) has occurred and use the fact that we assumed $\mu(s,1) \geq 9\mu(s,2)$ for all $s \in \mathcal{S}$

$$\begin{aligned}\beta_2 &= \mathbb{E}_{\mathcal{D}}\left[\sum_{s:\mu(s,\pi^\star(s))\geq 9\bar{\mu}(s)} \rho(s)[V_1^{\pi}(s) - V_1^{\hat{\pi}}(s)]\,\mathbb{1}\{\mathcal{E}_{\text{EMDP}}\}\right] \\ &\leq \mathbb{E}_{\mathcal{D},\rho}\left[[V^\star(s) - V^{\hat{\pi}}(s)]\,\mathbb{1}\{\mathcal{E}_o\}\right] + \mathbb{E}_{\mathcal{D},\rho}\left[[V^\star(s) - V^{\hat{\pi}}(s)]\,\mathbb{1}\{\mathcal{E}_o^c\}\right] \\ &\lesssim \mathbb{E}_{\mathcal{D},\rho}\left[[V^\star(s) - V^{\hat{\pi}}(s)]\,\mathbb{1}\{\mathcal{E}_o\}\right] + \frac{3}{N^{160}}.\end{aligned}$$

Here, the second line is by $\mathbb{1}\{\mathcal{E}_{\text{EMDP}}\} \leq 1$ and the last line follows from $V^\star(s) - V^{\hat{\pi}}(s) \leq 3$ and the probability of the complement event $\mathcal{E}_o^c$ given in (3.105).

Conditioned on the event $\mathcal{E}_o$, LCB-VI algorithm chooses the optimal action from every state at levels 2 and 3 and hence $V_2^{\hat{\pi}} = V_2^\star$ and we get

$$\begin{aligned}&\mathbb{E}_{\mathcal{D},\rho}\left[[V^\star(s) - V^{\hat{\pi}}(s)]\,\mathbb{1}\{\mathcal{E}_o\}\right] \\ &= \sum_s \rho(s)\,\mathbb{E}_{\mathcal{D}}[[Q^\star(s,1) - Q^{\hat{\pi}}(s,\hat{\pi}(s))]\,\mathbb{1}\{\mathcal{E}_o\}] \\ &= \sum_s \rho(s)\,\mathbb{E}_{\mathcal{D}}[r(s,1) + P_{s,1}\cdot V_2^\star - r(s,\hat{\pi}(s)) - P_{s,\hat{\pi}(s)}\cdot V_2^\star] \\ &= \sum_s \rho(s)\,\mathbb{E}_{\mathcal{D}}[(r(s,1) + P_{s,1}\cdot V_2^\star - r(s,2) - P_{s,2}\cdot V_2^\star)\,\mathbb{1}\{\hat{\pi}(s)\neq 1\}].\end{aligned}$$

By definition, we have $g_1(s) = r(s,1) + P_{s,1}\cdot V_2^\star - r(s,2) - P_{s,2}\cdot V_2^\star$. Therefore,

$$\begin{aligned}\mathbb{E}_{\mathcal{D},\rho}\left[V^\star(s) - V^{\hat{\pi}}(s)\,\mathbb{1}\{\mathcal{E}_o\}\right] &\leq \sum_s \rho(s)g(s)\,\mathbb{E}_{\mathcal{D}}[\mathbb{1}\{\hat{\pi}(s)\neq 1\}] \\ &= \sum_s \rho(s)g(s)\,\mathbb{P}(\hat{Q}(s,2) - \hat{Q}(s,1)\geq 0).\end{aligned}$$

We decompose the sub-optimality based on whether $\rho(s)g_1(s)$ is large

$$\begin{aligned}\mathbb{E}_{\mathcal{D}}[J(\pi^\star) - J(\hat{\pi})] \leq &\sum_s \rho(s)g(s)\,\mathbb{P}(\hat{Q}(s,2) - \hat{Q}(s,1)\geq 0)\,\mathbb{1}\left\{\rho(s)g(s) \leq 9\sqrt{\frac{2(C^\star-1)L}{N}}\right\} =: \tau_1 \\ &+ \sum_s \rho(s)g_1(s)\,\mathbb{P}(\hat{Q}(s,2) - \hat{Q}(s,1)\geq 0)\,\mathbb{1}\left\{\rho(s)g_1(s) > 9\sqrt{\frac{2(C^\star-1)L}{N}}\right\} =: \tau_2 \\ &+ \frac{3}{N^{160}}.\end{aligned}$$

The first term is bounded by

$$\tau_1 \leq \sum_s 9\sqrt{\frac{2(C^\star-1)L}{N}} = 18\sqrt{\frac{2(C^\star-1)L}{N}}.$$

The second term is bounded using (3.109)

$$\tau_2 \lesssim \frac{3}{N^{160}}.$$

Combining the bounds yields the following sub-optimality bound

$$\beta_2 \lesssim \sqrt{\frac{(C^\star - 1)L}{N}} + \frac{1}{N^{160}}.$$

Proof of inequality (3.101a)**.** From $\mu(s,1) \geq 9\mu(s,2)$, one has $N(s,1) \geq 9N(s,2)$ implying $b_h(s,2) \geq 3b_h(s,1)$. Therefore, we conclude that

$$b_h(s,2) - b_h(s,1) = \frac{1}{2}(b_h(s,2) - b_h(s,1)) + \frac{1}{2}(b_h(s,2) - b_h(s,1)) \geq \frac{1}{3}b_h(s,2) + b_h(s,1).$$

Proof of inequality (3.101b)**.** The concentrability assumption implies the following bound on $\mu(s,1)$

$$\frac{\bar{d}(s,1)}{C^\star} \leq \mu(s,1) \leq \frac{\bar{d}(s,1)}{C^\star} + 1 - \frac{1}{C^\star},$$

The upper bound is based on the fact that the probability mass of at least $1/C^\star$ is distributed on the optimal actions with a remaining mass of $1 - 1/C^\star$. Applying the above bounds to $b_h(s,1)$, gives

$$3\sqrt{\frac{LC^\star}{N(\bar{d}(s,1) + C^\star - 1)}} \leq b_h(s,1) = 3\sqrt{\frac{L}{N\mu(s,1)}} \leq 3\sqrt{\frac{LC^\star}{N\bar{d}^\star(s,1)}}.$$

3.11 Auxiliary lemmas

This section collects a few auxiliary lemmas that are useful in the analysis of LCB.

We begin with a simple extension of the conventional Hoeffding bound to the two-sample case.

Lemma 3.13. *Let $X_1, \ldots, X_n$ be i.i.d. in range $[0,1]$ with average $\mathbb{E}[X]$ and $Y_1, \ldots, Y_m$ be i.i.d. in range $[0,1]$ with average $\mathbb{E}[Y]$. Further assume that $\{X_i\}$ and $\{Y_j\}$ are independent. Then for any ϵ such that $\epsilon + \mathbb{E}[Y] - \mathbb{E}[X] \geq 0$, we have*

$$\mathbb{P}\left(\frac{1}{n}\sum_i X_i - \frac{1}{m}\sum_j Y_j > \epsilon\right) \leq \exp\left(-2\frac{(mn)(\epsilon + \mathbb{E}[Y] - \mathbb{E}[X])^2}{m+n}\right).$$

Proof. It is easily seen that

$$\begin{aligned}
&\mathbb{P}\left(\sum_{i=1}^{n} mX_i - \sum_{j=1}^{m} nY_j > mn\epsilon\right) \\
&= \mathbb{P}\left(\sum_{i=1}^{n}(mX_i - m\,\mathbb{E}[X]) - \sum_{j=1}^{m}(nY_j - \mathbb{E}[Y]) > mn(\epsilon + \mathbb{E}[Y] - \mathbb{E}[X])\right) \\
&\leq \exp\left(-2\frac{(mn)^2(\epsilon + \mathbb{E}[Y] - \mathbb{E}[X])^2}{nm(m+n)}\right) \\
&= \exp\left(-2\frac{(mn)(\epsilon + \mathbb{E}[Y] - \mathbb{E}[X])^2}{m+n}\right),
\end{aligned}$$

where the inequality is based on Hoeffding's inequality on independent random variables. □

The next lemma provides useful bounds for the inverse moments of a binomial random variable.

Lemma 3.14 (Bound on binomial inverse moments)**.** *Let* $n \sim \text{Binomial}(N, p)$*. For any* $k \geq 0$*, there exists a constant* c_k *depending only on* k *such that*

$$\mathbb{E}\left[\frac{1}{(n \vee 1)^k}\right] \leq \frac{c_k}{(Np)^k},$$

where $c_k = 1 + k2^{k+1} + k^{k+1} + k\left(\frac{16(k+1)}{e}\right)^{k+1}$.

Proof. The proof is adapted from that of Lemma 21 in Jiao et al. (2018).

To begin with, when $p \leq 1/N$, the statement is clearly true for $c_k = 1$. Hence we focus on the case when $p > 1/N$. We define a useful helper function $g_N(p)$ to be

$$g_N(p) \coloneqq \begin{cases} \frac{1}{p^k}, & p \geq \frac{1}{N}, \\ N^k - kN^{k+1}(p - \frac{1}{N}), & 0 \leq p < \frac{1}{N}. \end{cases}$$

Further denote $\hat{p} \coloneqq n/N$. The proof relies heavily on the following decomposition, which is an direct application of the triangle inequality:

$$\mathbb{E}\left[\frac{N^k}{(n \vee 1)^k}\right] \leq \left|\mathbb{E}\left[\frac{N^k}{(n \vee 1)^k} - g_N(\hat{p})\right]\right| + |\mathbb{E}[g_N(p) - g_N(\hat{p})]| + g_N(p). \tag{3.110}$$

This motivates us to take a closer look at the helper function $g_N(p)$. Simple algebra reveals that

$$g_N(p) \leq \frac{1}{p^k} \quad \text{and} \quad g_N(\hat{p}) - \frac{N^k}{(n \vee 1)^k} = kN^k\,\mathbb{1}\{\hat{p} = 0\}.$$

Substitute these two facts back into the decomposition (3.110) to reach

$$\mathbb{E}\left[\frac{N^k}{(n\vee 1)^k}\right] \leq kN^k(1-p)^N + \frac{1}{p^k} + |\mathbb{E}[g_N(p) - g_N(\hat{p})]|.$$

It remains to bound the term $|\mathbb{E}[(g_N(p) - g_N(\hat{p}))^2]|$. To this goal, one has

$$\begin{aligned}
|\mathbb{E}[(g_N(p) - g_N(\hat{p}))^2]| &\leq |\mathbb{E}[(g_N(p) - g_N(\hat{p}))^2 1\{\hat{p} \geq p/2\}]| + |\mathbb{E}[(g_N(p) - g_N(\hat{p}))^2 1\{\hat{p} \geq p/2\}]| \\
&\overset{(i)}{\leq} \sup_{\xi \geq p/2} |g_N'(\xi)|^2 \mathbb{E}[(p-\hat{p})^2] + \sup_{\xi > 0} |g_N'(\xi)|^2 p^2\, \mathbb{P}(\hat{p} \leq p/2) \\
&\overset{(ii)}{\leq} \frac{k^2}{(p/2)^{2k+2}} \frac{p(1-p)}{N} + k^2 N^{2k+2} p^2 e^{-Np/8}.
\end{aligned}$$

Here the inequality (i) follows from the mean value theorem, and the last one (ii) uses the derivative calculation as well as the tail bound for binomial random variables; see e.g., Exercise 4.7 in Mitzenmacher and Upfal (2017). As a result, we conclude that

$$\begin{aligned}
\mathbb{E}\left[\frac{N^k}{(n\vee 1)^k}\right] &\leq kN^k(1-p)^N + \frac{1}{p^k} + \sqrt{\mathbb{E}[(g_N(p) - g_N(\hat{p}))^2]} \\
&\leq kN^k(1-p)^N + \frac{1}{p^k} + \frac{k}{(p/2)^{k+1}} \sqrt{\frac{p(1-p)}{N}} + kN^{k+1} p e^{-Np/16} \\
&\leq kN^k(1-p)^N + \frac{1}{p^k} + \frac{k2^{k+1}}{p^k} + kN^{k+1} p e^{-Np/16},
\end{aligned}$$

where the last inequality holds since $p \geq 1/N$. Consequently, we have

$$\mathbb{E}\left[\frac{(Np)^k}{(n\vee 1)^k}\right] \leq 1 + k2^{k+1} + k(Np)^k(1-p)^N + k(Np)^{k+1} e^{-Np/16}.$$

Note that the following two bounds hold:

$$\begin{aligned}
\max_p k(Np)^k(1-p)^N &\leq k\Big(N\frac{k}{N+k}\Big)^k \Big(1 - \frac{k}{k+N}\Big)^N \leq k^{k+1}, \\
(Np)^k e^{-Np/16} &\leq \Big(\frac{16k}{e}\Big)^k.
\end{aligned}$$

The proof is now completed. □

The last lemma, due to Gilbert and Varshamov (Gilbert, 1952; Varshamov, 1957), is useful for constructing hard instances in various minimax lower bounds.

Lemma 3.15. *There exists a subset $\mathcal{V}$ of $\{-1,1\}^S$ such that (1) $|\mathcal{V}| \geq \exp(S/8)$ and (2) for any $v_i, v_j \in \mathcal{V}$, $v_i \neq v_j$, one has $\|v_i - v_j\|_1 \geq \frac{S}{2}$.*

Chapter 4

Learning to Make Decisions During Interactions

In this chapter, we focus on the sequential-decision making problem in the more classical setting of online RL in which the agent learns a policy while interacting with the environment and collecting data. Online RL is a useful tool for an agent to learn how to perform tasks, particularly when expert demonstrations are unavailable and reward information needs to be used instead (Sutton and Barto, 2018). To learn a satisfactory policy, an RL agent needs to effectively balance between exploration and exploitation, which remains a central question in RL (Ecoffet et al., 2019; Burda et al., 2018b). Exploration is particularly challenging in environments with sparse rewards. One popular approach to exploration is based on *intrinsic motivation*, often applied by adding an intrinsic reward (or bonus) to the extrinsic reward provided by the environment.

In provable exploration methods, bonus often captures the value estimate uncertainty and the agent takes an action that maximizes the upper confidence bound (UCB) (Agrawal and Jia, 2017; Azar et al., 2017; Jaksch et al., 2010; Kakade et al., 2018; Jin et al., 2018). In tabular setting, UCB bonuses are often constructed based on either Hoeffding's inequality, which only uses visitation counts, or Bernstein's inequality, which uses value function variance in addition to visitation counts. The latter is proved to be minimax near-optimal in environments with bounded rewards (Jin et al., 2018; Menard et al., 2021) as well as bounded total reward (Zhang et al., 2020e) and reward-free settings (Ménard et al., 2020; Kaufmann et al., 2021; Jin et al., 2020a; Zhang et al., 2020f). It remains an open question how one can efficiently compute confidence bounds to construct UCB bonus in non-linear function approximation. Furthermore, Bernstein-style bonuses are often hard to compute in practice beyond tabular setting, due to difficulties in computing value function variance.

In practice, various approaches are proposed to design intrinsic rewards: visitation pseudo-count bonuses estimate count-based UCB bonuses using function approximation (Bellemare et al., 2016; Burda et al., 2018b), curiosity-based bonuses seek states where model prediction error is high, uncertainty-based bonuses (Pathak et al., 2019; Shyam et al., 2019) adopt ensembles of networks for estimating variance of the Q-function, empowerment-

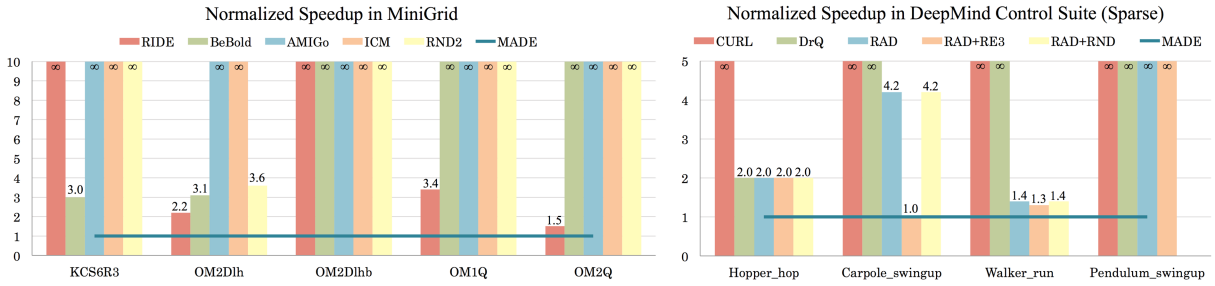

Figure 4.1: Normalized samples use of different methods with respect to MADE (smaller values are better). MADE consistency achieves a better sample efficiency compared to all other baselines. Infinity means the method fails to achieve maximum reward in given steps.

based approaches (Klyubin et al., 2005a; Gregor et al., 2016; Salge et al., 2014; Mohamed and Rezende, 2015) lead the agent to states over which the agent has control, and information gain bonuses (Kim et al., 2018) reward the agent based on the information gain between state-action pairs and next states.

Although the performance of practical intrinsic rewards is good in certain domains, empirically they are observed to suffer from issues such as detachment, derailment, and catastrophic forgetting (Agarwal et al., 2020a; Ecoffet et al., 2019). Moreover, these methods usually lack a clear objective and can get stuck in local optimum (Agarwal et al., 2020a). Indeed, the impressive performance currently achieved by some deep RL algorithms often revolves around manually designing dense rewards (Brockman et al., 2016), complicated exploration strategies utilizing a significant amount of domain knowledge (Ecoffet et al., 2019), or operating in the known environment regime (Silver et al., 2017; Moravčík et al., 2017).

Motivated by current practical challenges and the gap between theory and practice, we propose a new algorithm for exploration by maximizing deviation from explored regions. This yields a practical algorithm with strong empirical performance. To be specific, we make the following contributions:

1. Exploration via maximizing deviation Our approach is based on modifying the standard RL objective (i.e. the cumulative reward) by adding a regularizer that adaptively changes across iterations. The regularizer can be a general function depending on the state-action visitation density and previous state-action coverage. We then choose a particular regularizer that **MA**ximizes the **DE**viation (MADE) of the next policy visitation d^π from the regions covered by prior policies ρ^k_{cov}:

$$L_k(d^\pi) = J(d^\pi) + \tau_k \sum_{s,a} \sqrt{\frac{d^\pi(s,a)}{\rho^k_{\text{cov}}(s,a)}}. \tag{4.1}$$

Here, k is the iteration number, $J(d^\pi)$ is the standard RL objective, and the regularizer encourages $d^\pi(s,a)$ to be large when $\rho^k_{\text{cov}}(s,a)$ is small. We give an algorithm for solving

the regularized objective and prove that with access to an approximate planning oracle, it converges to the global optimum. We show that objective (4.1) results in an intrinsic reward that can be easily added to any RL algorithm to improve performance, as suggested by our empirical studies. Furthermore, the intrinsic reward applies a simple modification to the UCB-style bonus that considers prior visitation counts. This simple modification can also be added to existing bonuses in practice.

2. Tabular studies In the special case of tabular parameterization, we show that MADE only applies some simple adjustments to the Hoeffding-style count-based bonus. We compare the performance of MADE to Hoeffding and Bernstein bonuses in three different RL algorithms, for the exploration task in the stochastic diabolical bidirectional lock (Agarwal et al., 2020a; Misra et al., 2020), which has sparse rewards and local optima. Our results show that MADE robustly improves over the Hoeffding bonus and is competitive to the Bernstein bonus, across all three RL algorithms. Interestingly, MADE bonus and exploration strategy appear to be very close to the Bernstein bonus, *without computing or estimating variance*, suggesting that MADE potentially captures some environmental structures. Additionally, we empirically show that MADE regularizer can improve the optimization rate in policy gradient methods.

3. Experiments on MiniGrid and DeepMind Control Suite We empirically show that MADE works well when combined with model-free (IMAPLA (Espeholt et al., 2018), RAD (Laskin et al., 2020)) and model-based (Dreamer (Hafner et al., 2019)) RL algorithms, greatly improving the sample efficiency over existing baselines. When tested in the procedurally-generated MiniGrid environments, MADE manages to converge with two to five times fewer samples compared to state-of-the-art method BeBold (Zhang et al., 2020d). In DeepMind Control Suite (Tassa et al., 2020), we build upon the model-free method RAD (Laskin et al., 2020) and the model-based method Dreamer (Hafner et al., 2019), improving the return up to 150 in 500K steps compared to baselines. Figure 4.1 shows normalized sample size to achieve maximum reward with respect to our algorithm.

4.1 Background

As in Chapter 3, we consider the infinite-horizon discounted MDP, but we do not restrict the MDP to the tabular case and consider stationary (stochastic) policies $\pi \in \Delta(\mathcal{A} \mid \mathcal{S})$ instead of deterministic. Below, we review some definitions useful for the method and results presented in this chapter.

Policy mixture. For a sequence of policies $\mathcal{C}^k = (\pi_1, \ldots, \pi_k)$ with corresponding mixture distribution $w^k \in \Delta_{k-1}$, the policy mixture $\pi_{\text{mix},k} = (\mathcal{C}^k, w^k)$ is obtained by first sampling a policy from w^k and then following that policy over subsequent steps (Hazan et al., 2019). The mixture policy induces a state-action visitation density according to

$d^{\pi_{\text{mix}}}(s,a) = \sum_{i=1}^{k} w_i^k d^{\pi_i}(s,a)$. While the π_{mix} may not be stationary in general, there exists a stationary policy π' such that $d^{\pi'} = d^{\pi_{\text{mix}}}$; see Puterman (1990) for details.

Online reinforcement learning. Online RL is the problem of finding a policy with a maximum value from an unknown MDP, using samples collected during exploration. Oftentimes, the following objective is considered, which is a scalar summary of the performance of policy π:

$$J_M(\pi) \coloneqq \mathbb{E}_{s\sim\rho}[V^\pi(s)] = (1-\gamma)^{-1}\,\mathbb{E}_{(s,a)\sim d_\rho^\pi(\cdot,\cdot)}[r(s,a)]. \tag{4.2}$$

We drop index M when it is clear from context. We denote an optimal policy by $\pi^\star \in \arg\max_\pi J(\pi)$ and use the shorthand $V^\star \coloneqq V^{\pi^\star}$ to denote the optimal value function. It is straightforward to check that $J(\pi)$ can equivalently be represented by the expectation of the reward over the visitation measure of π. We slightly abuse the notation and sometimes write $J(d^\pi)$ to denote the RL objective.

4.2 Adaptive regularization of the RL objective

4.2.1 Regularization to guide exploration

In online RL, the agent faces a dilemma in each state: whether it should select a seemingly optimal policy (exploit) or it should explore different regions of the MDP. To allow flexibility in this choice and trade-off between exploration and exploitation, we propose to add a regularizer to the standard RL objective that changes throughout iterations of an online RL algorithm:

$$L_k(d^\pi) = \underbrace{J(d^\pi)}_{\text{exploitation}} + \tau_k \underbrace{R(d^\pi; \{d^{\pi_i}\}_{i=1}^k)}_{\text{exploration}}. \tag{4.3}$$

Here, $R(d^\pi; \{d^{\pi_i}\}_{i=1}^k)$ is a function of state-action visitation of π as well as the visitation of prior policies $\pi_1, \ldots, \pi_k$. The temperature parameter τ_k determines the strength of regularization. Objective (4.3) is a *population* objective in the sense that it does not involve empirical estimations affected by the randomness in sample collection. In the following section, we give our particular choice of regularizer and discuss how this objective can describe some popular exploration bonuses. We then provide a convergence guarantee for the regularized objective in Section 4.2.2.

4.2.2 Exploration via maximizing deviation from policy cover

We develop our exploration strategy MADE based on a simple intuition: maximizing the deviation from the explored regions, i.e. all states and actions visited by prior policies. We define *policy cover* at iteration k to be the density over regions explored by policies

$\pi_1, \ldots, \pi_k$, i.e. $\rho^k_{\text{cov}}(s,a) := \frac{1}{k}\sum_{i=1}^k d^{\pi_i}(s,a)$. We then design our regularizer to encourage d^π to be different from ρ^k_{cov}:

$$R_k(d^\pi; \{d^{\pi_i}\}_{i=1}^k) = \sum_{s,a} \sqrt{\frac{d^\pi(s,a)}{\rho^k_{\text{cov}}(s,a)}}. \tag{4.4}$$

It is easy to check that the maximizer of above function is $d^\pi(s,a) \propto \frac{1}{\rho^k_{\text{cov}}(s,a)}$. Our motivation behind this particular deviation is that it results in a simple modification of UCB bonus in tabular case.

We now compute the reward yielded by the new objective. First, define a policy mixture $\pi_{\text{mix},k}$ with policy sequence $(\pi_1, \ldots, \pi_k)$ and weights $((1-\eta)^{k-1}, (1-\eta)^{k-2}\eta, (1-\eta)^{k-3}\eta, , \ldots, \eta)$ for $\eta > 0$. Let $d^{\pi_{\text{mix},k}}$ be the visitation density of $\pi_{\text{mix},k}$. We compute the total reward at iteration k by taking the gradient of the new objective with respect to d^π at $d^{\pi_{\text{mix},k}}$:

$$r_k(s,a) = (1-\gamma)\nabla_d L_k(d)\big|_{d=d^{\pi_{\text{mix},k}}} = r(s,a) + (1-\gamma)\tau_k \nabla_d R_k(d; \{d^{\pi_i}\}_{i=1}^k)\big|_{d=d^{\pi_{\text{mix},k}}}, \tag{4.5}$$

which gives the following reward

$$r_k(s,a) = r(s,a) + \frac{(1-\gamma)\tau_k/2}{\sqrt{d^{\pi_{\text{mix},k}}(s,a)\rho^k_{\text{cov}}(s,a)}}. \tag{4.6}$$

The intrinsic reward above is constructed based on two densities: ρ^k_{cov} a uniform combination of past visitation densities and $\hat{d}^{\pi_{\text{mix},k}}$ a (almost) geometric mixture of the past visitation densities. As we will discuss shortly, policy cover $\rho^k_{\text{cov}}(s,a)$ is related to the visitation count of (s,a) pair in previous iterations and resembles count-based bonuses (Bellemare et al., 2016; Jin et al., 2018) or their approximates such as RND (Burda et al., 2018b). Therefore, for an appropriate choice of τ_k, MADE intrinsic reward decreases as the number of visitations increases.

MADE intrinsic reward is also proportional to $1/\sqrt{d^{\pi_{\text{mix},k}}(s,a)}$, which can be viewed as a correction applied to the count-based bonus. In effect, due to the decay of weights in $\pi_{\text{mix},k}$, the above construction gives a higher reward to (s,a) pairs visited earlier. Experimental results suggest that this correction may alleviate major difficulties in sparse reward exploration, namely detachment and catastrophic forgetting, by encouraging the agent to revisit forgotten states and actions.

Empirically, MADE's intrinsic reward is computed based on estimates $\hat{d}^{\pi_{\text{mix},k}}$ and $\hat{\rho}^k_{\text{cov}}$ from data collected by iteration k. Furthermore, practically we consider a smoothed version of the above regularizer by adding $\lambda > 0$ to both numerator and denominator; see Equation (4.7).

MADE intrinsic reward in tabular case. In tabular setting, the empirical estimation of policy cover is simply $\hat{\rho}^k_{\text{cov}}(s,a) = N_k(s,a)/N_k$, where $N_k(s,a)$ is the visitation count of (s,a)

Algorithm 6 Policy computation for adaptively regularized objective

Inputs: Iteration count K, planning error ϵ_p, visitation density error ϵ_d.
Initialize policy mixture $\pi_{\text{mix},1} =$ with $\mathcal{C}_1 = (\pi_1)$ and $w^1 = (1)$
for $k = 1, \ldots, K$ **do**
    Estimate the visitation density $\hat{d}^{\pi_{\text{mix},k}}$ of $\pi_{\text{mix},k}$ via a visitation density oracle.
    Compute reward $r_k(s,a) = r(s,a) + (1-\gamma)\tau_k \nabla_d R_k(d; \{\pi_i\}_{i=1}^k)\big|_{d=\hat{d}^{\pi_{\text{mix},k}}}$.
    Run approximate planning on modified MDP $M^k = (\mathcal{S}, \mathcal{A}, P, r_k, \gamma)$ and return π_{k+1}.
    Update policy mixture $\mathcal{C}^{k+1} = (C_k, \pi_{k+1})$ and $w^{k+1} = ((1-\eta)w^k, \eta)$.
Return: $\pi_{\text{mix},K} = (\mathcal{C}^k, w^k)$.

pair and N_k is the total count by iteration k. Thus, MADE simply modifies the Hoeffding-type bonus via the mixture density and has the following form: $\propto 1/\sqrt{\hat{d}^{\pi_{\text{mix},k}}(s,a) N_k(s,a)}$.

Bernstein bonus is another tabular UCB bonus that modifies Hoeffding bonus via an empirical estimate of the value function variance. Bernstein bonus is shown to improve over Hoeffding count-only bonus by exploiting additional environment structure (Zanette and Brunskill, 2019) and close the gap between algorithmic upper bounds and information-theoretic limits up to logarithmic factors (Zhang et al., 2020e,f). However, a practical and efficient implementation of a bonus that exploits variance information in non-linear function approximation parameterization still remains an open question; see Section 4.5 for further discussion. On the other hand, our proposed modification based on the mixture density can be easily and efficiently incorporated with non-linear parameterization.

Deriving some popular bonuses from regularization. We now discuss how the regularization in (4.3) can describe some popular bonuses. Exploration bonuses that only depend on state-action visitation counts can be expressed in the form (4.3) by setting the regularizer a linear function of d^π and the exploration bonus $r_i(s,a)$, i.e., $R_k(d^\pi; \{d^{\pi_i}\}_{i=1}^k) = \sum_{s,a} d^\pi(s,a) r_i(s,a)$. It is easy to check that taking the gradient of the regularizer with respect to d^π recovers $r_i(s,a)$. As another example, one can set the regularizer to Shannon entropy $R_k(d^\pi; \{d^{\pi_i}\}_{i=1}^k) = -\sum_{s,a} d^\pi(s,a) \log d^\pi(s,a)$, which gives the intrinsic reward $-\log d^\pi(s,a)$ (up to an additive constant) and recovers the result in the work Zhang et al. (2021a).

4.2.3 Solving the regularized objective

We pair MADE objective with the algorithm proposed by Hazan et al. (2019) extended to the adaptive objective. We provide convergence guarantees for Algorithm 6 in the following theorem whose proof is given in Appendix 4.7.1.

Theorem 4.1. *Consider the following regularizer for (4.3) with $\lambda > 0$ and a valid visitation density d*

$$R_\lambda(d; \{d^{\pi_i}\}_{i=1}^k) = \sum_{s,a} \sqrt{\frac{d(s,a) + \lambda}{\rho^k_{cov}(s,a) + \lambda}}, \tag{4.7}$$

Set $\tau_k = \tau/k^c$, where $0 < \tau < 1$ and $c > 0$. For any $\epsilon > 0$, there exists $\eta, \epsilon_p, \epsilon_d, c, B$ such that $\pi_{\text{mix},K}$ returned by Algorithm 6 after $K \geq \eta^{-1} \log(10B\epsilon^{-1})$ iterations satisfies $L_k(d^{\pi_{\text{mix},K}}) \geq \max_\pi L_k(d^\pi) - \epsilon$.

Remark 4.1. *One does not need to maintain the functional forms of past policies to estimate $\hat{d}^{\pi_{\text{mix},k}}$. Practically, one may truncate the dataset to a (prioritized) buffer and estimate the density over that buffer.*

4.3 A tabular study

We first study the performance of MADE in tabular toy examples. In the Bidirectional Lock experiment, we compare MADE to theoretically guaranteed Hoeffding-style and Bernstein-style bonuses in a sparse reward exploration task. In the Chain MDP, we investigate whether MADE's regularizer (4.4) provides any benefits in improving optimization rate in policy gradient methods.

4.3.1 Exploration in bidirectional lock

We consider a stochastic version of the bidirectional diabolical combination lock (Figure 4.3), which is considered a particularly difficult exploration task in tabular setting (Misra et al., 2020; Agarwal et al., 2020a). This environment is challenging because: (1) positive rewards are sparse, (2) a small negative reward is given when transiting to a good state and thus,

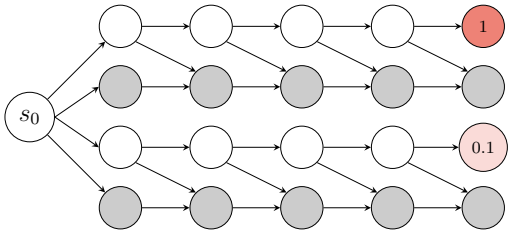

Figure 4.2: A stochastic bidirectional lock. In this environment, the agent starts at s_0 and enters one of the chains based on the selected action. Each chain has a positive reward at the end, H good states, and H dead states. Both actions available to the agent lead it to the dead state, one with probability one and the other with probability $p < 1$.

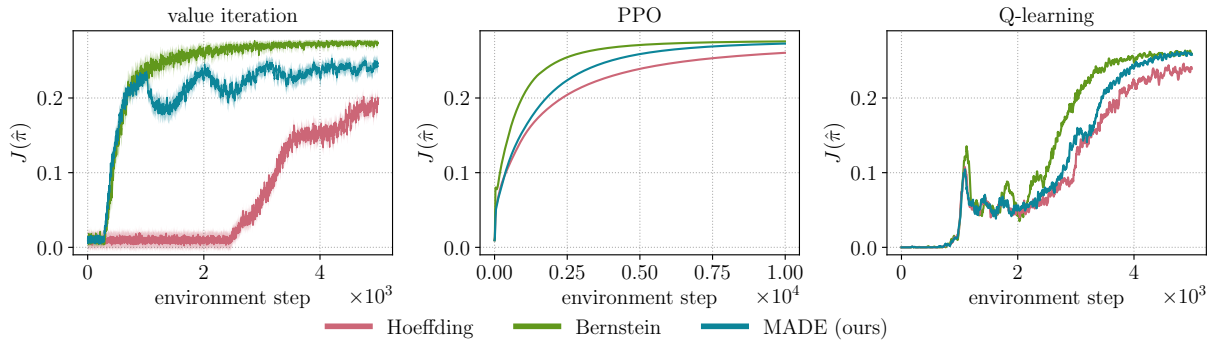

Figure 4.3: Performance of different count-based methods in the stochastic bidirectional lock environment. MADE performs better than the Hoeffding bonus and is comparable to the Bernstein bonus.

moving to a dead state is locally optimal, and (3) the agent may forget to explore one chain and get stuck in local minima upon receiving an end reward in one lock (Agarwal et al., 2020a).

RL algorithms and exploration strategies. We compare the performance of Hoeffding and Bernstein bonuses (Jin et al., 2018) to MADE in three different RL algorithms. To implement MADE in tabular setting, we simply use two buffers: one that stores all past state-action pairs to estimate ρ_{cov} and another one that only maintains the most recent B pairs to estimate d^{π}_{μ}. We use empirical counts to estimate both densities, which give a bonus $\propto 1/\sqrt{N_k(s,a)B_k(s,a)}$, where $N_k(s,a)$ is the total count and $B_k(s,a)$ is the recent buffer count of (s,a) pair. We combine three bonuses with three RL algorithms: (1) value iteration with bonus (He et al., 2020), (2) proximal policy optimization (PPO) with a model (Cai et al., 2020), and (3) Q-learning with bonus (Jin et al., 2018).

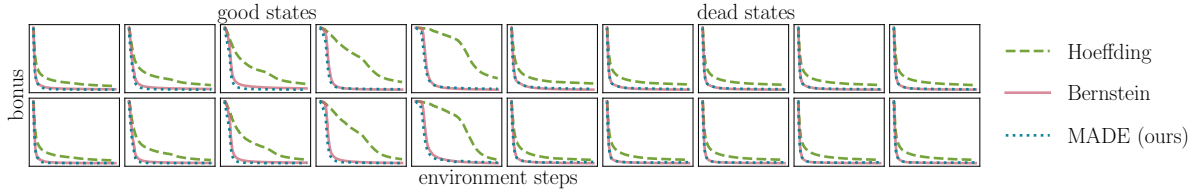

Figure 4.4: Values of Hoeffding, Bernstein, and MADE exploration bonus for all states and action 1 over environment steps in the bidirectional lock MDP. MADE bonus values closely follows Bernstein bonus values.

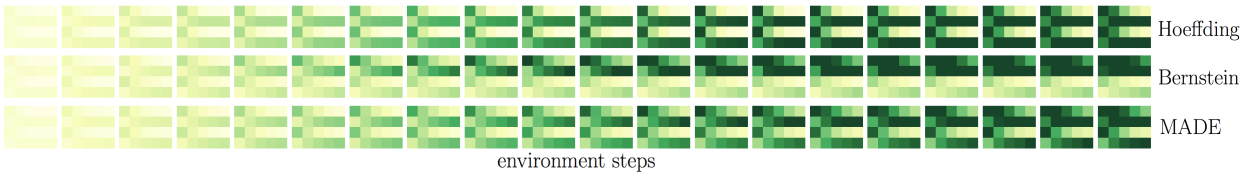

Figure 4.5: Heatmap of visitation counts in the bidirectional lock, plotted every 200 iterations. The exploration strategy of MADE appears to be closet to the Bernstein bonus.

Results. Figure 4.3 summarizes our results showing MADE improves over the Hoeffding bonus and is competitive to the Bernstein bonus in all three algorithms. Unlike Bernstein bonus that is hard to compute beyond tabular setting, MADE bonus design is simple and can be effectively combined with any deep RL algorithm. The experimental results suggest several interesting properties for MADE. First, MADE applies a simple modification to the Hoeffding bonus which improves the performance. Second, as illustrated in Figures 4.4 and 4.5, bonus values and exploration pattern of MADE is somewhat similar to the Bernstein bonus. This suggests that MADE may capture some structural information of the environment, similar to Bernstein bonus, which captures certain environmental properties such as the degree of stochasticity (Zanette and Brunskill, 2019).

4.3.2 Policy gradient in a chain MDP

We consider the chain MDP (Figure 4.6) presented in Agarwal et al. (2019b), which suffers from vanishing gradients with policy gradient approach (Sutton et al., 1999) as a positive reward is only achieved if the agent always takes action a_1. This leads to an exponential iteration complexity lower bound on the convergence of vanilla policy gradient approach

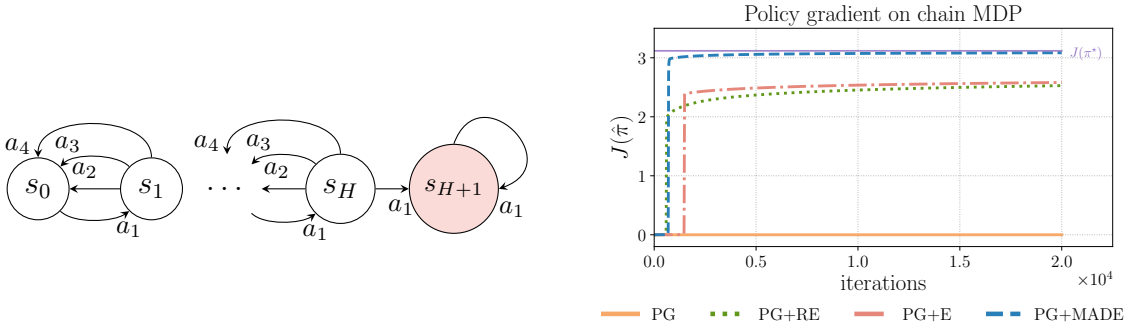

Figure 4.6: A deterministic chain MDP that suffers from vanishing gradients (Agarwal et al., 2019b). We consider a constrained tabular policy parameterization with $\pi(a|s) = \theta_{s,a}$ and $\sum_a \theta_{s,a} = 1$. The agent always starts from s_0 and the only non-zero reward is $r(s_{H+1}, a_1) = 1$.

even with access to exact gradients (Agarwal et al., 2019b). In this environment the agent always starts at state s_0 and recent guarantees on the global convergence of exact policy gradients are vacuous (Bhandari and Russo, 2019; Agarwal et al., 2019b; Mei et al., 2020). This is because the rates depend on the ratio between the optimal and learned visitation densities, known as *concentrability coefficient* (Kakade and Langford, 2002; Scherrer, 2014; Geist et al., 2017; Rashidinejad et al., 2021), or the ratio between the optimal visitation density and initial distribution (Agarwal et al., 2019b).

RL algorithms. Since our goal in this experiment is to investigate the optimization effects and not exploration, we assume access to exact gradients. In this setting, we consider MADE regularizer with the form $\sum_{s,a} \sqrt{d^\pi(s,a)}$. Note that policy gradients take gradient of the objective with respect to the policy parameters θ and not d^π. We compare optimizing the policy gradient objective with four methods: vanilla version PG (e.g. uses policy gradient theorem (Williams, 1992; Sutton et al., 1999; Konda and Tsitsiklis, 2000)), relative policy entropy regularization PG+RE (Agarwal et al., 2019b), policy entropy regularization PG+E (Mnih et al., 2016; Mei et al., 2020), and MADE regularization.

Results. Figure 4.6 illustrates our results on policy gradient methods. As expected (Agarwal et al., 2019b), the vanilla version has a very slow convergence rate. Both entropy and relative entropy regularization methods are proved to achieve a linear convergence rate of $\exp(-t)$ in the iteration count t (Mei et al., 2020; Agarwal et al., 2019b). Interestingly, MADE seems to outperforms the policy entropy regularizers, quickly converging to a globally optimal policy.

4.4 Experiments on MiniGrid and DeepMind Control Suite

In addition to the tabular setting, MADE can also be integrated with various model-free and model-based deep RL algorithms such as IMPALA (Espeholt et al., 2018), RAD (Lee et al., 2019a), and Dreamer (Hafner et al., 2019). As we will see shortly, MADE exploration strategy on MiniGrid (Chevalier-Boisvert et al., 2018) and DeepMind Control Suite (Tassa et al., 2020) tasks achieves state-of-the-art sample efficiency.

For a practical estimation of ρ^k_{cov} and $d^{\pi_{\text{mix},k}}$, we adopt the two buffer idea described in the tabular setting. However, since now the state space is high-dimensional, we use RND (Burda et al., 2018b) to estimate $N_k(s,a)$ (and thus ρ^k_{cov}) and use a variational auto-encoder (VAE) to estimate $d^{\pi_{\text{mix},k}}$. Specifically, for RND, we minimize the difference between a predictor network $\phi'(s,a)$ and a randomly initialized target network $\phi(s,a)$ and train it in an online manner as the agent collects data. We sample data from the recent buffer $\mathcal{B}$ to train a VAE. The length of $\mathcal{B}$ is a design choice for which we do an ablation study. Thus, the intrinsic

reward in deep RL setting takes the following form

$$(1-\gamma)\tau_k \frac{\|\phi(s,a)-\phi'(s,a)\|}{\sqrt{d^{\pi_{\text{mix},k}}(s,a)}}.$$

Model-free RL baselines. We consider several baselines in MiniGrid: **IMPALA** (Espeholt et al., 2018) is a variant of policy gradient algorithms which we use as the training baseline; **ICM** (Pathak et al., 2017) learns a forward and reverse model for predicting state transition and uses the forward model prediction error as intrinsic reward; **RND** (Burda et al., 2018b) trains a predictor network to mimic a randomly initialized target network as discussed above; **RIDE** (Raileanu and Rocktäschel, 2020) learns a representation similar to ICM and uses the difference of learned representations along a trajectory as intrinsic reward; **AMIGo** (Campero et al., 2020) learns a teacher agent to assign intrinsic reward; **BeBold** (Zhang et al., 2020d) adopts a regulated difference of novelty measure using RND. In DeepMind Control Suite, we consider **RE3** (Seo et al., 2021) as a baseline which uses a random encoder for state embedding followed by a k-nearest neighbour bonus for a maximum state coverage objective.

Model-based RL baselines. MADE can be combined with model-based RL algorithms to improve sample efficiency. For baselines, we consider **Dreamer**, which is a well-known model-based RL algorithm for DeepMind Control Suite, as well as **Dreamer+RE3**, which includes RE3 bonus on top of Dreamer.

MADE achieves state-of-the-art results on both navigation and locomotion tasks by a substantial margin, greatly improving the sample efficiency of the RL exploration in both model-free and model-based methods. Further details on experiments and exact hyperparameters are provided in Appendix 4.8.

4.4.1 Model-free RL on MiniGrid

MiniGrid (Chevalier-Boisvert et al., 2018) is a widely used benchmark for exploration in RL. Despite having symbolic states and a discrete action space, MiniGrid tasks are quite challenging. The easiest task is **MultiRoom** (MR) in which the agent needs to navigate to the goal by going to different rooms connected by the doors. In **KeyCorridor** (KC), the agent needs to search around different rooms to find the key and then use it to open the door. **ObstructedMaze** (OM) is a harder version of KC where the key is hidden in a box and sometimes the door is blocked by an obstruct. In addition to that, the entire environment is procedurally-generated. This adds another layer of difficulty to the problem.

From Figure 4.7 we can see that MADE manages to solve all the challenging tasks within 90M steps while all other baselines (except BeBold) only solve up to 50% of them. Compared to BeBold, MADE uses significantly (2-5 times) fewer samples.

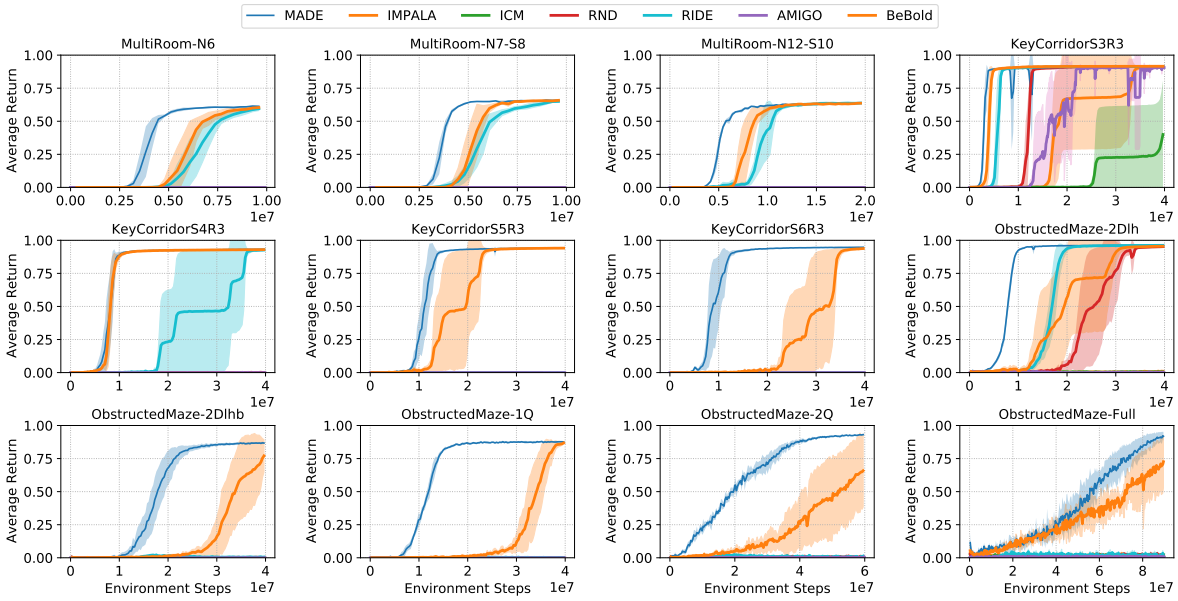

Figure 4.7: Results for various hard exploration tasks from MiniGrid. MADE successfully solves all the environments while other algorithms (except for BeBold) fail to solve several environments. MADE finds the optimal solution with 2-5 times fewer samples, yielding a much better sample efficiency.

4.4.2 Model-free RL on DeepMind Control

We also test MADE on image-based continuous control tasks of DeepMind Control Suite (Tassa et al., 2020), which is a collection of diverse control tasks such as Pendulum, Hopper, and Acrobot with realistic simulations. Compared to MiniGrid, these tasks are more realistic and complex as they involve stochastic transitions, high-dimensional states, and continuous actions. For baselines, we build our algorithm on top of RAD (Lee et al., 2019a), a strong model-free RL algorithm with a competitive sample efficiency. We compare our approach with ICM, RND, as well as RE3, which is the SOTA algorithm.[1] Note that we compare MADE to very strong baselines. Other algorithms such as DrQ (Kostrikov et al., 2020), CURL (Srinivas et al., 2020), ProtoRL (Yarats et al., 2021), SAC+AE (Yarats et al., 2019)) perform worse based on the results reported in the original papers. MADE show consistent improvement in sample efficiency: 2.6 times over RAD+RE3, 3.3 times over RAD+RND, 19.7 times over CURL, 15.0 times over DrQ and 3.8 times over RAD.

From Figure 4.8, we can see that MADE consistently improves sample efficiency compared to all baselines. For these tasks, RND and ICM do not perform well and even fail

[1] As we were not provided with the source code, we implemented ICM and RND ourselves. The performance for ICM is slightly worse than what the author reported, but the performance of RND and RE3 is similar.

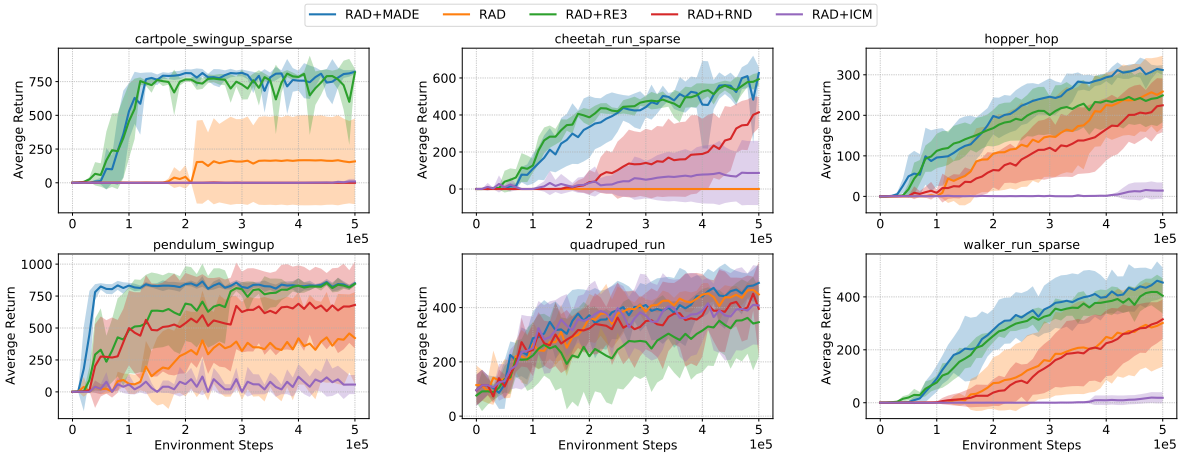

Figure 4.8: Results for several DeepMind control suite locomotion tasks. Comparing to all baselines, the performance of MADE is consistently better. Sometimes baseline methods even fail to solve the task.

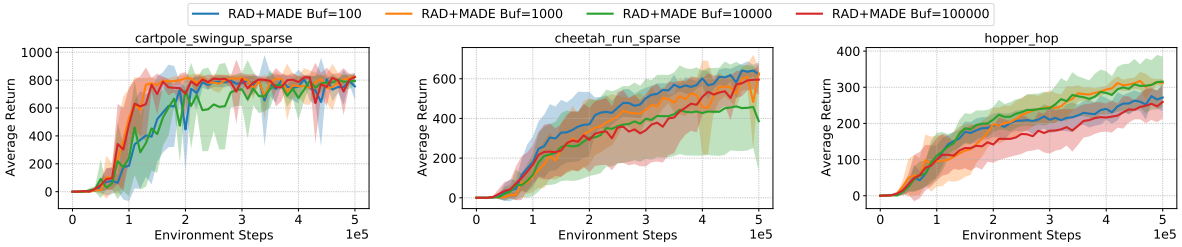

Figure 4.9: Ablation study on buffer size in MADE. The optimal buffer size varies in different tasks. We found buffer size of 10000 empirically works consistently reasonable.

on `Cartpole-Swingup`. RE3 achieves a comparable performance in two tasks, however, its performance on `Pendulum-Swingup`, `Quadruped-Run`, `Hopper-Hop` and `Walker-Run` is significantly worse than MADE. For example, in `Pendulum-Swingup`, MADE achieves a reward of around 800 in only 30K steps while RE3 requires 300k samples. In `Quadruped-Run`, there is a 150 reward gap between MADE and RE3, which seems to be still enlarging. These tasks show the strong performance of MADE in model-free RL.

Ablation study. We study how the buffer length affects the performance of our algorithm in some DeepMind Control tasks. Results illustrated in Figure 4.9 show that for different tasks the optimal buffer length is slightly different. We empirically found that using a buffer length of 1000 consistently works well across different tasks.

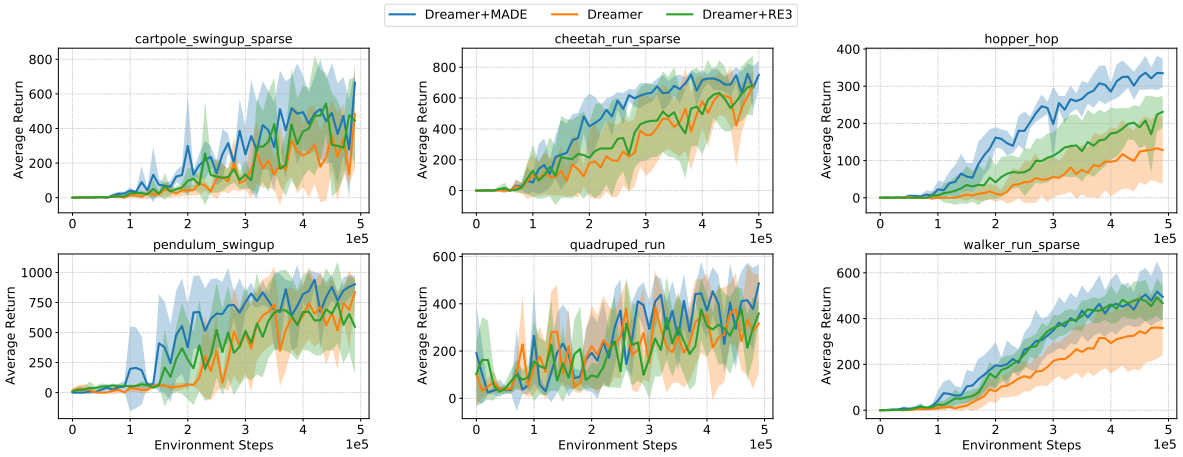

Figure 4.10: Results for DeepMind control suite locomotion tasks in model-based RL setting. Comparing to all baselines, the performance of MADE is consistently better. Some baseline methods even fail to solve the task.

4.4.3 Model-based RL on DeepMind Control

We also empirically verify the performance of MADE combined with the SOTA model-based RL algorithm Dreamer (Hafner et al., 2019). We compare MADE with Dreamer and Dreamer combined with RE3 in Figure 4.10. Results show that MADE has great sample efficiency in `Cheetah-Run-Sparse`, `Hopper-Hop` and `Pendulum-Swingup` environments. For example, in `Hopper-Hop`, MADE achieves more than 100 higher return than RE3 and 250 higher return than Dreamer, achieving a new SOTA result.

4.5 Related work

Provable optimistic exploration. Most provable exploration strategies are based on optimism in the face of uncertainty (OFU) principle. In tabular setting, model-based exploration algorithms include variants of UCB (Kearns and Singh, 2002; Brafman and Tennenholtz, 2002), UCRL (Lattimore and Hutter, 2012; Jaksch et al., 2010; Zanette and Brunskill, 2019; Kaufmann et al., 2021; Ménard et al., 2020), and Thompson sampling (Xiong et al., 2021; Agrawal and Jia, 2017; Russo, 2019) and value-based methods include optimistic Q-learning (Jin et al., 2018; Wang et al., 2019c; Strehl et al., 2006; Liu and Su, 2020; Menard et al., 2021) and value-iteration with UCB (Azar et al., 2017; Zhang et al., 2020e,f; Jin et al., 2020a). These methods are recently extended to linear MDP setting leading to a variety of model-based (Zhou et al., 2020a; Ayoub et al., 2020; Jia et al., 2020; Zhou et al., 2020b), value-based (Wang et al., 2019b; Jin et al., 2020b), and policy-based algorithms (Cai et al., 2020; Zanette et al., 2021; Agarwal et al., 2020a). Going beyond linear function approximation, systematic exploration strategies are developed based on structural assumptions

on MDP such as low Bellman rank (Jiang et al., 2017) and block MDP (Du et al., 2019a). These methods are either computationally intractable (Jiang et al., 2017; Sun et al., 2019; Ayoub et al., 2020; Zanette et al., 2020; Yang et al., 2020a; Dong et al., 2021; Wang et al., 2020b) or are only oracle efficient (Feng et al., 2020; Agarwal et al., 2020b). The recent work Feng et al. (2021) provides a sample efficient approach with non-linear policies, however, the algorithm requires maintaining the functional form of all prior policies.

Practical exploration via intrinsic reward. Apart from previously-discussed methods, other works give intrinsic reward based on the difference in (abstraction of) consecutive states (Zhang et al., 2019; Marino et al., 2019; Raileanu and Rocktäschel, 2020). However, this approach is inconsistent: the intrinsic reward does not converge to zero and thus, even with infinite samples, the final policy does not maximize the RL objective. Other intrinsic rewards try to estimate pseudo-counts (Bellemare et al., 2016; Tang et al., 2017; Burda et al., 2018b,a; Ostrovski et al., 2017; Badia et al., 2020), inspired by count-only UCB bonus. Though favoring novel states, practically these methods might suffer from *detachment and derailment* (Ecoffet et al., 2019, 2020), and *forgetting* (Agarwal et al., 2020a). More recent works propose a combination of different criteria. RIDE (Raileanu and Rocktäschel, 2020) learns a representation via a curiosity criterion and uses the difference of consecutive states along the trajectory as the bonus. AMIGo (Campero et al., 2020) learns a teacher agent for assigning rewards for exploration. Go-Explore (Ecoffet et al., 2019) explicitly decouples the exploration and exploitation stages, yielding a more sophisticated algorithm with many hand-tuned hyperparameters.

Maximum entropy exploration. Another line of work encourages exploration via maximizing some type of entropy. One category maximizes policy entropy (Mnih et al., 2016) or relative entropy (Agarwal et al., 2019b) in addition to the RL objective. The work of Flet-Berliac et al. (2021) modifies the RL objective by introducing an adversarial policy which results in the next policy to move away from prior policies while staying close to the current policy. In contrast, our approach focuses on the regions explored by prior policies as opposed to the prior policies themselves. Recently, effects of policy entropy regularization have been studied theoretically (Neu et al., 2017; Geist et al., 2019). In policy gradient methods with access to exact gradients, policy entropy regularization results in faster convergence by improving the optimization landscape (Mei et al., 2020, 2021; Ahmed et al., 2019; Cen et al., 2020). Another category considers maximizing the entropy of state or state-action visitation densities such as Shannon entropy (Hazan et al., 2019; Islam et al., 2019; Lee et al., 2019b; Seo et al., 2021) or Rényi entropy (Zhang et al., 2021a). Empirically, our approach achieves better performance over entropy-based methods.

Other exploration strategies. Besides intrinsic motivation, other strategies are also fruitful in encouraging the RL agent to visit a wide range of states. One example is exploration by injecting noise to the action action space (Lillicrap et al., 2015; Osband et al.,

2016; Hessel et al., 2017; Osband et al., 2019) or parameter space (Fortunato et al., 2018; Plappert et al., 2018). Another example is the reward-shaping category, in which diverse goals are set to guide exploration (Colas et al., 2019; Florensa et al., 2018; Nair et al., 2018; Pong et al., 2020).

4.6 Discussion

We introduce a new exploration strategy MADE based on maximizing deviation from explored regions. We show that by simply adding a regularizer to the original RL objective, we get an easy-to-implement intrinsic reward which can be incorporated with any RL algorithm. We provide a policy computation algorithm for this objective and prove that it converges to a global optimum, provided that we have access to an approximate planner. In tabular setting, MADE consistently improves over the Hoeffding bonus and shows competitive performance to the Bernstein bonus, while the latter is impractical to compute beyond tabular. We conduct extensive experiments on MiniGrid, showing a significant (over 5 times) reduction of the required sample size. MADE also performs well in DeepMind Control Suite when combined with both model-free and model-based RL algorithms, achieving SOTA sample efficiency results. One limitation of the current work is that it only uses the naive representations of states (e.g., one-hot representation in tabular case). In fact, exploration could be conducted much more efficiently if MADE is implemented with a more compact representation of states. We leave this direction to future work.

4.7 Convergence analysis of MADE algorithm

In this section, we provide a convergence rate analysis for Algorithm 6. Similar to Hazan et al. (2019), Algorithm 6 has access to an approximate density oracle and an approximate planner defined below:

- *Visitation density oracle:* We assume access to an approximate density estimator that takes in a policy π and a density approximation error $\epsilon_d \geq 0$ as inputs and returns $\hat{d}^\pi$ such that $\|d^\pi - \hat{d}^\pi\|_\infty \leq \epsilon_d$.
- *Approximate planning oracle:* We assume access to an approximate planner that, given any MDP M and error tolerance $\epsilon_p \geq 0$, returns a policy π such that $J_M(\pi) \geq \max_\pi J_M(\pi) - \epsilon_p$.

4.7.1 Proof of Theorem 4.1

We first give the following proposition that captures certain properties of the proposed objective. The proof is postponed to the end of this section.

Proposition 4.1. *Consider the following regularization for* $\lambda > 0$

$$R_\lambda(d; \{d^{\pi_i}\}_{i=1}^k) = \sum_{s,a} \sqrt{\frac{d(s,a)+\lambda}{\rho_{cov}(s,a)+\lambda}},$$

with $\tau_k = \tau/k^c$ *where* $\tau < 1, c > 0$. *There exist constants* β, B, *and* ξ *that only depend on MDP parameters and* λ *such that* $L_k(d) := J(d) + \tau_k R_\lambda(d; \{d^{\pi_i}\}_{i=1}^k)$ *satisfies the following regularity conditions for all* $k \geq 1$, *an appropriate choice of* c, *and valid visitation densities* d *and* d'*:*

(i) $L_k(d)$ *is concave in* d;

(ii) $L_k(d)$ *is* β*-smooth:* $\|\nabla L_k(d) - \nabla L_k(d')\|_\infty \leq \beta \|d - d'\|_\infty, \quad -\beta I \preceq \nabla^2 L_k(d) \preceq \beta I$;

(iii) $L_k(d)$ *is* B*-bounded:* $L_k(d) \leq B, \quad \|\nabla L_k(d)\|_\infty \leq B$;

(iv) There exists δ_k *such that* $\max_d L_{k+1}(d) - L_k(d) \leq \delta_k$ *and we have* $\sum_{i=0}^k (1-\eta)^i \delta_{k-i} \leq \tau\xi$.

Taking the above proposition as given for the moment, we prove Theorem 4.1 following steps similar to those of Hazan et al. (2019, Theorem 4.1). By construction of the mixture density $d^{\pi_{\text{mix},k}}$, we have

$$d^{\pi_{\text{mix},k}} = (1-\eta) d^{\pi_{\text{mix},k-1}} + \eta d^{\pi_k}.$$

Combining the above equation with the β-smoothness of $L_k(d)$ yields

$$\begin{aligned} L_k(d^{\pi_{\text{mix},k}}) &= L_k((1-\eta) d^{\pi_{\text{mix},k-1}} + \eta d^{\pi_k}) \\ &\geq L_k(d^{\pi_{\text{mix},k-1}}) + \eta \langle d^{\pi_k} - d^{\pi_{\text{mix},k-1}}, \nabla L_k(d^{\pi_{\text{mix},k-1}}) \rangle - \eta^2 \beta \|d^{\pi_k} - d^{\pi_{\text{mix},k-1}}\|_2^2 \\ &\geq L_k(d^{\pi_{\text{mix},k-1}}) + \eta \langle d^{\pi_k} - d^{\pi_{\text{mix},k-1}}, \nabla L_k(d^{\pi_{\text{mix},k-1}}) \rangle - 4\eta^2 \beta. \end{aligned} \tag{4.8}$$

Here the last inequality uses $\|d^{\pi_k} - d^{\pi_{\text{mix},k-1}}\|_2 \leq 2$. By property (ii), we bound $\langle d^{\pi_k}, \nabla L_k(d^{\pi_{\text{mix},k-1}}) \rangle$ according to

$$\begin{aligned} \langle d^{\pi_k}, \nabla L_k(d^{\pi_{\text{mix},k-1}}) \rangle &\geq \langle d^{\pi_k}, \nabla L_k(\hat{d}^{\pi_{\text{mix},k-1}}) \rangle - \beta \|d^{\pi_{\text{mix},k-1}} - \hat{d}^{\pi_{\text{mix},k-1}}\|_\infty \\ &\geq \langle d^{\pi_k}, \nabla L_k(\hat{d}^{\pi_{\text{mix},k-1}}) \rangle - \beta \epsilon_d, \end{aligned} \tag{4.9}$$

where in the last step we used the density oracle approximation error. Recall that we defined $r_k = (1-\gamma) \nabla L_k(\hat{d}^{\pi_{\text{mix},k-1}})$. Since π_k returned by the approximate planning oracle is an ϵ_p-optimal policy in M^k, we have $(1-\gamma)^{-1} \langle d^{\pi_k}, r_k \rangle \geq (1-\gamma)^{-1} \langle d^{\pi}, r_k \rangle - \epsilon_p$ for any policy π, including $\pi^\star$. Therefore,

$$\begin{aligned} \langle d^{\pi_k}, \nabla L_k(d^{\pi_{\text{mix},k-1}}) \rangle &\geq \langle d^{\pi^\star}, \nabla L_k(\hat{d}^{\pi_{\text{mix},k-1}}) \rangle - \epsilon_p - \beta \epsilon_d \\ &\geq \langle d^{\pi^\star}, \nabla L_k(d^{\pi_{\text{mix},k-1}}) \rangle - \epsilon_p - 2\beta \epsilon_d, \end{aligned} \tag{4.10}$$

where we used the density oracle approximation error once more in the second step. Going back to inequality (4.8), we further bound $L_k(d^{\pi_{\text{mix},k}})$ by

$$\begin{aligned} L_k(d^{\pi_{\text{mix},k}}) &\geq L_k(d^{\pi_{\text{mix},k-1}}) + \eta\langle d^{\pi_k} - d^{\pi_{\text{mix},k-1}}, \nabla L_k(d^{\pi_{\text{mix},k-1}})\rangle - 4\eta^2\beta \\ &\geq L_k(d^{\pi_{\text{mix},k-1}}) + \eta\langle d^{\pi^\star} - d^{\pi_{\text{mix},k-1}}, \nabla L_k(d^{\pi_{\text{mix},k-1}})\rangle - \eta\epsilon_p - 2\eta\beta\epsilon_d - 4\eta^2\beta \\ &\geq (1-\eta)L_k(d^{\pi_{\text{mix},k-1}}) + \eta L_k(d^{\pi^\star}) - 4\eta^2\beta - \eta\epsilon_p - 2\eta\beta\epsilon_d, \end{aligned}$$

where the last inequality is by concavity of $L_k(d)$. Therefore,

$$L_k(d^{\pi^\star}) - L_k(d^{\pi_{\text{mix},k}}) \leq (1-\eta)[L_k(d^{\pi^\star}) - L_k(d^{\pi_{\text{mix},k-1}})] + 2\eta\beta\epsilon_d + \eta\epsilon_p + 4\eta^2\beta.$$

By assumption (iv), we write

$$\begin{aligned} L_{K+1}(d^{\pi^\star}) - L_{K+1}(d^{\pi_{\text{mix},K}}) &\leq L_K(d^{\pi^\star}) - L_K(d^{\pi_{\text{mix},K}}) + 2\delta_K \\ &\leq (1-\eta)[L_K(d^{\pi^\star}) - L_K(d^{\pi_{\text{mix},K-1}})] + 2\delta_K + 2\eta\beta\epsilon_d + \eta\epsilon_p + 4\eta^2\beta \\ &\leq Be^{-\eta K} + 2\beta\epsilon_d + \epsilon_p + 4\eta\beta + 2\sum_{i=0}^{K}(1-\eta)^i\delta_{K-i} \\ &\leq Be^{-\eta K} + 2\beta\epsilon_d + \epsilon_p + 4\eta\beta + 2\tau\xi. \end{aligned}$$

It is straightforward to check that setting $\eta \leq 0.1\epsilon\beta^{-1}, \epsilon_p \leq 0.1\epsilon, \epsilon_d \leq 0.1\epsilon\beta^{-1}, \tau \leq 0.1\epsilon$, and the number of iterations $K \geq \eta^{-1}\log(10B\epsilon^{-1})$ yields the claim of Theorem 4.1.

Remark 4.2. *Since the temperature parameter τ_k in Proposition 4.1 goes to zero as k increases, one can show that the expected value of policy returned by Algorithm 6 converges to the maximum performance $J(\pi^\star)$.*

Proof of Proposition 4.1. For claim (ii), observe that $\nabla^2 L_k(d)$ is a diagonal matrix whose (s,a) diagonal term is given by

$$(\nabla^2 L_k(d))_{s,a} = \frac{-\tau}{4k^c} \times \frac{1}{(d(s,a)+\lambda)^{3/2}(\rho_{\text{cov}}(s,a)+\lambda)^{1/2}}.$$

The diagonal elements are bounded by $-1/(4\lambda^2) \leq (\nabla^2 L_k(d))_{s,a} \leq \frac{1}{4\lambda^2} =: \beta$. Furthermore, by Taylor's theorem, one has

$$\|\nabla L_k(d) - \nabla L_k(d')\|_\infty \leq \max_{(s,a),\alpha\in[0,1]} (\nabla^2 L_k(\alpha d + (1-\alpha)d'))\|d-d'\|_\infty \leq \beta\|d-d'\|_\infty.$$

Claim (i) is immediate from the above calculation as the Hessian $\nabla^2 L_k(d)$ is negative definite. Claim (iii) may be verified by explicit calculation:

$$\sum_{s,a} d(s,a)r(s,a) + \frac{\tau}{k^c}\sum_{s,a}\sqrt{\frac{d(s,a)+\lambda}{\rho_{\text{cov}}(s,a)+\lambda}} \leq SA\left(1+\sqrt{\frac{1+\lambda}{\lambda}}\right) =: B.$$

For claim (iv), we have

$$L_{k+1}(d) - L_k(d) \leq \sum_{s,a} \frac{\tau}{(k+1)^c} \sqrt{\frac{d(s,a)+\lambda}{\rho_{\text{cov}}(s,a)+\lambda}} \leq \frac{SA\tau}{(k+1)^c} \sqrt{\frac{1+\lambda}{\lambda}} =: \delta_k$$

We have

$$\sum_{i=0}^{k} (1-\eta)^i \delta_{k-i} = \tau SA \sqrt{\frac{1+\lambda}{\lambda}} \sum_{i=0}^{k} \frac{(1-\eta)^i}{(k-i+1)^c}.$$

For example, for $c = 2$, the above sum is bounded by $\sum_{n=1}^{\infty} 1/n^2 = \pi^2/6$. Thus, one can set $\xi := \frac{\pi^2 SA}{6} \sqrt{\frac{1+\lambda}{\lambda}}$. □

4.8 Experimental details

Source code is included in the supplemental material.

4.8.1 Bidirectional lock

Environment. For the bidirectional lock environment, one of the locks (randomly chosen) gives a larger reward of 1 and the other lock gives a reward of 0.1. Further details on this environment can be found in the work Agarwal et al. (2020a).

Exploration bonuses. We consider three exploration bonuses:

- Hoeffding-style bonus is equal to

  $$\frac{V_{\max}}{\sqrt{N_k(s,a)}},$$

  for every $s \in \mathcal{S}, a \in \mathcal{A}$, where $V_{\max}$ is the maximum possible value in an environment which we set to 1 for bidirectional lock.

- We use a Bernstein-style bonus

  $$\sqrt{\frac{\text{Var}_{s' \sim P_k(\cdot|s,a)} V_k(s')}{N_k(s,a)}} + \frac{1}{N_k(s,a)}$$

  based on the bonus proposed by He et al. (2020). P_k denotes an empirical estimation of transitions $P_k(s'|s,a) = N_k(s,a,s')/N_k(s,a)$, where $N_k(s,a,s')$ is the number of samples on transiting to s' starting from state s and taking action a.

- MADE's bonus is set to the following in tabular setting:

  $$\frac{1}{\sqrt{N_k(s,a) B_k(s,a)}}.$$

Algorithms. Below, we describe details on each tabular algorithm.

- **Value iteration.** We implement discounted value iteration given in (He et al., 2020) with all three bonuses.
- **PPO.** We implement a tabular version of the algorithm in (Cai et al., 2020), which is based on PPO with bonus. Specifically, the algorithm has the following steps: (1) sampling a new trajectory by running the stochastic policy π_k, (2) updating the empirical transition estimate P_k and exploration bonus, (3) computing Q-function Q_k of π_k over an MDP M_k with empirical transitions P_k and total reward r_k which is a sum of extrinsic reward and exploration bonus, and (4) updating the policy according to $\pi_{k+1}(a|s) \propto \pi_k(a|s)\exp(\alpha_k Q_k(s,a))$, where $\alpha_k = \sqrt{2\log(A)/HK}$ based on Cai et al. (2020, Theorem 13.1).
- **Q-learning.** We implement Q-learning with bonus based on the algorithms given by Jin et al. (2018).

4.8.2 Chain MDP

For the chain MDP described in Section 4.3.2, we use $H = 8$ and discount factor $\gamma = H/(H+1)$. We run policy gradient for a tabular softmax policy parameterization $\pi(s|a) = \theta_{s,a}$ with the following RL objectives. Since we use a simplex parameterization, we run *projected* gradient ascent.

- **Vanilla PG.** The vanilla version simply considers the standard RL objective $J(\pi_\theta)$. For the gradient $\nabla_\theta J(\pi_\theta)$, see e.g. Agarwal et al. (2019b, Equation (32)).
- **PG with relative policy entropy regularization.** We use the objective (with the additive constant dropped) given in Agarwal et al. (2019b, Equation (12)):

$$L(\pi_\theta) \coloneqq J(\pi_\theta) + \tau_k \sum_{s,a} \log \pi_\theta(a|s).$$

  Here, index k denotes the policy gradient step. This form of regularization is more aggressive than the policy entropy regularized objective discussed next. Partial derivatives of the above objective are simply

$$\frac{\partial L(\pi_\theta)}{\partial \theta_{s,a}} = \frac{\partial J(\pi_\theta)}{\partial \theta_{s,a}} + \tau_k \frac{1}{\theta_{s,a}},$$

  where the first term is analogous to the vanilla policy gradient.
- **PG with policy entropy regularization.** Policy entropy regularized objective (Williams and Peng, 1991; Mnih et al., 2016; Nachum et al., 2017; Mei et al., 2020) is

$$L(\pi_\theta) \coloneqq J(\pi_\theta) - \tau_k(1-\gamma)^{-1}\, \mathbb{E}_{(s,a)\sim d_\rho^{\pi_\theta}(\cdot,\cdot)}[\log \pi_\theta(a|s)].$$

  The gradient of the regularizer of the above objective is given in Lemma 4.1.

- **PG with MADE's regularization.** For MADE, we use the following objective

$$L(\pi_\theta) \coloneqq J(\pi_\theta) - \tau_k \sum_{s,a} \sqrt{d^\pi(s,a)}.$$

  The gradient of MADE's regularizer is computed in Lemma 4.2.

For all regularized objectives, we set $\tau_k = 0.1/\sqrt{k}$.

4.8.3 MiniGrid

We follow RIDE (Campero et al., 2020) and use the same hyperparameters for all the baselines. For ICM, RND, IMPALA, RIDE, BeBold and MADE, we use the learning rate 10^{-4}, batch size 32, unroll length 100, RMSProp optimizer with $\epsilon = 0.01$ and momentum 0. For entropy cost hyperparameters, we use 0.0005 for all the baselines except AMIGo. We provide the entropy cost for AMIGo below. We also test different values $\{0.01, 0.02, 0.05, 0.1, 0.5\}$ for the temperature hyperparameter in MADE. The best hyperparameters we found for each method are as follows. For **Bebold**, **RND**, and **MADE** we use intrinsic reward scaling factor of 0.1 for all environments. For **ICM** we use intrinsic reward scaling factor of 0.1 for KeyCorridor environments and 0.5 for the others. Hyperparameters in **RIDE** are exactly the same as **ICM**. For **AMIGo**, we use an entropy cost of 0.0005 for the student agent, and an entropy cost of 0.01 for the teacher agent.

4.8.4 DeepMind Control Suite

Environment. We use the publicly available environment DeepMind Control Suite (Tassa et al., 2020) without any modification (Figure 4.11). Following the task design of RE3 (Seo et al., 2021), we use `Cheetah_Run_Sparse` and `Walker_Run_Sparse`.

Figure 4.11: Visualization of various tasks in DeepMind Control Suite. DeepMind Control Suite includes image-based control tasks with physics simulation. We mainly experiment on locomotion tasks in this environment.

Model-free RL implementations. For the experiments, we use the baselines of RAD (Laskin et al., 2020), and we conduct a hyperparameter search over certain factors:

- **RND.** We search for the temperature parameter τ_k over $\{0.001, 0.01, 0.05, 0.1, 0.5, 10.0\}$ and choose the best for each task. Specifically we use $\tau_k = 0.1$ for `Pendulum_Swingup` and `Cheetah_Run_Sparse`, $\tau_k = 10$ for `Cartpole_Swingup_Sparse`, and $\tau_k = 0.05$ for others.
- **ICM.** We search for the temperature parameter τ_k over $\{0.001, 0.01, 0.05, 0.1, 0.5, 1.0\}$ and choose the best for each task. Specifically we select $\tau_k = 1.0$ for `Cheetah_Run_Sparse` and $\tau_k = 0.1$ for the others. For the total loss used in training the networks, to balance the coefficient between forward loss and inverse loss, we follow the convention and use $L_{\text{all}} = 0.2 \cdot L_{\text{forward}} + 0.8 \cdot L_{\text{inverse}}$, where L_{forward} is the loss of predicting the next state given current state-action pair and L_{inverse} is the loss for predicting the action given the current state and the next state.
- **RE3**. We use an initial scaling factor $\tau_0 = 0.05$ (the scaling factor of τ_k at step 0) and decay it afterwards in each step. Note that we use the number of clusters $M = 3$ with a decaying factor on the reward $\rho = \{0.00001, 0.000025\}$. Therefore, the final intrinsic reward scaling factor becomes: $\tau_k = \tau_0 e^{-\rho k}$.
- **MADE.** We search for the temperature parameter τ_k over $\{0.001, 0.01, 0.05, 0.1, 0.5\}$ and choose the best for each task. Specifically we select $\tau_k = 0.05$ for `Cartpole_Swingup_Sparse`, `Walker_Run_Sparse` and `Cheetah_Run_Sparse`, $\tau_k = 0.5$ for `Hopper_Hop` and `Pendulum_Swingup`, and $\tau_k = 0.001$ for `Quadruped_Run`.

We use the same network architecture for all the algorithms. Specifically, the encoder consists of 4 convolution layers with ReLU activations. There are kernels of size 3×3 with 32 channels for all layers, and stride 1 except for the first layer which has stride 2. The embedding is then followed by a LayerNorm.

Model-based Rl implementation Here we provide implementation details for the model-based RL experiments. We adopt Dreamer as a baseline and build all the algorithms on top of that.

- **RE3.** For RE3, we follow the hyperparameters given in the original paper. We use an initial scaling factor $\tau_0 = 0.1$ without decaying τ_k afterwards. The number of clusters is set to $M = 50$. We use a decaying factor on the reward $\rho = 0$.
- **MADE.** We search for the temperature parameter τ_k over $\{0.0005, 0.01, 0.05, 0.1, 0.5\}$ and choose the best for each map. Specifically we use 0.5 for `Cartpole_Swingup_Sparse`, `Cheetah_Run_Sparse` and `Hopper_Hop`, 0.01 for `Walker_Run_Sparse` and `Pendulum_Swingup` and 0.0005 for `Quadruped_Run`.

4.9 Gradient computations

In this section we compute the gradients for policy entropy and MADE regularizers used in the chain MDP experiment. Before presenting the lemmas, we define two other visitation densities. The state visitation density $d^\pi : \mathcal{S} \to [0, 1]$ is defined as

$$d^\pi(s) \coloneqq (1-\gamma) \sum_{t=0}^{\infty} \gamma^t \, \mathbb{P}_t(s_t = s; \pi),$$

where $\mathbb{P}_t(s_t = s; \pi)$ denotes the probability of visiting s at step t starting at $s_0 \sim \rho(\cdot)$ following policy π. The state-action visitation density starting at (s', a') is denoted by

$$d^\pi_{s',a'}(s, a) \coloneqq (1-\gamma) \sum_{t=0}^{\infty} \gamma^t \, \mathbb{P}_t(s_t = s, a_t = a; \pi, s_0 = s', a_0 = a').$$

The following lemma computes the gradient of policy entropy with respect to policy parameters.

Lemma 4.1. *For a policy π parameterized by θ, the gradient of the policy entropy*

$$R(\pi_\theta) \coloneqq - \mathbb{E}_{(s,a)\sim d^{\pi_\theta}_\rho(\cdot,\cdot)}[\log \pi_\theta(a|s)],$$

with respect to θ is given by

$$\nabla_\theta R(\pi_\theta) = \mathbb{E}_{(s,a)\sim d^{\pi_\theta}_\rho(\cdot,\cdot)} \left[\nabla_\theta \log \pi(a|s) \left(\frac{1}{1-\gamma} \langle d^\pi_{s,a}, -\log \pi \rangle \right) - \log \pi(a|s) \right].$$

Proof. By chain rule, we write

$$\nabla_\theta R(\pi_\theta) = -\sum_{s,a} \nabla_\theta d^\pi(s, a) \log \pi(a|s) + \sum_{s,a} d^\pi(s, a) \nabla_\theta \log \pi(a|s) = -\sum_{s,a} \nabla_\theta d^\pi(s, a) \log \pi(a|s).$$

The second equation uses the fact that $\mathbb{E}_{x\sim p(\cdot)}[\nabla_\theta \log p(x)] = 0$ for any density p and that $d^\pi(s, a) = d^\pi(s)\pi(a|s)$ as laid out below:

$$\sum_{s,a} d^\pi(s, a) \nabla_\theta \log \pi(a|s) = \sum_s d^\pi(s) \sum_a \pi(a|s) \nabla_\theta \log \pi(a|s) = 0.$$

By another application of chain rule, one can write

$$\nabla_\theta d^\pi(s, a) = \nabla_\theta [d^\pi(s)\pi(a|s)] = \nabla_\theta d^\pi(s)\pi(a|s) + d^\pi(s, a) \nabla_\theta \log \pi(a|s).$$

We further simplify $\nabla_\theta R(\pi_\theta)$ according to

$$\begin{aligned} \nabla_\theta R(\pi_\theta) &= -\sum_{s,a} \nabla_\theta d^\pi(s, a) \log \pi(a|s) \\ &= -\sum_{s,a} \nabla_\theta d^\pi(s)\pi(a|s) \log \pi(a|s) - \sum_{s,a} d^\pi(s, a) \nabla_\theta \log \pi(a|s) \log \pi(a|s). \end{aligned}$$

We substitute $\nabla_\theta d^\pi(s)$ based on Zhang et al. (2021a, Lemma D.1):

$$\begin{aligned}\nabla_\theta R(\pi_\theta) =& -\frac{1}{1-\gamma}\sum_{s',a'} d^\pi(s',a')\nabla_\theta \log(a'|s') \sum_{s,a} d_{s',a'}(s,a)\log\pi(a|s) \\ & - \sum_{s,a} d^\pi(s,a)\nabla_\theta \log\pi(a|s)\log\pi(a|s) \\ =& \mathbb{E}_{(s,a)\sim d_\rho^{\pi_\theta}(\cdot,\cdot)}\left[\nabla_\theta \log\pi(a|s)\left(\frac{1}{1-\gamma}\langle d^\pi_{s,a}, -\log\pi\rangle\right) - \log\pi(a|s)\right],\end{aligned}$$

where $\langle d^\pi_{s,a}, -\log\pi\rangle$ denotes the inner product between vectors $d^\pi_{s,a}$ and $-\log\pi$. This completes the proof. □

The following lemma computes the gradient of MADE regularizer with respect to policy parameters.

Lemma 4.2. *For a policy π parameterized by θ, the gradient of the regularizer*

$$R(\pi_\theta) := \sum_{s,a}\sqrt{d^\pi(s,a)},$$

with respect to θ is given by

$$\nabla_\theta R(\pi_\theta) = \frac{1}{2}\,\mathbb{E}_{(s,a)\sim d^\pi(\cdot,\cdot)}\left[\nabla_\theta \log\pi(a|s)\left(\frac{1}{1-\gamma}\langle d^\pi_{s,a}, \frac{1}{\sqrt{d^\pi}}\rangle + \frac{1}{\sqrt{d^\pi(s,a)}}\right)\right].$$

Proof. The proof is similar to that of Zhang et al. (2021a, Lemma D.3). We write $\nabla_\theta d^\pi(s,a) = \nabla_\theta d^\pi(s)\pi(a|s) + d^\pi(s,a)\nabla_\theta \log\pi(a|s)$ and conclude based on Zhang et al. (2021a, Lemma D.1) that

$$\begin{aligned}\nabla_\theta R(\pi_\theta) =& \frac{1}{2}\sum_{s,a}\frac{\nabla_\theta d^\pi(s,a)}{\sqrt{d^\pi(s,a)}} \\ =& \frac{1}{2(1-\gamma)}\sum_{s',a'} d^\pi(s',a')\nabla_\theta \log\pi(a'|s')\sum_{s,a} d^\pi_{s',a'}(s,a)\frac{1}{\sqrt{d^\pi(s,a)}} \\ & + \frac{1}{2}\sum_{s,a} d^\pi(s,a)\nabla_\theta\log\pi(a|s)\frac{1}{\sqrt{d^\pi(s,a)}} \\ =& \frac{1}{2}\,\mathbb{E}_{(s,a)\sim d^\pi(\cdot,\cdot)}\left[\nabla_\theta \log\pi(a|s)\left(\frac{1}{1-\gamma}\langle d^\pi_{s,a}, \frac{1}{\sqrt{d^\pi}}\rangle + \frac{1}{\sqrt{d^\pi(s,a)}}\right)\right].\end{aligned}$$

□

Chapter 5

Concluding Remarks

Throughout this thesis, we presented formulations for capturing a few of the challenges that AI faces in practice and proposed algorithmic solutions towards addressing these challenges. In addition to some specific future directions pointed out throughout the thesis, in this chapter we review a few important future directions for sequential decision-making and RL, from the author's perspective.

5.1 Foundations

Over the past few years, important progress has been made in theoretical foundations for RL. Foundations can elucidate the data requirements and failure modes of RL, complement empirical evaluations, and inspire new algorithm designs. For instance, hardness results such as sample complexity lower bounds exponential in horizon or feature dimensions have been established (Weisz et al., 2021; Du et al., 2019b; Wang et al., 2021; Foster et al., 2021). These results may explain why RL is significantly more challenging in practice compared to supervised learning and the error compounding observed in some applications such as robotics. Despite the progress, the gap between theory and practice of RL is unresolved. In the rest of this section, we take a look at several major open questions in RL foundations.

5.1.1 Optimism and pessimism with nonlinear function approximation

While information-theoretically near-optimal algorithms for optimism and pessimism principles for online and offline RL are recently proposed in tabular and linear function approximation settings, combining these principles with nonlinear function approximation remains challenging. Several works have considered optimism and pessimism with general function classes (Wang et al., 2020b; Uehara and Sun, 2021), however, the proposed algorithms are often computationally intractable or impractical.

One challenging aspect is the difficulty of constructing accurate confidence bounds, used in the upper and lower confidence bound methods, for nonlinear function classes and simple methods such as ensembles do not offer a competitive performance. This is also a major problem in practice. For instance, Rashid et al. (2020) shows empirical evidence that even with optimistic initialization, Q-functions parameterized by neural networks tend to quickly become "overconfident" about unseen regions, not correctly reflecting the true uncertainty. To alleviate this challenge, one direction is to pursue theoretically-grounded methods for accurate confidence estimation for nonlinear functions. Along these lines, a recent work by Tennenholtz et al. (2021) exploits recent advances in Riemannian geometry and builds confidence intervals over feature space, taking geodesic distance instead of Euclidean distance as a metric.

Another direction is to search for alternatives to confidence bound methods for applying optimism and pessimism principles. In online RL, alternatives to UCB include randomized exploration (Ishfaq et al., 2021), posterior sampling (Osband and Van Roy, 2017), and information-directed exploration (Russo and Van Roy, 2014). In offline RL, recent works Cheng et al. (2022); Zhan et al. (2022) provide promising initial results on alternatives to LCB methods, leaving optimal convergence rates and further empirical evaluations for future work.

5.1.2 Statistical and computational limits

Inspired by the empirical success of RL in large state spaces, a growing body of literature in RL theory has focused on identifying necessary and sufficient conditions on function classes and environment characteristics that admit statistically efficient RL.[1] Such conditions include linear MDPs (Yang and Wang, 2019; Jin et al., 2020b), linear Bellman complete (Zanette et al., 2020), block MDPs (Du et al., 2019a), bounded Eluder dimension (Russo and Van Roy, 2013), Bellman rank (Jiang et al., 2017), Witness rank (Sun et al., 2019), bilinear classes (Du et al., 2021), and Bellman Eluder dimension (Jin et al., 2021). However, computationally efficient algorithms do not exist for the majority of the mentioned settings.

In many of statistical learning problems, the information-theoretic statistical sample complexity is inherently lower than the sample size required by computationally efficient algorithms, which is referred to as the *statistical-computational gap* (Valiant, 1984; Decatur et al., 2000). This phenomena has been observed in a variety of high-dimensional problems such as sparse PCA (Wang et al., 2016), matrix completion (Chen, 2015), and learning neural networks (Mondelli and Montanari, 2019). A series of recent works have theoretically verified that some of the observed gaps are inherent. These works either prove failure of classes of efficient algorithms (Zdeborová and Krzakala, 2016; Barak et al., 2019; Kunisky et al., 2019)

[1] In the context of RL, statistically efficient algorithms are often required a sample size that is independent of state size, polynomial in the horizon, number of actions, and problem complexity measures, and polylogarithmic in number of hypotheses in (or the covering number of) the class (Jiang et al., 2017; Chen and Jiang, 2019)

or give reduction-based arguments relating statistical-computational gaps of problems to one another (Berthet and Rigollet, 2013; Gao et al., 2017).

Statistical-computational gap and computational efficiency limits in RL are major open problems, with the very recent work of Kane et al. (2022) making progress by proving the first computational lower bound for RL with linear function approximation. Further investigation of computational limits alongside statistical limits is an important question to pursue in the future.

5.1.3 Beyond worst-case optimality

Hardness results in RL theory involves constructing worst-case scenarios within a specific set of problems. Closely evaluating the information-theoretically hard examples, the examples appear to be "artificial" and far from real environments. Many RL algorithms tend to perform better in practice compared to what theory suggests and even work in settings that are prohibitive in theory. This gap indicates that the real environments may have more structure and may be in a sense easier than the worst-case scenarios.

An important direction towards theoretical foundations more aligned with practice is conducting *problem-dependent analysis*. This involves defining interesting measures that capture the complexity of RL problem instances and brings about further questions regarding fast rates and adaptive optimality of algorithms. In Chapter 3, we have presented one such measure for offline RL, the single-policy concentrability coefficient, that in a sense captures the offline dataset quality. Other problem-dependent measures appeared in literature include maximum conditional variance (Zanette and Brunskill, 2019) and sub-optimality gap (Simchowitz and Jamieson, 2019; Khamaru et al., 2021). Instance-dependent analysis provide tighter rates and differentiate problems in terms of complexity. Further research in this direction can help narrowing the gap between theory and practice of RL.

5.2 Generalization

Generalization refers to knowing what to do when faced with novel situations, relying on already learned abilities and understanding similarities. The ability to generalize is a core problem in intelligent systems. For the AI to be deployed in the "always changing" real world, it needs to be robust to variations in the environment and able to transfer skills and adapt to unseen environments.

In the context of supervised learning (SL), generalization is fairly understood. In practice, supervised learning achieves decent levels of generalization as demonstrated in the fields of computer vision and natural language processing. Supervised learning, particularly given i.i.d. samples, has a unified theory and provable guarantees which relies on classical complexity notions such as the VC dimension or Rademacher complexity.

Compared to supervised learning, generalization in reinforcement learning is more complex. A clear separation between SL and RL is due to the inability of data reuse in RL, with

an $\exp(H)$ sample size requirement for RL given a hypothesis policy class shown by Kearns et al. (1999). Generalization in RL can take on many forms including generalizing to new observations, new reward functions, new dynamics, and new tasks. Despite progress in theory and practice of RL, many open problems remain in RL generalization. In what follows, we describe a few generalization methods in RL and discuss under-explored directions; for further discussion of open challenges on deep RL generalization, see Kirk et al. (2021).

Representation learning. A key to generalization is obtaining abstractions that extract relevant and meaningful patterns in the complex observational data. While in some cases, a low dimensional feature representation might be known in advance, in the majority of real scenarios such features are unknown. Extracting low-dimensional features can be done via several methods. A common approach is using powerful function approximators such as deep neural networks, in end-to-end training which results in an implicit form of feature learning. Generic deep RL methods exploit this implicit representation learning to obtain a good performance.

Another possibility is conducting dimensionality reduction using an unsupervised pretraining step to explicitly represent complex data with low-dimensional features. In Chapter 2, we constructed features using linear dimensionality reduction. Nonlinear dimensionality reduction methods can further exploit the topological structure of data manifold (Van Der Maaten et al., 2009) and have been effective in many applications. Some examples of successful implementations include EEG classification (Krivov and Belyaev, 2016), source localization in wireless sensor networks (Ghafourian et al., 2020) and single-cell data visualization (Becht et al., 2019). Self-supervised pretraining is another powerful method that offers performance improvements across several modalities such as image, audio, and text (Baevski et al., 2022; Misra and Maaten, 2020; Zbontar et al., 2021; Grill et al., 2020) and also have been recently applied in reinforcement learning (Anand et al., 2019; Srinivas et al., 2020; Touati and Ollivier, 2021; Jiang et al., 2021; Yang and Nachum, 2021).

Despite improving RL performance, most of the current methods are the direct application of unsupervised and self-supervised representation learning framework from conventional vision and language tasks to RL. Designing RL-specific representation learning methods that possibly account for the Bellman equation and controllability constraints as well as understand what objective characterizes good representations in RL remain underexplored. Furthermore, the interplay between representation learning and exploration in online RL or partial dataset coverage in offline RL (Uehara et al., 2021b) are important directions for further research.

Learning world models. There is a significant gap between the generalization power of humans and current state of the artificially intelligent systems. Research in cognitive neuroscience suggests that humans build an abstract model of the world including relevant concepts and relationships (Forrester, 1971; Quiroga et al., 2005; Chang and Tsao, 2017), which can at least partially explain our generalization ability. Building such models gives

predictive ability which in turn affects our perception of the environment, actions, and decisions in different situations (Nortmann et al., 2015; Maus et al., 2013).

Learning world models is a promising method towards more sample-efficient and generalizable RL (Ha and Schmidhuber, 2018) that is currently underexplored compared to model-free algorithms (Kirk et al., 2021). Initial theoretical results have demonstrated complexity gaps between model-based and model-free methods in factored MDPs (Koller and Parr, 1999), linear quadratic regulators (Tu and Recht, 2019), online RL (Sun et al., 2019), and continuous control tasks (Dong et al., 2020). Recent empirical works such as Seo et al. (2020); Anand et al. (2021) suggest that learning a model allows for better generalization of learned abilities to new rewards, observations, and even changing dynamics.

Of course, learning a complete model of the environment can be demanding, especially when dealing with complex sensory inputs such as images. However, such observations from a real environment often contain complex yet irrelevant details. In the RL setting, the cumulative reward objective provides a signal to the agent about parts of the observations that are task-irrelevant and approaches such as value-aware (Farahmand et al., 2017) and policy-aware (Abachi, 2020) model learning aim at learning such partial models of the environment.

Learning abstract and partial models brings about a question: what are considered good abstractions and useful inductive biases to be incoprorated in models? Along these lines, popular topics such as disentangled representations (Bengio et al., 2019; Higgins et al., 2017; Locatello et al., 2019; Chen et al., 2018) and causal learning (Ke et al., 2022; Schölkopf et al., 2021) as well as some less-explored directions such as relational learning (Koller et al., 2007) and (statistical) predicate invention (Kok and Domingos, 2007) (which is the discovery of new concepts and relations from data) are interesting directions for further research.

Bibliography

Romina Abachi. *Policy-aware model learning for policy gradient methods*. PhD thesis, University of Toronto (Canada), 2020.

Yasin Abbasi-Yadkori, Dávid Pál, and Csaba Szepesvári. Improved algorithms for linear stochastic bandits. In *Advances in Neural Information Processing Systems*, pages 2312–2320, 2011.

Marc Abeille and Alessandro Lazaric. Improved regret bounds for Thompson sampling in linear quadratic control problems. In *International Conference on Machine Learning*, pages 1–9, 2018.

Alekh Agarwal, Nan Jiang, and Sham M Kakade. Reinforcement learning: Theory and algorithms. Technical report, Technical Report, Department of Computer Science, University of Washington, 2019a.

Alekh Agarwal, Sham M Kakade, Jason D Lee, and Gaurav Mahajan. On the theory of policy gradient methods: Optimality, approximation, and distribution shift. *arXiv preprint arXiv:1908.00261*, 2019b.

Alekh Agarwal, Mikael Henaff, Sham Kakade, and Wen Sun. PC-PG: Policy cover directed exploration for provable policy gradient learning. *arXiv preprint arXiv:2007.08459*, 2020a.

Alekh Agarwal, Sham Kakade, Akshay Krishnamurthy, and Wen Sun. Flambe: Structural complexity and representation learning of low rank MDPs. *arXiv preprint arXiv:2006.10814*, 2020b.

Alekh Agarwal, Sham Kakade, and Lin F Yang. Model-based reinforcement learning with a generative model is minimax optimal. In *Conference on Learning Theory*, pages 67–83. PMLR, 2020c.

Alekh Agarwal, Sham M Kakade, Jason D Lee, and Gaurav Mahajan. Optimality and approximation with policy gradient methods in Markov decision processes. In *Conference on Learning Theory*, pages 64–66. PMLR, 2020d.

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

Shipra Agrawal and Randy Jia. Optimistic posterior sampling for reinforcement learning: Worst-case regret bounds. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 1184–1194, 2017.

Zafarali Ahmed, Nicolas Le Roux, Mohammad Norouzi, and Dale Schuurmans. Understanding the impact of entropy on policy optimization. In *International Conference on Machine Learning*, pages 151–160. PMLR, 2019.

Ahmed M Alaa, Scott Hu, and Mihaela Schaar. Learning from clinical judgments: Semi-Markov-modulated marked Hawkes processes for risk prognosis. In *International Conference on Machine Learning*, pages 60–69. PMLR, 2017.

Norm Aleks, Stuart J Russell, Michael G Madden, Diane Morabito, Kristan Staudenmayer, Mitchell Cohen, and Geoffrey T Manley. Probabilistic detection of short events, with application to critical care monitoring. In *Advances in Neural Information Processing Systems*, pages 49–56, 2009.

Ankesh Anand, Evan Racah, Sherjil Ozair, Yoshua Bengio, Marc-Alexandre Côté, and R Devon Hjelm. Unsupervised state representation learning in Atari. *arXiv preprint arXiv:1906.08226*, 2019.

Ankesh Anand, Jacob Walker, Yazhe Li, Eszter Vértes, Julian Schrittwieser, Sherjil Ozair, Théophane Weber, and Jessica B Hamrick. Procedural generalization by planning with self-supervised world models. *arXiv preprint arXiv:2111.01587*, 2021.

Marcin Andrychowicz, Filip Wolski, Alex Ray, Jonas Schneider, Rachel Fong, Peter Welinder, Bob McGrew, Josh Tobin, OpenAI Pieter Abbeel, and Wojciech Zaremba. Hindsight experience replay. In *Advances in neural information processing systems*, pages 5048–5058, 2017.

Andras Antos, Rémi Munos, and Csaba Szepesvari. Fitted Q-iteration in continuous action-space mdps. In *Neural Information Processing Systems*, 2007.

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.

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.

Mohammad Gheshlaghi Azar, Rémi Munos, and Hilbert J Kappen. Minimax PAC bounds on the sample complexity of reinforcement learning with a generative model. *Machine learning*, 91(3):325–349, 2013.

Mohammad Gheshlaghi Azar, Ian Osband, and Rémi Munos. Minimax regret bounds for reinforcement learning. In *International Conference on Machine Learning*, pages 263–272. PMLR, 2017.

Adrià Puigdomènech Badia, Pablo Sprechmann, Alex Vitvitskyi, Daniel Guo, Bilal Piot, Steven Kapturowski, Olivier Tieleman, Martín Arjovsky, Alexander Pritzel, Andew Bolt, et al. Never give up: Learning directed exploration strategies. *arXiv preprint arXiv:2002.06038*, 2020.

Alexei Baevski, Wei-Ning Hsu, Qiantong Xu, Arun Babu, Jiatao Gu, and Michael Auli. Data2vec: A general framework for self-supervised learning in speech, vision and language. *arXiv preprint arXiv:2202.03555*, 2022.

Boaz Barak, Samuel Hopkins, Jonathan Kelner, Pravesh K Kothari, Ankur Moitra, and Aaron Potechin. A nearly tight sum-of-squares lower bound for the planted clique problem. *SIAM Journal on Computing*, 48(2):687–735, 2019.

Etienne Becht, Leland McInnes, John Healy, Charles-Antoine Dutertre, Immanuel WH Kwok, Lai Guan Ng, Florent Ginhoux, and Evan W Newell. Dimensionality reduction for visualizing single-cell data using UMAP. *Nature biotechnology*, 37(1):38–44, 2019.

Bernhard Beckermann and Alex Townsend. On the singular values of matrices with displacement structure. *SIAM Journal on Matrix Analysis and Applications*, 38(4):1227–1248, 2017.

David Belanger and Sham Kakade. A linear dynamical system model for text. In *International Conference on Machine Learning*, pages 833–842, 2015.

Marc Bellemare, Sriram Srinivasan, Georg Ostrovski, Tom Schaul, David Saxton, and Remi Munos. Unifying count-based exploration and intrinsic motivation. In *Advances in neural information processing systems*, pages 1471–1479, 2016.

Yoshua Bengio, Tristan Deleu, Nasim Rahaman, Rosemary Ke, Sébastien Lachapelle, Olexa Bilaniuk, Anirudh Goyal, and Christopher Pal. A meta-transfer objective for learning to disentangle causal mechanisms. *arXiv preprint arXiv:1901.10912*, 2019.

Jan Beran. *Statistics for Long-memory Processes*. Routledge, 2017.

Quentin Berthet and Philippe Rigollet. Complexity theoretic lower bounds for sparse principal component detection. In *Conference on learning theory*, pages 1046–1066. PMLR, 2013.

Jalaj Bhandari and Daniel Russo. Global optimality guarantees for policy gradient methods. *arXiv preprint arXiv:1906.01786*, 2019.

Mariusz Bojarski, Davide Del Testa, Daniel Dworakowski, Bernhard Firner, Beat Flepp, Prasoon Goyal, Lawrence D Jackel, Mathew Monfort, Urs Muller, Jiakai Zhang, et al. End to end learning for self-driving cars. *arXiv preprint arXiv:1604.07316*, 2016.

Ronen I Brafman and Moshe Tennenholtz. R-max: A general polynomial time algorithm for near-optimal reinforcement learning. *Journal of Machine Learning Research*, 3(Oct): 213–231, 2002.

Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. Openai gym. *arXiv preprint arXiv:1606.01540*, 2016.

Sébastien Bubeck, Rémi Munos, and Gilles Stoltz. Pure exploration in finitely-armed and continuous-armed bandits. *Theoretical Computer Science*, 412(19):1832–1852, 2011.

Jacob Buckman, Carles Gelada, and Marc G Bellemare. The importance of pessimism in fixed-dataset policy optimization. *arXiv preprint arXiv:2009.06799*, 2020.

Yuri Burda, Harri Edwards, Deepak Pathak, Amos Storkey, Trevor Darrell, and Alexei A Efros. Large-scale study of curiosity-driven learning. *arXiv preprint arXiv:1808.04355*, 2018a.

Yuri Burda, Harrison Edwards, Amos Storkey, and Oleg Klimov. Exploration by random network distillation. *arXiv preprint arXiv:1810.12894*, 2018b.

Qi Cai, Zhuoran Yang, Chi Jin, and Zhaoran Wang. Provably efficient exploration in policy optimization. In *International Conference on Machine Learning*, pages 1283–1294. PMLR, 2020.

Andres Campero, Roberta Raileanu, Heinrich Küttler, Joshua B Tenenbaum, Tim Rocktäschel, and Edward Grefenstette. Learning with AMIGo: Adversarially motivated intrinsic goals. *arXiv preprint arXiv:2006.12122*, 2020.

Shicong Cen, Chen Cheng, Yuxin Chen, Yuting Wei, and Yuejie Chi. Fast global convergence of natural policy gradient methods with entropy regularization. *arXiv preprint arXiv:2007.06558*, 2020.

Le Chang and Doris Y Tsao. The code for facial identity in the primate brain. *Cell*, 169(6): 1013–1028, 2017.

Shaunak Chatterjee and Stuart Russell. Why are DBNs sparse? In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 81–88, 2010.

Jinglin Chen and Nan Jiang. Information-theoretic considerations in batch reinforcement learning. *arXiv preprint arXiv:1905.00360*, 2019.

Ricky TQ Chen, Xuechen Li, Roger B Grosse, and David K Duvenaud. Isolating sources of disentanglement in variational autoencoders. *Advances in neural information processing systems*, 31, 2018.

SY Chen. Kalman filter for robot vision: A survey. *IEEE Transactions on Industrial Electronics*, 59(11):4409–4420, 2011.

Yudong Chen. Incoherence-optimal matrix completion. *IEEE Transactions on Information Theory*, 61(5):2909–2923, 2015.

Ching-An Cheng, Tengyang Xie, Nan Jiang, and Alekh Agarwal. Adversarially trained actor critic for offline reinforcement learning. *arXiv preprint arXiv:2202.02446*, 2022.

Nuttapong Chentanez, Andrew Barto, and Satinder Singh. Intrinsically motivated reinforcement learning. *Advances in neural information processing systems*, 17, 2004.

Maxime Chevalier-Boisvert, Lucas Willems, and Suman Pal. Minimalistic gridworld environment for OpenAI Gym. https://github.com/maximecb/gym-minigrid, 2018.

Cédric Colas, Pierre Fournier, Mohamed Chetouani, Olivier Sigaud, and Pierre-Yves Oudeyer. CURIOUS: Intrinsically motivated modular multi-goal reinforcement learning. In *International conference on machine learning*, pages 1331–1340. PMLR, 2019.

Huseyin Coskun, Felix Achilles, Robert DiPietro, Nassir Navab, and Federico Tombari. Long short-term memory Kalman filters: Recurrent neural estimators for pose regularization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5524–5532, 2017.

Christoph Dann and Emma Brunskill. Sample complexity of episodic fixed-horizon reinforcement learning. In *Proceedings of the 28th International Conference on Neural Information Processing Systems-Volume 2*, pages 2818–2826, 2015.

Peter Dayan and Bernard W Balleine. Reward, motivation, and reinforcement learning. *Neuron*, 36(2):285–298, 2002.

Sarah Dean, Horia Mania, Nikolai Matni, Benjamin Recht, and Stephen Tu. Regret bounds for robust adaptive control of the linear quadratic regulator. In *Advances in Neural Information Processing Systems*, pages 4188–4197, 2018.

Scott E Decatur, Oded Goldreich, and Dana Ron. Computational sample complexity. *SIAM Journal on Computing*, 29(3):854–879, 2000.

Omar Darwiche Domingues, Pierre Ménard, Emilie Kaufmann, and Michal Valko. Episodic reinforcement learning in finite MDPs: Minimax lower bounds revisited. *arXiv preprint arXiv:2010.03531*, 2020.

Kefan Dong, Yuping Luo, Tianhe Yu, Chelsea Finn, and Tengyu Ma. On the expressivity of neural networks for deep reinforcement learning. In *International Conference on Machine Learning*, pages 2627–2637. PMLR, 2020.

Kefan Dong, Jiaqi Yang, and Tengyu Ma. Provable model-based nonlinear bandit and reinforcement learning: Shelve optimism, embrace virtual curvature. *arXiv preprint arXiv:2102.04168*, 2021.

David L Donoho. Compressed sensing. *IEEE Transactions on Information Theory*, 52(4): 1289–1306, 2006.

David L Donoho, Richard C Liu, and Brenda MacGibbon. Minimax risk over hyperrectangles, and implications. *The Annals of Statistics*, pages 1416–1437, 1990.

Paul Doukhan, George Oppenheim, and Murad Taqqu. *Theory and Applications of Long-range Dependence*. Springer Science & Business Media, 2002.

Simon Du, Akshay Krishnamurthy, Nan Jiang, Alekh Agarwal, Miroslav Dudik, and John Langford. Provably efficient RL with rich observations via latent state decoding. In *International Conference on Machine Learning*, pages 1665–1674. PMLR, 2019a.

Simon Du, Sham Kakade, Jason Lee, Shachar Lovett, Gaurav Mahajan, Wen Sun, and Ruosong Wang. Bilinear classes: A structural framework for provable generalization in RL. In *International Conference on Machine Learning*, pages 2826–2836. PMLR, 2021.

Simon S Du, Sham M Kakade, Ruosong Wang, and Lin F Yang. Is a good representation sufficient for sample efficient reinforcement learning? In *International Conference on Learning Representations*, 2019b.

Simon S Du, Sham M Kakade, Ruosong Wang, and Lin F Yang. Is a good representation sufficient for sample efficient reinforcement learning? In *International Conference on Learning Representations*, 2020.

Yaqi Duan, Zeyu Jia, and Mengdi Wang. Minimax-optimal off-policy evaluation with linear function approximation. In *International Conference on Machine Learning*, pages 2701–2709. PMLR, 2020.

Hugh Durrant-Whyte and Tim Bailey. Simultaneous localization and mapping: Part I. *IEEE robotics & automation magazine*, 13(2):99–110, 2006.

Adrien Ecoffet, Joost Huizinga, Joel Lehman, Kenneth O Stanley, and Jeff Clune. Go-explore: A new approach for hard-exploration problems. *arXiv preprint arXiv:1901.10995*, 2019.

Adrien Ecoffet, Joost Huizinga, Joel Lehman, Kenneth O Stanley, and Jeff Clune. First return then explore. *arXiv preprint arXiv:2004.12919*, 2020.

Lasse Espeholt, Hubert Soyer, Remi Munos, Karen Simonyan, Volodymir Mnih, Tom Ward, Yotam Doron, Vlad Firoiu, Tim Harley, Iain Dunning, et al. Impala: Scalable distributed deep-RL with importance weighted actor-learner architectures. *arXiv preprint arXiv:1802.01561*, 2018.

Mohamad Kazem Shirani Faradonbeh, Ambuj Tewari, and George Michailidis. Optimism-based adaptive regulation of linear-quadratic systems. *arXiv preprint arXiv:1711.07230*, 2017.

Mohamad Kazem Shirani Faradonbeh, Ambuj Tewari, and George Michailidis. Finite time identification in unstable linear systems. *Automatica*, 96:342–353, 2018.

Amir Massoud Farahmand, Rémi Munos, and Csaba Szepesvári. Error propagation for approximate policy and value iteration. In *Advances in Neural Information Processing Systems*, 2010.

Amir-massoud Farahmand, Andre Barreto, and Daniel Nikovski. Value-aware loss function for model-based reinforcement learning. In *Artificial Intelligence and Statistics*, pages 1486–1494. PMLR, 2017.

Fei Feng, Ruosong Wang, Wotao Yin, Simon S Du, and Lin Yang. Provably efficient exploration for reinforcement learning using unsupervised learning. *Advances in Neural Information Processing Systems*, 33, 2020.

Fei Feng, Wotao Yin, Alekh Agarwal, and Lin F Yang. Provably correct optimization and exploration with non-linear policies. *arXiv preprint arXiv:2103.11559*, 2021.

Yihao Feng, Lihong Li, and Qiang Liu. A kernel loss for solving the Bellman equation. *arXiv preprint arXiv:1905.10506*, 2019.

Yannis Flet-Berliac, Johan Ferret, Olivier Pietquin, Philippe Preux, and Matthieu Geist. Adversarially guided actor-critic. In *International Conference on Learning Representations*, 2021.

Carlos Florensa, David Held, Xinyang Geng, and Pieter Abbeel. Automatic goal generation for reinforcement learning agents. In *International conference on machine learning*, pages 1515–1528. PMLR, 2018.

Jay W Forrester. Counterintuitive behavior of social systems. *Theory and decision*, 2(2): 109–140, 1971.

Meire Fortunato, Mohammad Gheshlaghi Azar, Bilal Piot, Jacob Menick, Ian Osband, Alex Graves, Vlad Mnih, Remi Munos, Demis Hassabis, Olivier Pietquin, et al. Noisy networks for exploration. *International Conference on Learning Representations*, 2018.

Dylan J Foster, Satyen Kale, Haipeng Luo, Mehryar Mohri, and Karthik Sridharan. Logistic regression: The importance of being improper. *arXiv preprint arXiv:1803.09349*, 2018.

Dylan J Foster, Sham M Kakade, Jian Qian, and Alexander Rakhlin. The statistical complexity of interactive decision making. *arXiv preprint arXiv:2112.13487*, 2021.

Justin Fu, Aviral Kumar, Ofir Nachum, George Tucker, and Sergey Levine. D4RL: Datasets for deep data-driven reinforcement learning. *arXiv preprint arXiv:2004.07219*, 2020.

Scott Fujimoto, Edoardo Conti, Mohammad Ghavamzadeh, and Joelle Pineau. Benchmarking batch deep reinforcement learning algorithms. *arXiv preprint arXiv:1910.01708*, 2019a.

Scott Fujimoto, David Meger, and Doina Precup. Off-policy deep reinforcement learning without exploration. In *International Conference on Machine Learning*, pages 2052–2062. PMLR, 2019b.

Wayne A Fuller and David P Hasza. Predictors for the first-order autoregressive process. *Journal of Econometrics*, 13(2):139–157, 1980.

Wayne A Fuller and David P Hasza. Properties of predictors for autoregressive time series. *Journal of the American Statistical Association*, 76(373):155–161, 1981.

Chao Gao, Zongming Ma, and Harrison H Zhou. Sparse CCA: Adaptive estimation and computational barriers. *The Annals of Statistics*, 45(5):2074–2101, 2017.

Florent Garcin, Boi Faltings, Olivier Donatsch, Ayar Alazzawi, Christophe Bruttin, and Amr Huber. Offline and online evaluation of news recommender systems at swissinfo.ch. In *Proceedings of the 8th ACM Conference on Recommender systems*, pages 169–176, 2014.

Matthieu Geist, Bilal Piot, and Olivier Pietquin. Is the Bellman residual a bad proxy? In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 3208–3217, 2017.

Matthieu Geist, Bruno Scherrer, and Olivier Pietquin. A theory of regularized Markov decision processes. In *International Conference on Machine Learning*, pages 2160–2169. PMLR, 2019.

Amin Ghafourian, Orestis Georgiou, Edmund Barter, and Thilo Gross. Wireless localization with diffusion maps. *Scientific Reports*, 10(1):1–10, 2020.

Udaya Ghai, Holden Lee, Karan Singh, Cyril Zhang, and Yi Zhang. No-regret prediction in marginally stable systems. *arXiv preprint arXiv:2002.02064*, 2020.

Seyed Kamyar Seyed Ghasemipour, Dale Schuurmans, and Shixiang Shane Gu. EMaQ: Expected-max Q-learning operator for simple yet effective offline and online RL. *arXiv preprint arXiv:2007.11091*, 2020.

Marzyeh Ghassemi, Marco Pimentel, Tristan Naumann, Thomas Brennan, David Clifton, Peter Szolovits, and Mengling Feng. A multivariate timeseries modeling approach to severity of illness assessment and forecasting in ICU with sparse, heterogeneous clinical data. In *Proceedings of the AAAI conference on artificial intelligence*, volume 29, 2015.

Edgar N Gilbert. A comparison of signalling alphabets. *The Bell system technical journal*, 31(3):504–522, 1952.

Alexander Goldenshluger and Assaf Zeevi. Nonasymptotic bounds for autoregressive time series modeling. *Annals of Statistics*, pages 417–444, 2001.

Omer Gottesman, Fredrik Johansson, Matthieu Komorowski, Aldo Faisal, David Sontag, Finale Doshi-Velez, and Leo Anthony Celi. Guidelines for reinforcement learning in healthcare. *Nature medicine*, 25(1):16–18, 2019.

Karol Gregor, Danilo Jimenez Rezende, and Daan Wierstra. Variational intrinsic control. *arXiv preprint arXiv:1611.07507*, 2016.

Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent: A new approach to self-supervised learning. *Advances in Neural Information Processing Systems*, 33:21271–21284, 2020.

Caglar Gulcehre, Ziyu Wang, Alexander Novikov, Tom Le Paine, Sergio Gómez Colmenarejo, Konrad Zolna, Rishabh Agarwal, Josh Merel, Daniel Mankowitz, Cosmin Paduraru, et al. RL unplugged: Benchmarks for offline reinforcement learning. *arXiv preprint arXiv:2006.13888*, 2020.

David Ha and Jürgen Schmidhuber. World models. *arXiv preprint arXiv:1803.10122*, 2018.

Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. *arXiv preprint arXiv:1912.01603*, 2019.

Botao Hao, Yaqi Duan, Tor Lattimore, Csaba Szepesvári, and Mengdi Wang. Sparse feature selection makes batch reinforcement learning more sample efficient. *arXiv preprint arXiv:2011.04019*, 2020.

Moritz Hardt, Tengyu Ma, and Benjamin Recht. Gradient descent learns linear dynamical systems. *The Journal of Machine Learning Research*, 19(1):1025–1068, 2018.

Rafael Hasminskii, Ildar Ibragimov, et al. On density estimation in the view of Kolmogorov's ideas in approximation theory. *The Annals of Statistics*, 18(3):999–1010, 1990.

Elad Hazan. Introduction to online convex optimization. *arXiv preprint arXiv:1909.05207*, 2019.

Elad Hazan, Karan Singh, and Cyril Zhang. Learning linear dynamical systems via spectral filtering. In *Advances in Neural Information Processing Systems*, pages 6702–6712, 2017.

Elad Hazan, Holden Lee, Karan Singh, Cyril Zhang, and Yi Zhang. Spectral filtering for general linear dynamical systems. In *Advances in Neural Information Processing Systems*, pages 4634–4643, 2018.

Elad Hazan, Sham Kakade, Karan Singh, and Abby Van Soest. Provably efficient maximum entropy exploration. In *International Conference on Machine Learning*, pages 2681–2691, 2019.

Jiafan He, Dongruo Zhou, and Quanquan Gu. Nearly minimax optimal reinforcement learning for discounted MDPs. *arXiv preprint arXiv:2010.00587*, 2020.

Matteo Hessel, Joseph Modayil, Hado Van Hasselt, Tom Schaul, Georg Ostrovski, Will Dabney, Dan Horgan, Bilal Piot, Mohammad Azar, and David Silver. Rainbow: Combining improvements in deep reinforcement learning. *arXiv preprint arXiv:1710.02298*, 2017.

Irina Higgins, Arka Pal, Andrei Rusu, Loic Matthey, Christopher Burgess, Alexander Pritzel, Matthew Botvinick, Charles Blundell, and Alexander Lerchner. DARLA: Improving zero-shot transfer in reinforcement learning. In *International Conference on Machine Learning*, pages 1480–1490. PMLR, 2017.

Daniel Hsu, Sham Kakade, Tong Zhang, et al. A tail inequality for quadratic forms of subgaussian random vectors. *Electronic Communications in Probability*, 17, 2012.

Andrew Ilyas, Logan Engstrom, Shibani Santurkar, Dimitris Tsipras, Firdaus Janoos, Larry Rudolph, and Aleksander Madry. A closer look at deep policy gradients. In *International Conference on Learning Representations*, 2019.

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.

Riashat Islam, Peter Henderson, Maziar Gomrokchi, and Doina Precup. Reproducibility of benchmarked deep reinforcement learning tasks for continuous control. *International Conference on Learning Representations*, 2017.

Riashat Islam, Zafarali Ahmed, and Doina Precup. Marginalized state distribution entropy regularization in policy optimization. *arXiv preprint arXiv:1912.05128*, 2019.

Thomas Jaksch, Ronald Ortner, and Peter Auer. Near-optimal regret bounds for reinforcement learning. *Journal of Machine Learning Research*, 11(4), 2010.

Natasha Jaques, Asma Ghandeharioun, Judy Hanwen Shen, Craig Ferguson, Agata Lapedriza, Noah Jones, Shixiang Gu, and Rosalind Picard. Way off-policy batch deep reinforcement learning of implicit human preferences in dialog. *arXiv preprint arXiv:1907.00456*, 2019.

Adel Javanmard and Li Zhang. The minimax risk of truncated series estimators for symmetric convex polytopes. In *2012 IEEE International Symposium on Information Theory Proceedings*, pages 1633–1637. IEEE, 2012.

Zeyu Jia, Lin Yang, Csaba Szepesvari, and Mengdi Wang. Model-based reinforcement learning with value-targeted regression. In *Learning for Dynamics and Control*, pages 666–686. PMLR, 2020.

Haoming Jiang, Bo Dai, Mengjiao Yang, Tuo Zhao, and Wei Wei. Towards automatic evaluation of dialog systems: A model-free off-policy evaluation approach. *arXiv preprint arXiv:2102.10242*, 2021.

Nan Jiang. On value functions and the agent-environment boundary. *arXiv preprint arXiv:1905.13341*, 2019.

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.

Jiantao Jiao, Yanjun Han, and Tsachy Weissman. Minimax estimation of the ℓ_1 distance. *IEEE Transactions on Information Theory*, 64(10):6672–6706, 2018.

Chi Jin, Zeyuan Allen-Zhu, Sebastien Bubeck, and Michael I Jordan. Is Q-learning provably efficient? In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pages 4868–4878, 2018.

Chi Jin, Akshay Krishnamurthy, Max Simchowitz, and Tiancheng Yu. Reward-free exploration for reinforcement learning. In *International Conference on Machine Learning*, pages 4870–4879. PMLR, 2020a.

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, 2020b.

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.

Ying Jin, Zhuoran Yang, and Zhaoran Wang. Is pessimism provably efficient for offline RL? *arXiv preprint arXiv:2012.15085*, 2020c.

Thomas Kailath, Ali H Sayed, and Babak Hassibi. *Linear Estimation*. Number BOOK. Prentice Hall, 2000.

Sham Kakade and Peter Dayan. Dopamine: generalization and bonuses. *Neural Networks*, 15(4-6):549–559, 2002.

Sham Kakade and John Langford. Approximately optimal approximate reinforcement learning. In *In Proc. 19th International Conference on Machine Learning*. Citeseer, 2002.

Sham Kakade, Mengdi Wang, and Lin F Yang. Variance reduction methods for sublinear reinforcement learning. *arXiv preprint arXiv:1802.09184*, 2018.

Daniel Kane, Sihan Liu, Shachar Lovett, and Gaurav Mahajan. Computational-statistical gaps in reinforcement learning. *arXiv preprint arXiv:2202.05444*, 2022.

Emilie Kaufmann, Pierre Ménard, Omar Darwiche Domingues, Anders Jonsson, Edouard Leurent, and Michal Valko. Adaptive reward-free exploration. In *Algorithmic Learning Theory*, pages 865–891. PMLR, 2021.

Nan Rosemary Ke, Silvia Chiappa, Jane Wang, Jorg Bornschein, Theophane Weber, Anirudh Goyal, Matthew Botvinic, Michael Mozer, and Danilo Jimenez Rezende. Learning to induce causal structure. *arXiv preprint arXiv:2204.04875*, 2022.

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

Michael Kearns, Yishay Mansour, and Andrew Ng. Approximate planning in large POMDPs via reusable trajectories. *Advances in Neural Information Processing Systems*, 12, 1999.

Koulik Khamaru, Ashwin Pananjady, Feng Ruan, Martin J Wainwright, and Michael I Jordan. Is temporal difference learning optimal? An instance-dependent analysis. *SIAM Journal on Mathematics of Data Science*, 3(4):1013–1040, 2021.

Rahul Kidambi, Aravind Rajeswaran, Praneeth Netrapalli, and Thorsten Joachims. MOReL: Model-based offline reinforcement learning. *arXiv preprint arXiv:2005.05951*, 2020.

Hyoungseok Kim, Jaekyeom Kim, Yeonwoo Jeong, Sergey Levine, and Hyun Oh Song. Emi: Exploration with mutual information. *arXiv preprint arXiv:1810.01176*, 2018.

Robert Kirk, Amy Zhang, Edward Grefenstette, and Tim Rocktäschel. A survey of generalisation in deep reinforcement learning. *arXiv preprint arXiv:2111.09794*, 2021.

Alexander S Klyubin, Daniel Polani, and Chrystopher L Nehaniv. All else being equal be empowered. In *European Conference on Artificial Life*, pages 744–753. Springer, 2005a.

Alexander S Klyubin, Daniel Polani, and Chrystopher L Nehaniv. Empowerment: A universal agent-centric measure of control. In *2005 IEEE Congress on Evolutionary Computation*, volume 1, pages 128–135. IEEE, 2005b.

Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Sara Beery, et al. WILDS: A benchmark of in-the-wild distribution shifts. *arXiv preprint arXiv:2012.07421*, 2020.

Stanley Kok and Pedro Domingos. Statistical predicate invention. In *Proceedings of the 24th international conference on Machine learning*, pages 433–440, 2007.

Daphne Koller and Ronald Parr. Computing factored value functions for policies in structured MDPs. In *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence*, pages 1332–1339, 1999.

Daphne Koller, Nir Friedman, Sašo Džeroski, Charles Sutton, Andrew McCallum, Avi Pfeffer, Pieter Abbeel, Ming-Fai Wong, Chris Meek, Jennifer Neville, et al. *Introduction to statistical relational learning*. MIT press, 2007.

Vijay R Konda and John N Tsitsiklis. Actor-critic algorithms. In *Advances in neural information processing systems*, pages 1008–1014. Citeseer, 2000.

Ilya Kostrikov, Denis Yarats, and Rob Fergus. Image augmentation is all you need: Regularizing deep reinforcement learning from pixels. *arXiv preprint arXiv:2004.13649*, 2020.

Mark Kozdoba, Jakub Marecek, Tigran Tchrakian, and Shie Mannor. Online learning of linear dynamical systems: Exponential forgetting in Kalman filters. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 4098–4105, 2019.

Victor Kozyakin. On accuracy of approximation of the spectral radius by the Gelfand formula. *Linear Algebra and its Applications*, 431(11):2134–2141, 2009.

Akshay Krishnamurthy, Alekh Agarwal, and John Langford. PAC reinforcement learning with rich observations. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, pages 1848–1856, 2016.

Egor Krivov and Mikhail Belyaev. Dimensionality reduction with isomap algorithm for EEG covariance matrices. In *2016 4th International Winter Conference on Brain-Computer Interface (BCI)*, pages 1–4. IEEE, 2016.

Aviral Kumar and Sergey Levine. Offline reinforcement learning: From algorithms to practical challenges. https://sites.google.com/view/offlinerltutorial-neurips2020/home, 2020.

Aviral Kumar, Justin Fu, George Tucker, and Sergey Levine. Stabilizing off-policy Q-learning via bootstrapping error reduction. *arXiv preprint arXiv:1906.00949*, 2019.

Aviral Kumar, Aurick Zhou, George Tucker, and Sergey Levine. Conservative Q-learning for offline reinforcement learning. *arXiv preprint arXiv:2006.04779*, 2020.

Aviral Kumar, Joey Hong, Anikait Singh, and Sergey Levine. When should we prefer offline reinforcement learning over behavioral cloning? *arXiv preprint arXiv:2204.05618*, 2022.

Dmitriy Kunisky, Alexander S Wein, and Afonso S Bandeira. Notes on computational hardness of hypothesis testing: Predictions using the low-degree likelihood ratio. *arXiv preprint arXiv:1907.11636*, 2019.

Vitaly Kuznetsov and Mehryar Mohri. Generalization bounds for non-stationary mixing processes. *Machine Learning*, 106(1):93–117, 2017.

Tze Leung Lai and Zhiliang Ying. Recursive identification and adaptive prediction in linear stochastic systems. *SIAM Journal on Control and Optimization*, 29(5):1061–1090, 1991.

Tze Leung Lai, Ching Zong Wei, et al. Least squares estimates in stochastic regression models with applications to identification and control of dynamic systems. *The Annals of Statistics*, 10(1):154–166, 1982.

Sascha Lange, Thomas Gabel, and Martin Riedmiller. Batch reinforcement learning. In *Reinforcement learning*, pages 45–73. Springer, 2012.

Romain Laroche, Paul Trichelair, and Remi Tachet Des Combes. Safe policy improvement with baseline bootstrapping. In *International Conference on Machine Learning*, pages 3652–3661. PMLR, 2019.

Michael Laskin, Kimin Lee, Adam Stooke, Lerrel Pinto, Pieter Abbeel, and Aravind Srinivas. Reinforcement learning with augmented data. *arXiv preprint arXiv:2004.14990*, 2020.

Tor Lattimore and Marcus Hutter. PAC bounds for discounted MDPs. In *International Conference on Algorithmic Learning Theory*, pages 320–334. Springer, 2012.

Lucien Le Cam. *Asymptotic Methods in Statistical Decision Theory*. Springer Science & Business Media, 2012.

Kimin Lee, Kibok Lee, Jinwoo Shin, and Honglak Lee. Network randomization: A simple technique for generalization in deep reinforcement learning. *arXiv preprint arXiv:1910.05396*, 2019a.

Lisa Lee, Benjamin Eysenbach, Emilio Parisotto, Eric Xing, Sergey Levine, and Ruslan Salakhutdinov. Efficient exploration via state marginal matching. *arXiv preprint arXiv:1906.05274*, 2019b.

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

Gen Li, Yuting Wei, Yuejie Chi, Yuantao Gu, and Yuxin Chen. Breaking the sample size barrier in model-based reinforcement learning with a generative model. *arXiv preprint arXiv:2005.12900*, 2020.

Peng Liao, Zhengling Qi, and Susan Murphy. Batch policy learning in average reward Markov decision processes. *arXiv preprint arXiv:2007.11771*, 2020.

Timothy P Lillicrap, Jonathan J Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*, 2015.

Boyi Liu, Qi Cai, Zhuoran Yang, and Zhaoran Wang. Neural trust region/proximal policy optimization attains globally optimal policy. In *Neural Information Processing Systems*, 2019.

Shuang Liu and Hao Su. Regret bounds for discounted MDPs. *arXiv preprint arXiv:2002.05138*, 2020.

Yao Liu, Adith Swaminathan, Alekh Agarwal, and Emma Brunskill. Provably good batch reinforcement learning without great exploration. *arXiv preprint arXiv:2007.08202*, 2020.

Lennart Ljung. Convergence of an adaptive filter algorithm. *International Journal of Control*, 27(5):673–693, 1978.

Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Raetsch, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. Challenging common assumptions in the unsupervised learning of disentangled representations. In *international conference on machine learning*, pages 4114–4124. PMLR, 2019.

George G Lorentz, Manfred von Golitschek, and Yuly Makovoz. *Constructive Approximation: Advanced Problems*, volume 304. Springer, 1996.

Cong Ma, Banghua Zhu, Jiantao Jiao, and Martin J Wainwright. Minimax off-policy evaluation for multi-armed bandits. *arXiv preprint arXiv:2101.07781*, 2021.

Zongming Ma and Yihong Wu. Volume ratio, sparsity, and minimaxity under unitarily invariant norms. *IEEE Transactions on Information Theory*, 61(12):6939–6956, 2015.

Horia Mania, Stephen Tu, and Benjamin Recht. Certainty equivalent control of LQR is efficient. *arXiv preprint arXiv:1902.07826*, 2019.

Kenneth Marino, Abhinav Gupta, Rob Fergus, and Arthur Szlam. Hierarchical RL using an ensemble of proprioceptive periodic policies. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=SJz1x20cFQ.

Gerrit W Maus, Jason Fischer, and David Whitney. Motion-dependent representation of space in area MT+. *Neuron*, 78(3):554–562, 2013.

Jincheng Mei, Chenjun Xiao, Csaba Szepesvari, and Dale Schuurmans. On the global convergence rates of softmax policy gradient methods. In *International Conference on Machine Learning*, pages 6820–6829. PMLR, 2020.

Jincheng Mei, Yue Gao, Bo Dai, Csaba Szepesvari, and Dale Schuurmans. Leveraging non-uniformity in first-order non-convex optimization. *arXiv preprint arXiv:2105.06072*, 2021.

Pierre Ménard, Omar Darwiche Domingues, Anders Jonsson, Emilie Kaufmann, Edouard Leurent, and Michal Valko. Fast active learning for pure exploration in reinforcement learning. *arXiv preprint arXiv:2007.13442*, 2020.

Pierre Menard, Omar Darwiche Domingues, Xuedong Shang, and Michal Valko. UCB momentum Q-learning: Correcting the bias without forgetting. *arXiv preprint arXiv:2103.01312*, 2021.

Shahar Mendelson. Learning without concentration. In *Conference on Learning Theory*, pages 25–39, 2014.

John Miller and Moritz Hardt. Stable recurrent models. *arXiv preprint arXiv:1805.10369*, 2018.

Dipendra Misra, Mikael Henaff, Akshay Krishnamurthy, and John Langford. Kinematic state abstraction and provably efficient rich-observation reinforcement learning. In *International conference on machine learning*, pages 6961–6971. PMLR, 2020.

Ishan Misra and Laurens van der Maaten. Self-supervised learning of pretext-invariant representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6707–6717, 2020.

Michael Mitzenmacher and Eli Upfal. *Probability and computing: Randomization and probabilistic techniques in algorithms and data analysis*. Cambridge university press, 2017.

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.

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.

Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. Asynchronous methods for deep reinforcement learning. In *International conference on machine learning*, pages 1928–1937, 2016.

Shakir Mohamed and Danilo Jimenez Rezende. Variational information maximisation for intrinsically motivated reinforcement learning. In *Advances in neural information processing systems*, pages 2125–2133, 2015.

Mehryar Mohri and Afshin Rostamizadeh. Rademacher complexity bounds for non-iid processes. In *Advances in Neural Information Processing Systems*, pages 1097–1104, 2009.

Marco Mondelli and Andrea Montanari. On the connection between learning two-layer neural networks and tensor decomposition. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 1051–1060. PMLR, 2019.

Matej Moravčík, Martin Schmid, Neil Burch, Viliam Lisỳ, Dustin Morrill, Nolan Bard, Trevor Davis, Kevin Waugh, Michael Johanson, and Michael Bowling. Deepstack: Expert-level artificial intelligence in heads-up no-limit poker. *Science*, 356(6337):508–513, 2017.

Rémi Munos. Error bounds for approximate policy iteration. In *Proceedings of the Twentieth International Conference on International Conference on Machine Learning*, pages 560–567, 2003.

Rémi Munos. Performance bounds in ℓ_p-norm for approximate value iteration. *SIAM journal on control and optimization*, 46(2):541–561, 2007.

Ofir Nachum and Bo Dai. Reinforcement learning via Fenchel-Rockafellar duality. *arXiv preprint arXiv:2001.01866*, 2020.

Ofir Nachum, Mohammad Norouzi, Kelvin Xu, and Dale Schuurmans. Bridging the gap between value and policy based reinforcement learning. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 2772–2782, 2017.

Ofir Nachum, Yinlam Chow, Bo Dai, and Lihong Li. Dualdice: Behavior-agnostic estimation of discounted stationary distribution corrections. In *Advances in Neural Information Processing Systems*, pages 2315–2325, 2019a.

Ofir Nachum, Bo Dai, Ilya Kostrikov, Yinlam Chow, Lihong Li, and Dale Schuurmans. AlgaeDICE: Policy gradient from arbitrary experience. *arXiv preprint arXiv:1912.02074*, 2019b.

Kimia Nadjahi, Romain Laroche, and Rémi Tachet des Combes. Safe policy improvement with soft baseline bootstrapping. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 53–68. Springer, 2019.

Ashvin V Nair, Vitchyr Pong, Murtaza Dalal, Shikhar Bahl, Steven Lin, and Sergey Levine. Visual reinforcement learning with imagined goals. *Advances in Neural Information Processing Systems*, 31:9191–9200, 2018.

Gergely Neu, Anders Jonsson, and Vicenç Gómez. A unified view of entropy-regularized Markov decision processes. *arXiv preprint arXiv:1705.07798*, 2017.

Xinkun Nie, Emma Brunskill, and Stefan Wager. Learning when-to-treat policies. *Journal of the American Statistical Association*, pages 1–18, 2020.

Nikolay Nikolov, Johannes Kirschner, Felix Berkenkamp, and Andreas Krause. Information-directed exploration for deep reinforcement learning. *arXiv preprint arXiv:1812.07544*, 2018.

Uri Nodelman, Christian R Shelton, and Daphne Koller. Continuous time Bayesian networks. In *Proceedings of the Eighteenth conference on Uncertainty in artificial intelligence*, pages 378–387, 2002.

Nora Nortmann, Sascha Rekauzke, Selim Onat, Peter König, and Dirk Jancke. Primary visual cortex represents the difference between past and present. *Cerebral Cortex*, 25(6): 1427–1440, 2015.

Ian Osband and Benjamin Van Roy. Why is posterior sampling better than optimism for reinforcement learning? In *International conference on machine learning*, pages 2701–2710. PMLR, 2017.

Ian Osband, Charles Blundell, Alexander Pritzel, and Benjamin Van Roy. Deep exploration via bootstrapped DQN. In *Advances in neural information processing systems*, pages 4026–4034, 2016.

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

Georg Ostrovski, Marc G Bellemare, Aaron van den Oord, and Rémi Munos. Count-based exploration with neural density models. *arXiv preprint arXiv:1703.01310*, 2017.

Yi Ouyang, Mukul Gagrani, and Rahul Jain. Learning-based control of unknown linear systems with Thompson sampling. *arXiv preprint arXiv:1709.04047*, 2017.

Yunpeng Pan, Ching-An Cheng, Kamil Saigol, Keuntaek Lee, Xinyan Yan, Evangelos Theodorou, and Byron Boots. Agile autonomous driving using end-to-end deep imitation learning. *arXiv preprint arXiv:1709.07174*, 2017.

Robert S Parker, Francis J Doyle, and Nicholas A Peppas. A model-based algorithm for blood glucose control in type I diabetic patients. *IEEE Transactions on biomedical engineering*, 46(2):148–157, 1999.

Deepak Pathak, Pulkit Agrawal, Alexei A Efros, and Trevor Darrell. Curiosity-driven exploration by self-supervised prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 16–17, 2017.

Deepak Pathak, Dhiraj Gandhi, and Abhinav Gupta. Self-supervised exploration via disagreement. *arXiv preprint arXiv:1906.04161*, 2019.

Xue Bin Peng, Aviral Kumar, Grace Zhang, and Sergey Levine. Advantage-weighted regression: Simple and scalable off-policy reinforcement learning. *arXiv preprint arXiv:1910.00177*, 2019.

Allan Pinkus. *N-widths in Approximation Theory*, volume 7. Springer Science & Business Media, 2012.

Matthias Plappert, Rein Houthooft, Prafulla Dhariwal, Szymon Sidor, Richard Y Chen, Xi Chen, Tamim Asfour, Pieter Abbeel, and Marcin Andrychowicz. Parameter space noise for exploration. *International Conference on Learning Representations*, 2018.

Vitchyr Pong, Murtaza Dalal, Steven Lin, Ashvin Nair, Shikhar Bahl, and Sergey Levine. Skew-Fit: State-covering self-supervised reinforcement learning. In *International Conference on Machine Learning*, pages 7783–7792. PMLR, 2020.

Martin L Puterman. Markov decision processes. *Handbooks in operations research and management science*, 2:331–434, 1990.

R Quian Quiroga, Leila Reddy, Gabriel Kreiman, Christof Koch, and Itzhak Fried. Invariant visual representation by single neurons in the human brain. *Nature*, 435(7045):1102–1107, 2005.

Roberta Raileanu and Tim Rocktäschel. Ride: Rewarding impact-driven exploration for procedurally-generated environments. *arXiv preprint arXiv:2002.12292*, 2020.

Nived Rajaraman, Lin F Yang, Jiantao Jiao, and Kannan Ramachandran. Toward the fundamental limits of imitation learning. *arXiv preprint arXiv:2009.05990*, 2020.

Tabish Rashid, Bei Peng, Wendelin Boehmer, and Shimon Whiteson. Optimistic exploration even with a pessimistic initialisation. *arXiv preprint arXiv:2002.12174*, 2020.

Paria Rashidinejad, Xiao Hu, and Stuart Russell. Patient-adaptable intracranial pressure morphology analysis using a probabilistic model-based approach. *Physiological Measurement*, 41(10):104003, 2020a.

Paria Rashidinejad, Navaneeth Jamadagni, Arun Raghavan, Craig Schelp, and Charles Gordon. Techniques for accurately estimating the reliability of storage systems, November 26 2020b. US Patent App. 15/930,779.

Paria Rashidinejad, Jiantao Jiao, and Stuart Russell. SLIP: Learning to predict in unknown dynamical systems with long-term memory. *Advances in Neural Information Processing Systems*, 33:5716–5728, 2020c.

Paria Rashidinejad, Banghua Zhu, Cong Ma, Jiantao Jiao, and Stuart Russell. Bridging offline reinforcement learning and imitation learning: A tale of pessimism. *Advances in Neural Information Processing Systems*, 34, 2021.

B Ash. Robert. Information theory, 1990.

Stéphane Ross and Drew Bagnell. Efficient reductions for imitation learning. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 661–668, 2010.

Daniel Russo. Worst-case regret bounds for exploration via randomized value functions. *arXiv preprint arXiv:1906.02870*, 2019.

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

Daniel Russo and Benjamin Van Roy. Learning to optimize via information-directed sampling. *Advances in Neural Information Processing Systems*, 27, 2014.

Christoph Salge, Cornelius Glackin, and Daniel Polani. Empowerment: An introduction. In *Guided Self-Organization: Inception*, pages 67–114. Springer, 2014.

Tim Salimans and Richard Chen. Learning Montezuma's revenge from a single demonstration. *arXiv preprint arXiv:1812.03381*, 2018.

Tuhin Sarkar and Alexander Rakhlin. Near optimal finite time identification of arbitrary linear dynamical systems. *arXiv preprint arXiv:1812.01251*, 2018.

Bruno Scherrer. Should one compute the temporal difference fix point or minimize the bellman residual? The unified oblique projection view. *arXiv preprint arXiv:1011.4362*, 2010.

Bruno Scherrer. Approximate policy iteration schemes: A comparison. In *International Conference on Machine Learning*, pages 1314–1322. PMLR, 2014.

Bernhard Schölkopf, Francesco Locatello, Stefan Bauer, Nan Rosemary Ke, Nal Kalchbrenner, Anirudh Goyal, and Yoshua Bengio. Toward causal representation learning. *Proceedings of the IEEE*, 109(5):612–634, 2021.

Younggyo Seo, Kimin Lee, Ignasi Clavera Gilaberte, Thanard Kurutach, Jinwoo Shin, and Pieter Abbeel. Trajectory-wise multiple choice learning for dynamics generalization in reinforcement learning. *Advances in Neural Information Processing Systems*, 33:12968–12979, 2020.

Younggyo Seo, Lili Chen, Jinwoo Shin, Honglak Lee, Pieter Abbeel, and Kimin Lee. State entropy maximization with random encoders for efficient exploration. *arXiv preprint arXiv:2102.09430*, 2021.

Pranav Shyam, Wojciech Jaśkowski, and Faustino Gomez. Model-based active exploration. In *International Conference on Machine Learning*, pages 5779–5788, 2019.

Aaron Sidford, Mengdi Wang, Xian Wu, Lin F Yang, and Yinyu Ye. Near-optimal time and sample complexities for solving discounted Markov decision process with a generative model. *arXiv preprint arXiv:1806.01492*, 2018a.

Aaron Sidford, Mengdi Wang, Xian Wu, and Yinyu Ye. Variance reduced value iteration and faster algorithms for solving Markov decision processes. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 770–787. SIAM, 2018b.

Noah Y Siegel, Jost Tobias Springenberg, Felix Berkenkamp, Abbas Abdolmaleki, Michael Neunert, Thomas Lampe, Roland Hafner, and Martin Riedmiller. Keep doing what worked: Behavioral modelling priors for offline reinforcement learning. *arXiv preprint arXiv:2002.08396*, 2020.

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.

David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, et al. Mastering the game of Go without human knowledge. *nature*, 550(7676):354–359, 2017.

David Silver, Satinder Singh, Doina Precup, and Richard S Sutton. Reward is enough. *Artificial Intelligence*, 299:103535, 2021.

Max Simchowitz and Dylan J Foster. Naive exploration is optimal for online LQR. *arXiv preprint arXiv:2001.09576*, 2020.

Max Simchowitz and Kevin G Jamieson. Non-asymptotic gap-dependent regret bounds for tabular MDPs. *Advances in Neural Information Processing Systems*, 32, 2019.

Max Simchowitz, Horia Mania, Stephen Tu, Michael I Jordan, and Benjamin Recht. Learning without mixing: towards a sharp analysis of linear system identification. *Proceedings of Machine Learning Research*, 75:1–35, 2018.

Max Simchowitz, Ross Boczar, and Benjamin Recht. Learning linear dynamical systems with semi-parametric least squares. *arXiv preprint arXiv:1902.00768*, 2019.

Aravind Srinivas, Michael Laskin, and Pieter Abbeel. CURL: Contrastive unsupervised representations for reinforcement learning. *arXiv preprint arXiv:2004.04136*, 2020.

Alex Strehl, John Langford, Sham Kakade, and Lihong Li. Learning from logged implicit exploration data. *arXiv preprint arXiv:1003.0120*, 2010.

Alexander L Strehl, Lihong Li, Eric Wiewiora, John Langford, and Michael L Littman. PAC model-free reinforcement learning. In *Proceedings of the 23rd international conference on Machine learning*, pages 881–888, 2006.

Wen Sun, Nan Jiang, Akshay Krishnamurthy, Alekh Agarwal, and John Langford. Model-based RL in contextual decision processes: PAC bounds and exponential improvements over model-free approaches. In *Conference on Learning Theory*, pages 2898–2933. PMLR, 2019.

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

Richard S Sutton, David McAllester, Satinder Singh, and Yishay Mansour. Policy gradient methods for reinforcement learning with function approximation. In *Proceedings of the 12th International Conference on Neural Information Processing Systems*, pages 1057–1063, 1999.

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

Csaba Szepesvári and Rémi Munos. Finite time bounds for sampling based fitted value iteration. In *Proceedings of the 22nd international conference on Machine learning*, pages 880–887, 2005.

Haoran Tang, Rein Houthooft, Davis Foote, Adam Stooke, OpenAI Xi Chen, Yan Duan, John Schulman, Filip DeTurck, and Pieter Abbeel. #Exploration: A study of count-based exploration for deep reinforcement learning. In *Advances in neural information processing systems*, pages 2753–2762, 2017.

Yuval Tassa, Saran Tunyasuvunakool, Alistair Muldal, Yotam Doron, Piotr Trochim, Siqi Liu, Steven Bohez, Josh Merel, Tom Erez, Timothy Lillicrap, et al. dm_control: Software and tasks for continuous control. *arXiv preprint arXiv:2006.12983*, 2020.

Guy Tennenholtz, Nir Baram, and Shie Mannor. Latent geodesics of model dynamics for offline reinforcement learning. In *Deep RL Workshop NeurIPS 2021*, 2021.

Philip S Thomas, Georgios Theocharous, Mohammad Ghavamzadeh, Ishan Durugkar, and Emma Brunskill. Predictive off-policy policy evaluation for nonstationary decision problems, with applications to digital marketing. In *AAAI*, pages 4740–4745, 2017.

Ahmed Touati and Yann Ollivier. Learning one representation to optimize all rewards. *arXiv preprint arXiv:2103.07945*, 2021.

Anastasios Tsiamis and George Pappas. Online learning of the Kalman filter with logarithmic regret. *arXiv preprint arXiv:2002.05141*, 2020.

Anastasios Tsiamis and George J Pappas. Finite sample analysis of stochastic system identification. *arXiv preprint arXiv:1903.09122*, 2019.

Stephen Tu and Benjamin Recht. The gap between model-based and model-free methods on the linear quadratic regulator: An asymptotic viewpoint. In *Conference on Learning Theory*, pages 3036–3083. PMLR, 2019.

Masatoshi Uehara and Wen Sun. Pessimistic model-based offline reinforcement learning under partial coverage. *arXiv preprint arXiv:2107.06226*, 2021.

Masatoshi Uehara, Jiawei Huang, and Nan Jiang. Minimax weight and Q-function learning for off-policy evaluation. In *International Conference on Machine Learning*, pages 9659–9668. PMLR, 2020.

Masatoshi Uehara, Masaaki Imaizumi, Nan Jiang, Nathan Kallus, Wen Sun, and Tengyang Xie. Finite sample analysis of minimax offline reinforcement learning: Completeness, fast rates and first-order efficiency. *arXiv preprint arXiv:2102.02981*, 2021a.

Masatoshi Uehara, Xuezhou Zhang, and Wen Sun. Representation learning for online and offline RL in low-rank MDPs. *arXiv preprint arXiv:2110.04652*, 2021b.

Leslie G Valiant. A theory of the learnable. *Communications of the ACM*, 27(11):1134–1142, 1984.

Laurens Van Der Maaten, Eric Postma, Jaap Van den Herik, et al. Dimensionality reduction: A comparative. *J Mach Learn Res*, 10(66-71):13, 2009.

Rom Rubenovich Varshamov. Estimate of the number of signals in error correcting codes. *Docklady Akad. Nauk, SSSR*, 117:739–741, 1957.

Roman Vershynin. *High-Dimensional Probability: An Introduction with Applications in Data Science*, volume 47. Cambridge University Press, 2018.

Oriol Vinyals, Igor Babuschkin, Wojciech M Czarnecki, Michaël Mathieu, Andrew Dudzik, Junyoung Chung, David H Choi, Richard Powell, Timo Ewalds, Petko Georgiev, et al. Grandmaster level in StarCraft II using multi-agent reinforcement learning. *Nature*, 575 (7782):350–354, 2019.

Lingxiao Wang, Qi Cai, Zhuoran Yang, and Zhaoran Wang. Neural policy gradient methods: Global optimality and rates of convergence. In *International Conference on Learning Representations*, 2019a.

Lu Wang, Wei Zhang, Xiaofeng He, and Hongyuan Zha. Supervised reinforcement learning with recurrent neural network for dynamic treatment recommendation. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2447–2456, 2018.

Ruosong Wang, Dean P Foster, and Sham M Kakade. What are the statistical limits of offline rl with linear function approximation? *arXiv preprint arXiv:2010.11895*, 2020a.

Ruosong Wang, Russ R Salakhutdinov, and Lin Yang. Reinforcement learning with general value function approximation: Provably efficient approach via bounded Eluder dimension. *Advances in Neural Information Processing Systems*, 33, 2020b.

Tengyao Wang, Quentin Berthet, and Richard J Samworth. Statistical and computational trade-offs in estimation of sparse principal components. *The Annals of Statistics*, 44(5): 1896–1930, 2016.

Yining Wang, Ruosong Wang, Simon S Du, and Akshay Krishnamurthy. Optimism in reinforcement learning with generalized linear function approximation. *arXiv preprint arXiv:1912.04136*, 2019b.

Yuanhao Wang, Kefan Dong, Xiaoyu Chen, and Liwei Wang. Q-learning with UCB exploration is sample efficient for infinite-horizon MDP. In *International Conference on Learning Representations*, 2019c.

Yuanhao Wang, Ruosong Wang, and Sham Kakade. An exponential lower bound for linearly realizable mdp with constant suboptimality gap. *Advances in Neural Information Processing Systems*, 34, 2021.

Ziyu Wang, Alexander Novikov, Konrad Zolna, Josh S Merel, Jost Tobias Springenberg, Scott E Reed, Bobak Shahriari, Noah Siegel, Caglar Gulcehre, Nicolas Heess, et al. Critic regularized regression. *Advances in Neural Information Processing Systems*, 33, 2020c.

CZ Wei. Adaptive prediction by least squares predictors in stochastic regression models with applications to time series. *The Annals of Statistics*, pages 1667–1682, 1987.

Yuting Wei and Martin J Wainwright. The local geometry of testing in ellipses: Tight control via localized Kolmogorov widths. *IEEE Transactions on Information Theory*, 2020.

Yuting Wei, Billy Fang, Martin J Wainwright, et al. From Gauss to Kolmogorov: Localized measures of complexity for ellipses. *Electronic Journal of Statistics*, 14(2):2988–3031, 2020.

Gellért Weisz, Philip Amortila, and Csaba Szepesvári. Exponential lower bounds for planning in MDPs with linearly-realizable optimal action-value functions. In *Algorithmic Learning Theory*, pages 1237–1264. PMLR, 2021.

Jenna Wiens, Suchi Saria, Mark Sendak, Marzyeh Ghassemi, Vincent X Liu, Finale Doshi-Velez, Kenneth Jung, Katherine Heller, David Kale, Mohammed Saeed, et al. Do no harm: A roadmap for responsible machine learning for health care. *Nature medicine*, 25 (9):1337–1340, 2019.

Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4):229–256, 1992.

Ronald J Williams and Jing Peng. Function optimization using connectionist reinforcement learning algorithms. *Connection Science*, 3(3):241–268, 1991.

Yifan Wu, George Tucker, and Ofir Nachum. Behavior regularized offline reinforcement learning. *arXiv preprint arXiv:1911.11361*, 2019.

Tengyang Xie and Nan Jiang. Batch value-function approximation with only realizability. *arXiv preprint arXiv:2008.04990*, 2020.

Tengyang Xie, Nan Jiang, Huan Wang, Caiming Xiong, and Yu Bai. Policy finetuning: Bridging sample-efficient offline and online reinforcement learning. *Advances in neural information processing systems*, 34, 2021.

Zhihan Xiong, Ruoqi Shen, and Simon S Du. Randomized exploration is near-optimal for tabular MDP. *arXiv preprint arXiv:2102.09703*, 2021.

Tian Xu, Ziniu Li, and Yang Yu. Error bounds of imitating policies and environments. *Advances in Neural Information Processing Systems*, 33, 2020.

Lin Yang and Mengdi Wang. Sample-optimal parametric Q-learning using linearly additive features. In *International Conference on Machine Learning*, pages 6995–7004. PMLR, 2019.

Mengjiao Yang and Ofir Nachum. Representation matters: Offline pretraining for sequential decision making. *arXiv preprint arXiv:2102.05815*, 2021.

Zhuoran Yang, Chi Jin, Zhaoran Wang, Mengdi Wang, and Michael I Jordan. Bridging exploration and general function approximation in reinforcement learning: Provably efficient kernel and neural value iterations. *arXiv preprint arXiv:2011.04622*, 2020a.

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. *arXiv preprint arXiv:2011.04622*, 2020b.

Denis Yarats, Amy Zhang, Ilya Kostrikov, Brandon Amos, Joelle Pineau, and Rob Fergus. Improving sample efficiency in model-free reinforcement learning from images. *arXiv preprint arXiv:1910.01741*, 2019.

Denis Yarats, Rob Fergus, Alessandro Lazaric, and Lerrel Pinto. Reinforcement learning with prototypical representations. *arXiv preprint arXiv:2102.11271*, 2021.

Ming Yin, Yu Bai, and Yu-Xiang Wang. Near optimal provable uniform convergence in off-policy evaluation for reinforcement learning. *arXiv preprint arXiv:2007.03760*, 2020.

Ming Yin, Yu Bai, and Yu-Xiang Wang. Near-optimal offline reinforcement learning via double variance reduction. *arXiv preprint arXiv:2102.01748*, 2021.

Bin Yu. Rates of convergence for empirical processes of stationary mixing sequences. *The Annals of Probability*, pages 94–116, 1994.

Bin Yu. Assouad, fano, and le cam. In *Festschrift for Lucien Le Cam*, pages 423–435. Springer, 1997.

Chengpu Yu, Lennart Ljung, and Michel Verhaegen. Identification of structured state-space models. *Automatica*, 90:54–61, 2018.

Tianhe Yu, Garrett Thomas, Lantao Yu, Stefano Ermon, James Zou, Sergey Levine, Chelsea Finn, and Tengyu Ma. MOPO: Model-based offline policy optimization. *arXiv preprint arXiv:2005.13239*, 2020.

Tianhe Yu, Aviral Kumar, Rafael Rafailov, Aravind Rajeswaran, Sergey Levine, and Chelsea Finn. COMBO: Conservative offline model-based policy optimization. *arXiv preprint arXiv:2102.08363*, 2021.

Ekim Yurtsever, Jacob Lambert, Alexander Carballo, and Kazuya Takeda. A survey of autonomous driving: Common practices and emerging technologies. *IEEE Access*, 8: 58443–58469, 2020.

Andrea Zanette. Exponential lower bounds for batch reinforcement learning: Batch RL can be exponentially harder than online RL. *arXiv preprint arXiv:2012.08005*, 2020.

Andrea Zanette and Emma Brunskill. Tighter problem-dependent regret bounds in reinforcement learning without domain knowledge using value function bounds. In *International Conference on Machine Learning*, pages 7304–7312. PMLR, 2019.

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.

Andrea Zanette, Ching-An Cheng, and Alekh Agarwal. Cautiously optimistic policy optimization and exploration with linear function approximation. *arXiv preprint arXiv:2103.12923*, 2021.

Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction. In *International Conference on Machine Learning*, pages 12310–12320. PMLR, 2021.

Lenka Zdeborová and Florent Krzakala. Statistical physics of inference: Thresholds and algorithms. *Advances in Physics*, 65(5):453–552, 2016.

Wenhao Zhan, Baihe Huang, Audrey Huang, Nan Jiang, and Jason D Lee. Offline reinforcement learning with realizability and single-policy concentrability. *arXiv preprint arXiv:2202.04634*, 2022.

Chuheng Zhang, Yuanying Cai, and Longbo Huang Jian Li. Exploration by maximizing Rényi entropy for reward-free RL framework. *Proceedings of the AAAI Conference on Artificial Intelligence*, 2021a.

Fuzhen Zhang. *The Schur complement and its applications*, volume 4. Springer Science & Business Media, 2006.

Jingwei Zhang, Niklas Wetzel, Nicolai Dorka, Joschka Boedecker, and Wolfram Burgard. Scheduled intrinsic drive: A hierarchical take on intrinsically motivated exploration. *arXiv preprint arXiv:1903.07400*, 2019.

Junyu Zhang, Alec Koppel, Amrit Singh Bedi, Csaba Szepesvari, and Mengdi Wang. Variational policy gradient method for reinforcement learning with general utilities. *arXiv preprint arXiv:2007.02151*, 2020a.

Ruiyi Zhang, Bo Dai, Lihong Li, and Dale Schuurmans. GenDICE: Generalized offline estimation of stationary values. In *International Conference on Learning Representations*, 2020b.

Shangtong Zhang and Richard S Sutton. A deeper look at experience replay. *arXiv preprint arXiv:1712.01275*, 2017.

Shantong Zhang, Bo Liu, and Shimon Whiteson. GradientDICE: Rethinking generalized offline estimation of stationary values. *arXiv preprint arXiv:2001.11113*, 2020c.

Tianjun Zhang, Huazhe Xu, Xiaolong Wang, Yi Wu, Kurt Keutzer, Joseph E Gonzalez, and Yuandong Tian. BeBold: Exploration beyond the boundary of explored regions. *arXiv preprint arXiv:2012.08621*, 2020d.

Tianjun Zhang, Paria Rashidinejad, Jiantao Jiao, Yuandong Tian, Joseph E Gonzalez, and Stuart Russell. MADE: Exploration via maximizing deviation from explored regions. *Advances in Neural Information Processing Systems*, 34, 2021b.

Zihan Zhang, Xiangyang Ji, and Simon S Du. Is reinforcement learning more difficult than bandits? A near-optimal algorithm escaping the curse of horizon. *arXiv preprint arXiv:2009.13503*, 2020e.

Zihan Zhang, Yuan Zhou, and Xiangyang Ji. Almost optimal model-free reinforcement learning via reference-advantage decomposition. *Advances in Neural Information Processing Systems*, 33, 2020f.

Dongruo Zhou, Quanquan Gu, and Csaba Szepesvari. Nearly minimax optimal reinforcement learning for linear mixture Markov decision processes. *arXiv preprint arXiv:2012.08507*, 2020a.

Dongruo Zhou, Jiafan He, and Quanquan Gu. Provably efficient reinforcement learning for discounted MDPs with feature mapping. *arXiv preprint arXiv:2006.13165*, 2020b.