ROBUST DECISION-MAKING UNDER DATA CORRUPTION

by

Yiding Chen

A dissertation submitted in partial fulfillment of
the requirements for the degree of

Doctor of Philosophy

(Computer Sciences)

at the

UNIVERSITY OF WISCONSIN–MADISON

2023

Date of final oral examination: 12/19/2023

The dissertation is approved by the following members of the Final Oral Committee:
Yudong Chen, Associate Professor, Computer Sciences
Kirthevasan Kandasamy, Assistant Professor, Computer Sciences
Qiaomin Xie, Assistant Professor, Industrial and Systems Engineering
Xiaojin Zhu, Professor, Computer Sciences

To my parents, Jianzhang Chen and Xuefeng Ding

CONTENTS

LIST OF TABLES

LIST OF FIGURES

1 INTRODUCTION

In this thesis, we study robust decision-making under data corruption. In its most general setting, the learner seeks to determine a "good" decision, which can be a proficient policy in reinforcement learning or a nearly-optimal strategy in a zero-sum game. Operating within an unknown environment, the learner relies on data for insights. However, unlike conventional learning settings, a formidable adversary is introduced, capable of manipulating a portion of the dataset, compelling the learner to grapple with a compromised information source.

Data corruption introduces a formidable threat to sequential decision-making processes. The adversary's capacity to manipulate data opens avenues for misleading the agent into making suboptimal decisions, particularly in security-sensitive applications. Examples range from data manipulation causing the collision of self-driving vehicles Behzadan and Munir (2019), to Twitter users misleading chatbots into expressing misogynistic and racist remarks Neff and Nagy (2016), and adversaries compromising forward collision warning systems through false or delayed alerts Ma et al. (2021). Recognizing the severe consequences of data corruption, it becomes imperative to identify effective defense strategies.

There are three entities in this learning setting: the environment, the adversary, and the learner. The environment generates data, the learner selects a learning strategy, and the adversary crafts an attack strategy. Observing the dataset, the adversary generates a corrupted version based on the attack strategy, and the learner, oblivious to the attack strategy, must navigate the complexities of learning from this corrupted dataset. Crucially, the learner lacks visibility into the attack strategy, rendering reverse engineering and recovery of the original data unattainable. Conversely, the adversary adapts its attack strategy based on the observed learning strategy.

The presence of data corruption amplifies the complexities of decision-making. Traditional methods falter in the face of data corruption. To illustrate, mean estimation serves as a foundational problem in decision-making. For example, in the multi-armed bandit, the learner relies on a good mean estimation to evaluate an

arm. Given i.i.d. samples $x_1, x_2, \ldots, x_N$ from some Gaussian distribution $\mathcal{N}(\mu, \sigma^2)$, empirical mean estimator has an error upper bounded by $\tilde{O}(\frac{\sigma}{\sqrt{N}})$. Nonetheless, empirical mean estimation is vulnerable to data corruption: when as few as one data point is corrupted, the estimation error can be arbitrarily large because the adversary can mislead the empirical mean estimation to output an arbitrary value. As a result, the attacker's ability to alter the empirical mean, even with a single manipulated data point, can mislead the learner into choosing suboptimal arms.

Robust statistics emerge as a robust defense against data corruption, particularly in the context of robust mean estimation Huber (1992). By mitigating the influence of outliers, robust statistics exhibit minimal estimation error even in the presence of potentially unbounded data corruption. For example, when ϵ-fraction of the Gaussian samples are corrupted, trimmed-mean estimation achieves an error rate $\tilde{O}\left(\sigma\epsilon\sqrt{\log\frac{1}{\epsilon}} + \frac{\sigma}{\sqrt{N}}\right)$ by removing the left and right tails Lugosi and Mendelson (2021). Even in scenarios where the corrupted data is unbounded, the error rate remains manageable. This error rate consists of data corruption effects and statistical errors due to finite samples.

Extending this insight to complex decision-making problems, such as reinforcement learning and two-player zero-sum games, robust statistics emerge as a potent defense. By applying robust mean estimation, the impact of corrupted data is mitigated, enabling the learner to identify nearly-optimal choices with sufficient data. However, various challenges and opportunities arise in specific problem settings:

- In **online reinforcement learning**, where the adversary strategically designs attacks within a fixed budget, a comprehensive defense strategy must account for diverse attack scenarios. Possible attack patterns include: 1. concentrating the data corruption in the *initial stages* to mislead the exploration; 2. postponing the attacks until the *end of the learning process* to corrupt the final model; 3. *uniformly* distributes data corruption across the entire learning process. We resolve this online data corruption challenge by presenting a robust version of policy gradient algorithm in Chapter 3;

- In **offline learning setting**, the learner is given an offline dataset and can not

interact with the environment to get more information. The adversary may strategically concentrate data corruption along some particular direction or on certain state-action pairs. We study the corruption-robust offline learning setting in Chapter 4;

- In a **distributed learning environment** comprising both regular learning agents and adversarial agents, each fixed throughout the learning process, corrupted data comes solely from the set of Byzantine agents. In contrast to the standard setting, this additional information presents a batch structure to enhance learner performance. An effective learning algorithm should leverage this batch structure to improve robustness while managing communication and switching costs inherent to the distributed nature of the problem. We explore this Byzantine-robust distributed RL setting in Chapter 5;

- Tabular Markov decision processes (MDPs) feature a discrete set of policies, creating a safe zone with error tolerance around the optimal policy along with gap conditions for optimality. A dedicated analysis should utilize these **gap conditions** to achieve optimal policy recovery against data corruption. In the domain of two-player zero-sum games, the situation grows more intricate. Despite the discrete nature of the action space, certain games exclusively feature mixed-strategy Nash equilibria. In such cases, the clarity of gap conditions diminishes, introducing complexity to the analysis. We study the gap-dependent analysis of MDPs and two-player zero-sum games in Chapter 6 and 7 respectively.

- The preceding discussion has predominantly centered on decision-making with an adversary seeking to degenerate the performance of the learning algorithm. However, in scenarios where agents prioritize individual benefits without adversarial intent, incentives emerge as an alternative to robust statistics. As shown in Chapter 8, in these instances, **incentivizing truthful data-sharing** can eliminate the need for robust statistical defenses.

2.1 Reinforcement Learning

Reinforcement learning aims to find the optimal policy in a Markov Decision Process (MDP) (Sutton and Barto, 2018). In the online setting, the UCB-type algorithm uses optimistic bonuses to encourage exploration, which achieves the optimal regret rate in tabular MDP Azar et al. (2017); Dann et al. (2017). It has also been applied in MDP with continuous stats space Jin et al. (2020b); In the offline setting, the pessimistic algorithm uses pessimistic bonuses to account for the lack of further interactions with the environment. It is proved to be efficient Jin et al. (2021); Rashidinejad et al. (2021). Xie et al. (2020); Cui and Du (2022) generalize the optimism and pessimism principles to multi-agent settings. Policy Gradient Williams (1992); Sutton et al. (1999) and Policy optimization methods are widely used in real-world application Kakade and Langford (2002); Schulman et al. (2015b, 2017) and have shown amazing performance on challenging problems Berner et al. (2019); Akkaya et al. (2019). Using refined analysis, gap-dependent analysis achieves a faster convergence rate in the offline setting Wang et al. (2022) and logarithmic regret in the online setting Wagenmaker et al. (2022). Parallel RL deploys large-scale models in distributed system (Kretchmar, 2002). (Horgan et al., 2018; Espeholt et al., 2018) provide distributed architecture for deep reinforcement learning by parallelizing the data-generating process. (Dubey and Pentland, 2021; Agarwal et al., 2021; Chen et al., 2021a) provide the first sets of theoretical guarantees for performance and communication cost in parallel RL. Most of the prior work deal with i.i.d. data, where the error only comes from the randomness of the environment. In contrast, this thesis considers data corruption on top of the randomness, making it a strictly harder problem. The classic algorithms fail under such data corruption. We show that under certain assumptions, it's still possible to get a meaningful estimation of the environment and learn a reasonable policy even under data corruption by utilizing robust mean estimation. We give a detailed discussion below.

2.2 Robust Statistics

Robust statistics studies the estimation problem when a fraction of the dataset are corrupted. It was first studied by Tukey (1960); Anscombe (1960); Huber (1992). Recently, Diakonikolas et al. (2016); Lai et al. (2016) present the first computational and sample-efficient results for dimension-free error guarantees in high-dimension robust mean estimation problems. These results have been applied as the fundamental tools to solve supervised and unsupervised learning tasks Prasad et al. (2020); Diakonikolas et al. (2019b,c). Interested readers may refer to Diakonikolas and Kane (2023) for a survey on robust statistics. In this thesis, we resolve the robust sequential decision-making problem using a similar insight.

In the distributed learning setting, each agent collects a data batch. In the Byzantine-robust learning setting, robust statistics get an improved error rate due to the batch structure Yin et al. (2018); Zhu et al. (2023). This insight inspires our work in Chapter 5. In particular, we generalize this study to a setting with uneven batch sizes by balancing the batch weighting carefully.

There is also a line of work studying the mean estimation for heavy-tailed distribution. The challenge there is not the data corruption, but instead, the distribution itself. Lugosi and Mendelson (2019b) uses the median-of-means framework to achieve an optimal subGaussian error rate when the distribution has bounded covariance. However, the known algorithms to compute the estimator of Lugosi and Mendelson (2019b) have running time exponential in the dimension. HOPKINS (2020) provides the first polynomial-time algorithm with the optimal error rate. Lugosi and Mendelson (2021) studies the robust mean estimation problem with both heavy-tailed distribution and data corruption. It achieves the optimal error rate but requires exponential computation. Diakonikolas et al. (2020) showed that any stability-based robust mean estimator, e.g. the estimator in Diakonikolas et al. (2016, 2017) achieves optimal error with (near-)subGaussian rates. Bubeck et al. (2013); Yu et al. (2018); Medina and Yang (2016); Shao et al. (2018); Dubey et al. (2020) studies heavy-tailed bandits, where the reward distribution may not

even have finite variance. Inspired by Lugosi and Mendelson (2021), we show, in Chapter 6, that trimmed-mean estimation achieves optimal error rate for the core mean estimation problem in this setting and is robust to data corruption at the same time.

2.3 Attack and Defense in Reinforcement Learning

Reinforcement learning is vulnerable to adversarial attacks. In this section, we show that the classic Upper Confidence Bound (UCB) and Lower Confidence Bound (LCB) algorithms fail with a strong attacker while robust statistics can be adapted to provide efficient defense.

For illustration purposes, we consider a 2-arm bandit problem with arm indexed by $\{1, 2\}$. The reward of the first arm is drawn from $\mathcal{N}(\mu_1, \sigma^2)$ while the reward of the second arm is drawn from $\mathcal{N}(\mu_2, \sigma^2)$. Without loss of generality, we assume $\mu_1 > \mu_2$ and $\mu_1, \mu_2 \in [0, 1]$.

Offline Learning Setting

In the offline learning setting, the learning algorithm has access to an offline dataset collected by some behavior policy. Suppose the behavior behavior pulls each arm evenly and the offline dataset consists of N reward instantiations from each arm:

$$\{r_{1,i}\}_{i=1}^{N} \cup \{r_{2,i}\}_{i=1}^{N}.$$

Using standard concentration results, we provide an upper bound on the estimation error for empirical mean estimation on the rewards distribution: with probability at least $1 - \delta$, the following holds,

$$|\hat{\mu}_1 - \mu_1| \leq b_1, \quad |\hat{\mu}_2 - \mu_2| \leq b_2, \tag{2.1}$$

where $\hat{\mu}_k = \frac{1}{N}\sum_{k=1}^{N} r_{k,i}$ for $k = 1, 2$ and $b_1 = b_2 = \sigma\sqrt{\frac{2\log\frac{4}{\delta}}{N}}$. The right-hand side b_1, b_2 is the confidence bound on the mean estimation. To account for the uncertainty, an LCB algorithm Jin et al. (2021); Rashidinejad et al. (2021) simply choose the arm with highest lower confidence bound:

$$\hat{i} \in \operatorname*{argmax}_{i\in\{1,2\}} \hat{\mu}_i - b_i.$$

Standard results show that the suboptimality of $\hat{i}$ is at most:

$$\mu_1 - \mu_{\hat{i}} \le 2\sigma\sqrt{\frac{2\log\frac{4}{\delta}}{N}},$$

with probability at least $1-\delta$.

Consider an ϵ-strong adversary who can inspect the whole dataset and replace ϵ-fraction of the rewards from each arm. We first show that the standard LCB algorithm fails in the presence of such adversary. The adversary can simply change $r_{1,1}$ and $r_{2,1}$ to

$$-\sum_{i=2}^{N} r_{1,i}, \quad N - \sum_{i=2}^{N} r_{2,i}$$

respectively while keeping the other reward entries unchanged. With such data corruption, the LCBs of arm 1 and 2 become:

$$-\sigma\sqrt{\frac{2\log\frac{4}{\delta}}{N}}, \quad 1 - \sigma\sqrt{\frac{2\log\frac{4}{\delta}}{N}}$$

respectively. In this case, the learner will always choose arm 2 and suffer a suboptimality $\mu_1 - \mu_2$ regardless of the choice of μ_1, μ_2 and the instantiation of the dataset.

The main issue is: with data corruption, the empirical mean is no longer accurate and the bounds in (2.1) are no longer valid.

Robust mean estimation is an effective approach to defend against data corruption. When the corruption level ϵ is not too large and N is sufficiently large,

trimmed-mean estimation, as a standard robust mean estimator, achieves a nearly-optimal error bound Lugosi and Mendelson (2021). Let $\hat{\mu}_1^{\mathrm{TM}}$ and $\hat{\mu}_2^{\mathrm{TM}}$ be the output of the trimmed-mean estimation given the corrupted dataset, the results in Lugosi and Mendelson (2021) show that, with probability at least $1-\delta$,

$$\left|\hat{\mu}_1^{\mathrm{TM}} - \mu_1\right| \le b_1^{\mathrm{TM}}, \quad \left|\hat{\mu}_2^{\mathrm{TM}} - \mu_2\right| \le b_2^{\mathrm{TM}},$$

where $b_1^{\mathrm{TM}} = b_2^{\mathrm{TM}} = C_1\sigma\epsilon\sqrt{\log\frac{1}{\epsilon}} + C_2\sigma\sqrt{\frac{\log\frac{1}{\delta}}{N}}$ and C_1, C_2 are absolute constants. When using trimmed-mean estimation as a sub-routine, the robustified LCB algorithm outputs

$$\hat{i}^{\mathrm{TM}} \in \operatorname*{argmax}_{i\in\{1,2\}} \hat{\mu}_i^{\mathrm{TM}} - b_i^{\mathrm{TM}}$$

The suboptimality of $\hat{i}_{\mathrm{TM}}$ is at most:

$$\mu_1 - \mu_{\hat{i}^{\mathrm{TM}}} \le 2C_1\sigma\epsilon\sqrt{\log\frac{1}{\epsilon}} + 2C_2\sigma\sqrt{\frac{\log\frac{1}{\delta}}{N}} \tag{2.2}$$

with probability at least $1-\delta$. At a high level, $C_1\sigma\epsilon\sqrt{\log\frac{1}{\epsilon}}$ is the bias term, which is the effect of data corruption while $C_2\sigma\sqrt{\frac{\log\frac{1}{\delta}}{N}}$ is the statistical error due to finite dataset. When the corruption level ϵ is small and N is large, the right hand side of (2.2) can be much smaller than $\mu_1 - \mu_2$, the suboptimality of the classic LCB leaner under data corruption.

Online Learning Setting

In the online learning setting, the learner interacts with the environment to collect data. Suppose the learner interacts with the environment for T rounds. In each round t, the UCB algorithm builds the upper bound for each arm i:

$$\hat{\mu}_{i,t-1} + \sigma\sqrt{\frac{2\log\frac{4}{\delta}}{N_{i,t-1}}},$$

where $N_{i,t-1}$ is the number of pulls on arm i in the first $t-1$ round and $\hat{\mu}_{i,t-1}$ is the empirical mean of arm i using the data collected in the first $t-1$ rounds. The arm with the highest UCB will be pulled in each round. Standard analysis Lattimore and Szepesvári (2020) shows that the regret of UCB algorithm is $O(\sqrt{T \log \frac{T}{\delta}})$ with probability at least $1-\delta$.

We consider an ϵ-strong adversary adapted from the offline setting: the adversary can change the reward entries to arbitrary values, as long as in each round, there are at most ϵ fraction of corrupted data for each arm. Similar settings are studied in Niss and Tewari (2020); Kapoor et al. (2019).

We first show that the classic UCB algorithm fails under such adversary. Consider the following attack strategy: suppose arm 1 is pulled for the $\lceil \frac{1}{\epsilon} \rceil$-th time in round t_0 and $t_0 << T$, the adversary changes the reward r_{1,t_0} from this arm pull to:

$$(N_{1,t_0-1}+1)\left(-\sigma\sqrt{2\log\frac{4}{\delta}}\right) - N_{1,t_0-1}\hat{\mu}_{1,t_0-1}$$

and keeps all other rewards unchanged. Then in round t_0+1, the UCB of arm 1 is

$$\begin{aligned}
&\frac{1}{N_{1,t_0-1}+1}\left(N_{1,t_0-1}\hat{\mu}_{1,t_0-1} + (N_{1,t_0-1}+1)\left(-\sigma\sqrt{2\log\frac{4}{\delta}}\right) - N_{1,t_0-1}\hat{\mu}_{1,t_0-1}\right) \\
&+ \sigma\sqrt{\frac{2\log\frac{4}{\delta}}{N_{i,t_0-1}+1}} \\
=& -\sigma\sqrt{2\log\frac{4}{\delta}} + \sigma\sqrt{\frac{2\log\frac{4}{\delta}}{N_{i,t_0-1}+1}} \\
<& 0 \leq \mu_2.
\end{aligned}$$

By standard concentration results, we know that the UCB of arm 2 is always not less than μ_2 throughout the learning process with probability at least $1-\delta$. This means, with probability at least $1-\delta$, the UCB algorithm will no longer choose arm 1 after round t_0. As a result, it will suffer a regret $\Theta(T(\mu_1-\mu_2))$, which is linear in T.

On the other hand, using a scheme similar to the offline setting, the robustified UCB algorithm based on the trimmed-mean estimation and modified confidence bound achieves a regret bound $O\Big(\sigma\epsilon\sqrt{\log\frac{1}{\epsilon}}T + \sqrt{T\log\frac{T}{\delta}}\Big)$ for any ϵ-strong adversary with probability at least $1-\delta$. Even though the first term is still linear in T, it can be much smaller than $\Theta(T(\mu_1-\mu_2))$, the regret of the classic UCB algorithm, when ϵ is small.

Related Work

Huang et al. (2017) shows that adversarial attacks can significantly degrade the performance of neural network policies at test time. Ma et al. (2019) characterizes a framework for batch policy poisoning. The learner can be forced to learn a pre-chosen target policy with a small attack cost. Zhang et al. (2020a) studies the reward-poisoning attack against reinforcement learning. The feasibility of attack and robustness certification are studied. On the defense side, Niss and Tewari (2020); Kapoor et al. (2019) study corruption robust multi-armed bandits under data corruption by using robust mean estimator to estimate the expected reward of each arm. However, our work presents a more comprehensive study in MDP and multi-agent settings. Lykouris et al. (2019) study corruption robust reinforcement learning. In their setting, only a constant number of episodes can be corrupted by the adversary. The majority of their technical effort is dedicated to being agnostic to the corruption level. In contrast, using robust statistics, our work allows a constant fraction of data corruption.

3 ROBUST ONLINE REINFORCEMENT LEARNING

In this chapter, we study the robust reinforcement learning in the online setting. We first formally define the robust RL problem and present the information-theoretical limits. We show that surprisingly the natural policy gradient (NPG) method retains a natural robustness property if the reward corruption is bounded. As the main result of this chapter, we develop a Filtered Policy Gradient (FPG) algorithm that can achieve a nearly-optimal suboptimality gap even with unbounded reward corruption. Complimentary to the theoretical results, we show that a neural implementation of our methods achieves strong robust learning performance on the MuJoCo continuous control benchmarks.

This Chapter is joint work with Xuezhou Zhang, Xiaojin Zhu and Wen Sun. The author Yiding Chen contributed to the theoretical analysis of robust statistics.

3.1 Introduction

Policy gradient methods are a popular class of Reinforcement Learning (RL) methods among practitioners, as they are amenable to parametric policy classes Schulman et al. (2015b, 2017), resilient to modeling assumption mismatches Agarwal et al. (2019a, 2020a), and they directly optimizing the cost function of interest. However, one current drawback of these methods and most existing RL algorithms is the lack of robustness to data corruption, which severely limits their applications to high-stack decision-making domains with highly noisy data, such as autonomous driving, quantitative trading, or medical diagnosis.

In fact, data corruption can be a larger threat in the RL paradigm than in traditional supervised learning, because supervised learning is often applied in a controlled environment where data are collected and cleaned by highly-skilled data scientists and domain experts, whereas RL agents are developed to learn in the wild using raw feedbacks from the environment. While the increasing autonomy and less supervision mark a step closer to the goal of general artificial intelligence, they

also make the learning system more susceptible to data corruption: autonomous vehicles can misread traffic signs when the signs are contaminated by adversarial stickers Eykholt et al. (2018); chatbot can be mistaught by a small group of tweeter users to make misogynistic and racist remarks Neff and Nagy (2016); recommendation systems can be fooled by a small number of fake clicks/reviews/comments to rank products higher than they should be. Despite the many vulnerabilities, *robustness* against data corruption in RL has not been extensively studied only until recently.

The existing works on *robust* RL are mostly theoretical and can be viewed as a successor of the adversarial bandit literature. However, several drawbacks of this line of approach make them insufficient to modern real-world threats faced by RL agents. We elaborate them below:

1. **Reward vs. transition contamination**: The majority of prior works on adversarial RL focus on reward contamination Even-Dar et al. (2009); Neu et al. (2010, 2012); Zimin and Neu (2013); Rosenberg and Mansour (2019); Jin et al. (2020a), while in reality the adversary often has stronger control during the adversarial interactions. For example, when a chatbot interacts with an adversarial user, the user has full control over both the rewards and transitions during that conversation episode.

2. **Density of contamination**: The existing works that do handle adversarial/time-varying transitions can only tolerate *sublinear* number of interactions being corrupted Lykouris et al. (2019); Cheung et al. (2019); Ornik and Topcu (2019); Ortner et al. (2019). These methods would fail when the adversary's attack budget also grows linearly with time, which is often the case in practice.

3. **Practicability**: The majority of these work focuses on the setting of tabular MDPs and cannot be applied to real-world RL problems that have large state and action spaces and require function approximations.

In this work, we address the above shortcomings by developing a variant of natural policy gradient (NPG) methods that, under the linear value function assumption,

are provably robust against strongly adaptive adversaries, who can **arbitrarily contaminate** both rewards and transitions in ϵ fraction of all learning episodes. Our algorithm does not need to know ϵ, and is adaptive to the contamination level. Specifically, it guarantees to find an $\tilde{O}(\epsilon^{1/4})$-optimal policy in a polynomial number of steps. Complementarily, we also present a corresponding lower-bound, showing that no algorithm can consistently find a better than $\Omega(\epsilon)$ optimal policy, even with infinite data. In addition to the theoretical results, we also develop a neural network implementation of our algorithm which is shown to achieve strong robustness performance on the MuJoCo continuous control benchmarks Todorov et al. (2012), proving that our algorithm can be applied to real-world, high-dimensional RL problems.

3.2 Related Work

Policy Gradient and Policy Optimization Policy Gradient Williams (1992); Sutton et al. (1999) and Policy optimization methods are widely used in practice Kakade and Langford (2002); Schulman et al. (2015b, 2017) and have demonstrated amazing performance on challenging applications Berner et al. (2019); Akkaya et al. (2019). Unlike model-based approach or Bellman-backup based approaches, PG methods directly optimize the objective function and are often more robust to model-misspecification Agarwal et al. (2020a). In addition to being robust to model-misspecification, we show in this work that vanilla NPG is also robust to constant fraction and bounded adversarial corruption on both rewards and transitions. Additional discussions on other RL algorithms in standard stochastic MDPs can be found in appendix A.1.

RL with adversarial rewards. Almost all prior works on adversarial RL study the setting where the reward functions can be adversarial but the transitions are still stochastic and remain unchanged throughout the learning process. Specifically, at the beginning of each episode, the adversary must decide on a reward function for this episode, and can not change it for the rest of the episode. Also, the majority of these works focus on tabular MDPs. Early works on adversarial

MDPs assume a known transition function and full-information feedback. For example, Even-Dar et al. (2009) proposes the algorithm MDP-E and proves a regret bound of $\tilde{O}(\tau\sqrt{T\log A})$ in the non-episodic setting, where τ is the mixing time of the MDP; Later, Zimin and Neu (2013) consider the episodic setting and propose the O-REPS algorithm which applies Online Mirror Descent over the space of occupancy measures, a key component adopted by Rosenberg and Mansour (2019) and Jin et al. (2020a). O-REPS achieves the optimal regret $\tilde{O}(\sqrt{H^2T\log(SA)})$ in this setting. Several works consider the harder bandit feedback model while still assuming known transitions. The work Neu et al. (2010) achieves regret $\tilde{O}(\sqrt{H^3AT}/\alpha)$ assuming that all states are reachable with some probability α under all policies. Later, Neu et al. (2010) eliminates the dependence on α but only achieves $O(T^{2/3})$ regret. The O-REPS algorithm of Zimin and Neu (2013) again achieves the optimal regret $\tilde{O}(\sqrt{H^3SAT})$. To deal with unknown transitions, Neu et al. (2012) proposes the Follow the Perturbed Optimistic Policy algorithm and achieves $\tilde{O}(\sqrt{H^2S^2A^2T})$ regret given full-information feedback. Combining the idea of confidence sets and Online Mirror Descent, the UC-O-REPS algorithm of Rosenberg and Mansour (2019) improves the regret to $\tilde{O}(\sqrt{H^2S^2AT})$. A few recent works start to consider the hardest setting assuming unknown transition as well as bandit feedback. Rosenberg and Mansour (2019) achieves $O(T^{3/4})$ regret, which is improved by Jin et al. (2020a) to $\tilde{O}(\sqrt{H^2S^2AT})$, matching the regret of UC-O-REPS in the full information setting. Also, note that the lower bound of $\Omega(\sqrt{H^2SAT})$ Jin et al. (2018) still applies. In summary, it is found that on tabular MDPs with oblivious reward contamination, an $O(\sqrt{T})$ regret can still be achieved. Recent improvements include best-of-both-worlds algorithms Jin and Luo (2020), data-dependent bound Lee et al. (2020) and extension to linear function approximation Neu and Olkhovskaya (2020).

RL with adversarial transitions and rewards. Very few prior works study the problem of both adversarial transitions and adversarial rewards, in fact, only one that we are aware of Lykouris et al. (2019). They study a setting where only a constant C number of episodes can be corrupted by the adversary, and most of their technical effort dedicate to designing an algorithm that is agnostic to C, i.e. the algorithm doesn't need to know the contamination level ahead of time. As a result,

their algorithm takes a multi-layer structure and cannot be easily implemented in practice. Their algorithm achieves a regret of $O(C\sqrt{T})$ for tabular MDPs and $O(C^2\sqrt{T})$ for linear MDPs, which unfortunately becomes vacuous when $C \geq \Omega(\sqrt{T})$ and $C \geq \Omega(T^{1/4})$, respectively. Note that the contamination ratio C/T approaches zero when T increases, and hence their algorithm cannot handle constant fraction contamination. Notably, in all of the above works, the adversary can *partially adapt* to the learner's behavior, in the sense that the adversary can pick an adversary MDP $\mathcal{M}_k$ or reward function r_k at the start of episode k based on the history of interactions so far. However, it can no longer adapt its strategy after the episode starts, and therefore, the learner can still use a randomization strategy to trick the adversary.

A separate line of work studies the *online MDP* setting, where the MDP is not adversarial but *slowly* change over time, and the amount of change is bounded under a total-variation metric Cheung et al. (2019); Ornik and Topcu (2019); Ortner et al. (2019); Domingues et al. (2020). Due to the slow-changing nature of the environment, algorithms in these works typically uses a sliding window approach where the algorithm keeps throwing away old data and only learns a policy from recent data, assuming that most of them come from the MDP that the agent is currently experiencing. These methods typically achieve a regret in the form of $O(\Delta^c K^{1-c})$, where Δ is the total variation bound. It is worth noting that all of these regrets become vacuous when the amount of variation is linear in time, i.e. $\Delta \geq \Omega(T)$. Separately, it is shown that when both the transitions and the rewards are adversarial in every episode, the problem is at least as hard as stochastic parity problem, for which no computationally efficient algorithm exists Yadkori et al. (2013).

Learning robust controller. A different type of robustness has also been considered in RL Pinto et al. (2017); Derman et al. (2020) and robust control Zhou and Doyle (1998); Petersen et al. (2012), where the goal is to learn a control policy that is robust to potential misalignment between the training and deployment environment. Such approaches are often conservative, i.e. the learned polices are sub-optimal even if there is no corruption. In comparison, our approach can learn

as effectively as standard RL algorithms without corruption.

Robust statistics. One of the most important discoveries in modern robust statistics is that there exists computationally efficient and robust estimator that can learn near-optimally even under the strongest adaptive adversary. For example, in the classic problem of Gaussian mean estimation, the recent works Diakonikolas et al. (2016); Lai et al. (2016) present the first computational and sample-efficient algorithms. The algorithm in Diakonikolas et al. (2016) can generate a robust mean estimate $\hat{\mu}$, such that $\|\hat{\mu} - \mu\|_2 \leq O(\epsilon\sqrt{\log(1/\epsilon)})$ under ϵ corruption. Crucially, the error bound does not scale with the dimension d of the problem, suggesting that the estimator remains robust even in high dimensional problems. Similar results have since been developed for robust mean estimation under weaker assumptions Diakonikolas et al. (2017), and for supervised learning and unsupervised learning tasks Charikar et al. (2017); Diakonikolas et al. (2019b). We refer readers to Diakonikolas and Kane (2019) for a more thorough survey of recent advances in high-dimensional robust statistics.

3.3 Problem Definitions

A Markov Decision Process (MDP) $\mathcal{M} = (\mathcal{S}, \mathcal{A}, P, r, \gamma, \mu_0)$ is specified by a state space $\mathcal{S}$, an action space $\mathcal{A}$, a transition model $P : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$ (where $\Delta(\mathcal{S})$ denotes a distribution over $\mathcal{S}$), a (stochastic and possibly unbounded) reward function $r : \mathcal{S} \times \mathcal{A} \to \Delta(\mathbb{R})$, a discounting factor $\gamma \in [0, 1)$, and an initial state distribution $\mu_0 \in \Delta(\mathcal{S})$, i.e. $s_0 \sim \mu_0$. In this paper, we assume that $\mathcal{A}$ is a small and finite set, and denote $A = |\mathcal{A}|$. A policy $\pi : \mathcal{S} \to \Delta(\mathcal{A})$ specifies a decision-making strategy in which the agent chooses actions based on the current state, i.e., $a \sim \pi(\cdot|s)$.

The value function $V^\pi : \mathcal{S} \to \mathbb{R}$ is defined as the expected discounted sum of future rewards, starting at state s and executing π, i.e.

$$V^\pi(s) := \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) | \pi, s_0 = s\right],$$

where the expectation is taken with respect to the randomness of the policy and environment $\mathcal{M}$. Similarly, the *state-action* value function $Q^\pi : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is defined as $Q^\pi(s,a) := \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) | \pi, s_0 = s, a_0 = a\right]$.

We define the discounted state-action distribution d_s^π of a policy π: $d_{s'}^\pi(s,a) := (1-\gamma)\sum_{t=0}^{\infty} \gamma^t \mathrm{Pr}^\pi(s_t = s, a_t = a | s_0 = s')$, where $\mathrm{Pr}^\pi(s_t = s, a_t = a | s_0 = s')$ is the probability that $s_t = s$ and $a_t = a$, after we execute π from $t = 0$ onwards starting at state s' in model $\mathcal{M}$. Similarly, we define $d_{s',a'}^\pi(s,a)$ as: $d_{s',a'}^\pi(s,a) := (1-\gamma)\sum_{t=0}^{\infty} \gamma^t \mathrm{Pr}^\pi(s_t = s, a_t = s | s_0 = s', a_0 = a')$. For any state-action distribution ν, we write $d_\nu^\pi(s,a) := \sum_{(s',a') \in \mathcal{S}\times\mathcal{A}} \nu(s',a') d_{s',a'}^\pi(s,a)$. For ease of presentation, we assume that the agent can reset to $s_0 \sim \mu_0$ at any point in the trajectory. We denote $d_\nu^\pi(s) = \sum_a d_\nu^\pi(s,a)$.

The goal of the agent is to find a policy π that maximizes the expected value from the starting state s_0, i.e. the optimization problem is: $\max_\pi V^\pi(\mu_0) \triangleq \mathbb{E}_{s\sim\mu_0} V^\pi(s)$, where the $\max$ is over some policy class.

For completeness, we specify a d_ν^π-sampler and an unbiased estimator of $Q^\pi(s,a)$ in Algorithm 1, which are standard in discounted MDPs Agarwal et al. (2019a, 2020a). The d_ν^π sampler samples (s,a) i.i.d from d_ν^π, and the Q^π sampler returns an unbiased estimate of $Q^\pi(s,a)$ for a given pair (s,a) by a single roll-out from (s,a). Later, when we define the contamination model and the sample complexity of learning, we treat each call of d_ν^π-sampler (optionally followed by a $Q^\pi(s,a)$-estimator) as a *single episode*, as in practice both of these procedures can be achieved in a single roll-out from μ_0.

Assumption 3.3.1 (Linear Q function)**.** *For the theoretical analysis, we focus on the setting of linear value function approximation. In particular, we assume that there exists a feature map $\phi : \mathcal{S} \times \mathcal{A} \to \mathbb{R}^d$, such that for any $(s,a) \in \mathcal{S} \times \mathcal{A}$ and any policy $\pi : \mathcal{S} \to \Delta_{\mathcal{A}}$, we have*

$$Q^\pi(s,a) = \phi(s,a)^\top w^\pi, \text{ for some } \|w^\pi\| \leq W \tag{3.1}$$

We also assume that the feature is bounded, i.e. $\max_{s,a} \|\phi(s,a)\|_2 \leq 1$, and the reward function has bounded first and second moments, i.e. $\mathbb{E}[r(s,a)] \in [0,1]$ and Var$(r(s,a)) \leq$

σ^2 *for all* (s, a).

Remark 3.3.1. Assumption 4.2.1 is satisfied, for example, in tabular MDPs and linear MDPs of Jin et al. (2020b) or Yang and Wang (2019a). Unlike most theoretical RL literature, we allow the reward to be stochastic and unbounded. Such a setup aligns better with applications with a low signal-to-noise ratio and motivates the requirement for nontrivial robust learning techniques.

Notation. When clear from context, we write $d^\pi(s, a)$ and $d^\pi(s)$ to denote $d^\pi_{\mu_0}(s, a)$ and $d^\pi_{\mu_0}(s)$ respectively. For iterative algorithms which obtain policies at each episode, we let V^i,Q^i and A^i denote the corresponding quantities associated with episode i. For a vector v, we denote $\|v\|_2 = \sqrt{\sum_i v_i^2}$, $\|v\|_1 = \sum_i |v_i|$, and $\|v\|_\infty = \max_i |v_i|$. We use Uniform$(\mathcal{A})$ (in short Unif$_\mathcal{A}$) to represent a uniform distribution over the set $\mathcal{A}$.

The Contamination Model

In this paper, we study the robustness of policy gradient methods under the ϵ-*contamination model*, a widely studied adversarial model in the robust statistics literature, e.g. see Diakonikolas et al. (2016). In the classic robust mean estimation problem, given a dataset D and a learning algorithm f, the ϵ-contamination model assumes that the adversary has full knowledge of the dataset D and the learning algorithm f, and can arbitrarily change ϵ-fraction of the data in the dataset and then send the contaminated data to the learner. The goal of the learner is to identify an $O(\text{poly}(\epsilon))$-optimal estimator of the mean despite the ϵ-contamination.

Unfortunately, the original ϵ-contamination model is defined for the offline learning setting and does not directly generalize to the online setting, because it doesn't specify the availability of knowledge and the order of actions between the adversary and the learner in the time dimension. In this paper, we define the ϵ-contamination model for online learning as follows:

Definition 3.3.1 (ϵ-contamination model for Reinforcement Learning)**.** Given ϵ and the clean MDP $\mathcal{M}$, an ϵ-contamination adversary operates as follows:

1. The adversary has full knowledge of the MDP $\mathcal{M}$ and the learning algorithm, and observes all the historical interactions.I

2. At any time step t, the adversary observes the current state-action pair (s_t, a_t), as well as the reward and next state returned by the environment, (r_t, s_{t+1}). He then can decide whether to replace (r_t, s_{t+1}) with an arbitrary reward and next state $(r_t^\dagger, s_{t+1}^\dagger) \in \mathbb{R} \times \mathcal{S}$.

3. The only constraint on the adversary is that if the learning process terminates after T episodes, he can contaminate in at most ϵT episodes.

Compared to the standard adversarial models studied in online learning Shalev-Shwartz et al. (2011), adversarial bandits Bubeck and Cesa-Bianchi (2012); Lykouris et al. (2018); Gupta et al. (2019) and adversarial RL Lykouris et al. (2019); Jin et al. (2020a), the ϵ-contamination model in Definition 3.3.1 is stronger in several ways: (1) The adversary can adaptively attack after observing the action of the learner as well as the feedback from the clean environments; (2) the adversary can perturb the data arbitrarily (any real-valued reward and any next state from the state space) rather than sampling it from a pre-specified bounded adversarial reward function or adversarial MDP.

Given the contamination model, our first result is a lower-bound, showing that under the ϵ-contamination model, one can only hope to find an $O(\epsilon)$-optimal policy. Exact optimal policy identification is not possible even with infinite data.

Theorem 3.3.1 (lower bound)**.** *For any algorithm, there exists an MDP such that the algorithm fails to find an* $\left(\frac{\epsilon}{2(1-\gamma)}\right)$*-optimal policy under the* ϵ*-contamination model with a probability of at least* $1/4$*.*

Background on NPG

Given a differentiable parameterized policy $\pi_\theta : \mathcal{S} \to \Delta(\mathcal{A})$, NPG can be written in the following actor-critc style update form. With the dataset $\{s_i, a_i, \widehat{Q}^{\pi_\theta}(s_i, a_i)\}_{i=1}^N$

where $s_i, a_i \sim d_\nu^{\pi_\theta}$, and $\widehat{Q}^{\pi_\theta}(s_i, a_i)$ is unbiased estimate of $Q^{\pi_\theta}(s, a)$ (e.g., via Q^π-estimator), we have

$$\widehat{w} \in \operatorname*{argmin}_{w:\|w\|_2 \le W} \sum_{i=1}^{N} \left(w^\top \nabla \log \pi_\theta(a_i|s_i) - \widehat{Q}^{\pi_\theta}(s_i, a_i) \right)^2$$
$$\theta' = \theta + \eta \widehat{w}. \tag{3.2}$$

In theoretical part of this work, we focus on softmax linear policy, i.e., $\pi_\theta(a|s) \propto \exp(\theta^\top \phi(s, a))$. In this case, note that $\nabla \log \pi_\theta(a|s) = \phi(s, a)$, and it is not hard to verify that the policy update procedure is equivalent to:

$$\pi_{\theta'}(a|s) \propto \pi_\theta(a|s) \exp\left(\eta \widehat{w}^\top \phi(s, a)\right), \quad \forall s, a,$$

which is equivalent to running Mirror Descent on each state with a reward vector $\widehat{w}^\top \phi(s, \cdot) \in \mathbb{R}^{|\mathcal{A}|}$. We refer readers to Agarwal et al. (2019a) for more detailed explanation of NPG and the equivalence between the form in Eq. (3.2) and the classic form that uses Fisher information matrix. Similar to Agarwal et al. (2019a), we make the following assumption of having access to an exploratory reset distribution, under which it has been shown that NPG can converge to the optimal policy without contamination.

Assumption 3.3.2 (Relative condition number)**.** *With respect to any state-action distribution υ, define:*

$$\Sigma_\upsilon = \mathbb{E}_{s,a\sim\upsilon}\left[\phi_{s,a}\phi_{s,a}^\top\right],$$

and define

$$\sup_{w\in\mathbb{R}^d} \frac{w^\top \Sigma_{d^\star} w}{w^\top \Sigma_\nu w} = \kappa, \text{ where } d^*(s, a) = d_{\mu_0}^{\pi^*}(s) \circ \mathit{Unif}_{\mathcal{A}}(a)$$

We assume κ is finite and small w.r.t. a reset distribution ν available to the learner at training time.

3.4 The natural robustness of NPG against bounded corruption

Our first result shows that, surprisingly, NPG can already be robust against ϵ-contamination, if the adversary can only generate small and bounded rewards. In particular, we assume that the adversarial rewards is bounded in $[0, 1]$ (the feature $\phi(s, a)$ is already bounded).

Theorem 3.4.1 (Natural robustness of NPG)**.** *Under assumptions 4.2.1 and 3.3.2, given a desired optimality gap α, there exists a set of hyperparameters agnostic to the contamination level ϵ, such that Algorithm 2 guarantees with a $poly(1/\alpha, 1/(1-\gamma), |\mathcal{A}|, W, \sigma, \kappa)$ sample complexity that under ϵ-contamination with adversarial rewards bounded in $[0, 1]$, we have*

$$\mathbb{E}\left[V^*(\mu_0) - V^{\hat{\pi}}(\mu_0)\right] \leq \tilde{O}\left(\max\left[\alpha, W\sqrt{\frac{|\mathcal{A}|\kappa\epsilon}{(1-\gamma)^3}}\right]\right)$$

where $\hat{\pi}$ is the uniform mixture of $\pi^{(1)}$ through $\pi^{(T)}$.

A few remarks are in order.

Remark 3.4.1 (Agnostic to the contamination level ϵ)**.** It is worth emphasizing that to achieve the above bound, the hyperparameters of NPG are agnostic to the value of ϵ, and so the algorithm can be applied in the more realistic setting where the agent does not have knowledge of the contamination level ϵ, similar to what's achieved in Lykouris et al. (2019) with a complicated nested structure. The same property is also achieved by the FPG algorithm in the next section.

Remark 3.4.2 (Dimension-independent robustness guarantee)**.** Theorem 3.4.1 guarantees that NPG can find an $O(\epsilon^{1/2})$-optimal policy after polynomial number of episodes, provided that $|\mathcal{A}|$ and κ are small. Conceptually, the relative condition number κ indicates how well-aligned the initial state distribution is to the occupancy distribution of the optimal policy. A good initial distribution can have a κ as small as 1, and so κ is independent of d. Interested readers can refer to Agarwal et al.

(2019a) (Remark 6.3) for additional discussion on the relative condition number. Here, importantly, the optimality gap does not directly scale with d, and so the guarantee will not blow up on high-dimensional problems. This is an important attribute of robust learning algorithms heavily emphasized in the traditional robust statistics literature.

The proof of Theorem 3.4.1 relies on the following NPG regret lemma, first developed by Even-Dar et al. (2009) for the MDP-Expert algorithm and later extend to NPG by Agarwal et al. (2019a, 2020a):

Lemma 3.4.1 (NPG Regret Lemma)**.** *Suppose Assumption 4.2.1 and 3.3.2 hold and Algorithm 2 starts with* $\theta^{(0)} = 0$, $\eta = \sqrt{2\log|\mathcal{A}|/(W^2T)}$. *Suppose in addition that the* (*random*) *sequence of iterates satisfies the assumption that*

$$\mathbb{E}\left[\mathbb{E}_{s,a\sim d^{(t)}}\left[\left(Q^{\pi^{(t)}}(s,a) - \phi(s,a)^\top w^{(t)}\right)^2\right]\right] \leq \epsilon_{stat}^{(t)}.$$

Then, we have that

$$\begin{aligned}\mathbb{E}\left[\sum_{t=1}^{T}\{V^*(\mu_0) - V^{(t)}(\mu_0)\}\right] \\ \leq \frac{W}{1-\gamma}\sqrt{2\log|\mathcal{A}|T} + \sum_{t=1}^{T}\sqrt{\frac{4|\mathcal{A}|\kappa\epsilon_{stat}^{(t)}}{(1-\gamma)^3}}.\end{aligned} \tag{3.3}$$

Intuitively, Lemma 3.4.1 decompose the regret of NPG into two terms. The first term corresponds to the regret of standard mirror descent procedure, which scales with $\sqrt{T}$. The second term corresponds to the estimation error on the Q value, which acts as the reward signal for mirror descent. When not under attack, estimation error $\epsilon_{stat}^{(t)}$ goes to zero as the number of samples M gets larger, which in turn implies the global convergence of NPG. However, when under bounded attack, the generalization error $\epsilon_{stat}^{(t)}$ will not go to zero even with infinite data. Nevertheless, we can show that it is bounded by $O(\epsilon^{(t)})$ when the sample size M is large enough, where $\epsilon^{(t)}$ denotes the fraction of episodes being corrupted in iteration t. Note that by definition, we have $\sum_t \varepsilon^{(t)} \leq \varepsilon T$.

Algorithm 1 d_ν^π sampler and Q^π estimator

1: **Function** d_ν^π-sampler
2: **Input**: A reset distribution $\nu \in \Delta(\mathcal{S} \times \mathcal{A})$.
3: Sample $s_0, a_0 \sim \nu$.
4: Execute π from s_0, a_0; at any step t with (s_t, a_t), return (s_t, a_t) with probability $1 - \gamma$.
5: **Function** Q^π-estimator
6: **Input**: current state-action (s, a), a policy π.
7: Execute π from $(s_0, a_0) = (s, a)$; at step t with (s_t, a_t), terminate with probability $1 - \gamma$.
8: **Return**: $\widehat{Q}^\pi(s, a) = \sum_{i=0}^{t} r(s_i, a_i)$.

Lemma 3.4.2 (Robustness of linear regression under bounded contamination)**.** *Suppose the adversarial rewards are bounded in* $[0, 1]$*, and in a particular iteration* t*, the adversary contaminates* $\epsilon^{(t)}$ *fraction of the episodes, then given* M *episodes, it is guaranteed that with probability at least* $1 - \delta$*,*

$$\mathbb{E}_{s,a\sim d^{(t)}}\left[\left(Q^{\pi^{(t)}}(s,a) - \phi(s,a)^\top w^{(t)}\right)^2\right] \leq 4\left(W^2 + WH\right)\left(\epsilon^{(t)} + \sqrt{\frac{8}{M}\log\frac{4d}{\delta}}\right). \tag{3.4}$$

where $H = (\log\delta - \log M)/\log\gamma$ *is the effective horizon.*

This along with the NPG regret lemma guarantees that the expected regret of NPG is bounded by $O(\sqrt{T} + M^{-1/4} + \sqrt{\epsilon}T)$ which in turn guarantees to identify an $O(\sqrt{\epsilon})$-optimal policy.

3.5 FPG: Robust NPG against unbounded corruption

Our second result is the Filtered Policy Gradient (FPG) algorithm, a robust variant of the NPG algorithm Kakade (2001); Agarwal et al. (2019a) that can be robust against arbitrary and *potentially unbounded* data corruption. Specifically, FPG replace the standard linear regression solver in NPG with a statistically robust

Algorithm 2 Natural Policy Gradient (NPG)

Require: Learning rate η; number of episodes per iteration M

Initialize $\theta^{(0)} = 0$.

for $t = 0, 1, \ldots, T-1$ **do**

  Call Algorithm 1 M times with $\pi^{(t)}$ to obtain a dataset that consist of $s_i, a_i \sim d_\nu^{(t)}$ and $\widehat{Q}^{(t)}(s_i, a_i)$, $i \in [M]$.

  Solve the linear regression problem

$$w^{(t)} = \operatorname*{argmin}_{\|w\|_2 \le W} \sum_{i=1}^{M} \left(\widehat{Q}^{(t)}(s_i, a_i) - w^\top \nabla_\theta \phi(s_i, a_i) \right)^2$$

  Update $\theta^{(t+1)} = \theta^{(t)} + \eta w^{(t)}$.

end for

Algorithm 3 Robust Linear Regression via `SEVER`

Input: Dataset $\{(x_i, y_i)\}_{i=1:M}$, a standard linear regression solver $\mathcal{L}$, and parameter $\sigma' \in \mathbb{R}_+$.

Initialize $S \leftarrow \{1, \ldots, M\}$, $f_i(w) = \|y_i - w^\top x_i\|^2$.

repeat

  $w \leftarrow \mathcal{L}(\{(x_i, y_i)\}_{i \in S})$. ▷ Run learner on S.

  Let $\widehat{\nabla} = \frac{1}{|S|} \sum_{i \in S} \nabla f_i(w)$.

  Let $G = [\nabla f_i(w) - \widehat{\nabla}]_{i \in S}$ be the $|S| \times d$ matrix of centered gradients.

  Let v be the top right singular vector of G.

  Compute the vector τ of *outlier scores* defined via $\tau_i = \left((\nabla f_i(w) - \widehat{\nabla}) \cdot v \right)^2$.

  $S' \leftarrow S$

  if $\frac{1}{|S|} \sum_{i \in S} \tau_i \le c_0 \cdot {\sigma'}^2$, for some constant $c_0 > 1$ **then**

    $S = S'$ ▷ We only filter out points if the variance is larger than an appropriately chosen threshold.

  else

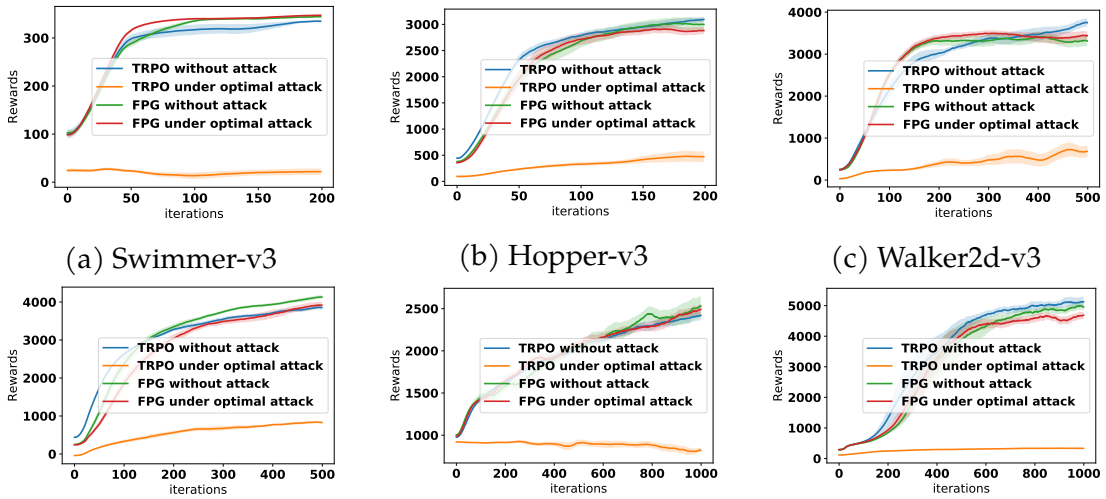
    Draw T from Uniform$[0, \max_i \tau_i]$.

    $S = \{i \in S : \tau_i < T\}$.

  end if

until $S = S'$.

Return w.



(a) Swimmer-v3 (b) Hopper-v3 (c) Walker2d-v3

(d) HalfCheetah-v3 (e) Ant-v3 (f) Humanoid-v3

Figure 3.1: Experiment Results on the 6 MuJoTo benchmarks.

alternative. In this work, we use the SEVER algorithm Diakonikolas et al. (2019b). In practice, one can substitute it with any computationally efficient robust linear regression solver. We show that FPG can find an $O(\epsilon^{1/4})$-optimal policy under ϵ-contamination with a polynomial number of samples.

Theorem 3.5.1. *Under assumptions 4.2.1 and 3.3.2, given a desired optimality gap α, there exists a set of hyperparameters agnostic to the contamination level ϵ, such that Algorithm 2, using Algorithm 3 as the linear regression solver, guarantees with a $poly(1/\alpha, 1/(1-\gamma), |\mathcal{A}|, W, \sigma, \kappa)$ sample complexity that under ϵ-contamination, we have*

$$\mathbb{E}\left[V^*(\mu_0) - V^{\hat{\pi}}(\mu_0)\right] \leq \tilde{O}\left(\max\left[\alpha, \sqrt{\frac{|\mathcal{A}|\kappa\left(W^2 + \sigma W\right)}{(1-\gamma)^4}}\epsilon^{1/4}\right]\right). \tag{3.5}$$

where $\hat{\pi}$ is the uniform mixture of $\pi^{(1)}$ through $\pi^{(T)}$.

The proof of Theorem 3.5.1 relies on a similar result to Lemma 3.4.2, which shows that if we use Algorithm 3 as the linear regression subroutine, then $\epsilon_{stat}^{(t)}$

can be bounded by $O(\sqrt{\epsilon^{(t)}})$ when the sample size M is large enough, even under unbounded ϵ-contamination.

Lemma 3.5.1 (Robustness of SEVER under unbounded contamination)**.** *Suppose the adversarial rewards are unbounded, and in a particular iteration t, the adversarial contaminate $\epsilon^{(t)}$ fraction of the episodes, then given M episodes, it is guaranteed that if $\epsilon^{(t)} \leq c$, for some absolute constant c, and any constant $\tau \in [0, 1]$, we have*

$$\begin{aligned} &\mathbb{E}\left[\mathbb{E}_{s,a\sim d^{(t)}}\left[\left(Q^{\pi^{(t)}}(s,a) - \phi(s,a)^\top w^{(t)}\right)^2\right]\right] \\ &\leq O\left(\left(W^2 + \frac{\sigma W}{1-\gamma}\right)\left(\sqrt{\epsilon^{(t)}} + f(d,\tau)M^{-\frac{1}{2}} + \tau\right)\right). \end{aligned} \tag{3.6}$$

where $f(d,\tau) = \sqrt{d\log d} + \sqrt{\log(1/\tau)}$.

In Lemma 3.5.1, c is the break point of SEVER and is an absolute constant that does not depend on the data, and $(1-\tau)$ is the probability that the clean data satisfies a certain stability condition which suffices for robust learning.

3.6 Experiments

In the theoretical analysis, we rely on the assumption of linear Q function, finite action space and exploratory initial state distribution to prove the robustness guarantees for NPG and FPG. In this section, we present a practical implementation of FPG, based on the *Trusted Region Policy Optimization* (TRPO) algorithm Schulman et al. (2015a), in which the conjugate gradient step (equivalent to the linear regression step in Alg. 2) is robustified with SEVER. The pseudo-code and implementation details are discussed in appendix A.7. In this section, we demonstrate its empirical performance on the MuJoCo benchmarks Todorov et al. (2012), a set of high-dimensional continuous control domains where both assumptions no longer holds, and show that FPG can still consistently performs near-optimally with and without attack.

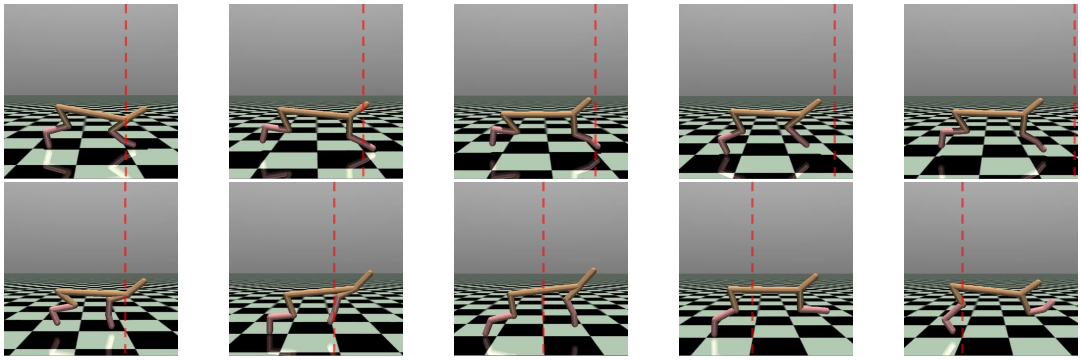
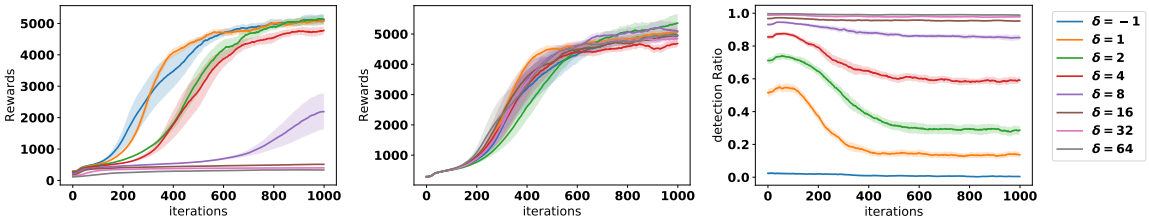


Figure 3.2: Consecutive Frames of Half-Cheetah trained with TRPO (top row) and FPG (bottom row) respectively under $\delta = 100$ attack. TRPO was fooled to learn a "running backward" policy, contrasted with the normal "running forward" policy learned by FPG.



(a) TRPO Rewards (b) FPG Rewards (c) FPG Detection Ratio

Figure 3.3: Detailed Results on Humanoid-v3.

Attack mechanism: While designing and calculating the *optimal* attack strategy against a deep RL algorithm is still a challenging problem and active area of research Ma et al. (2019); Zhang et al. (2020a), here we describe the poisoning strategy used in our empirical evaluation, which, despite being simple, can fool non-robust RL algorithms with ease. Conceptually, policy gradient methods can be viewed as a stochastic gradient ascent method, where each iteration can be simplified as:

$$\theta^{(t+1)} = \theta^{(t)} + g^{(t)} \tag{3.7}$$

where $g^{(t)}$ is a gradient step that ideally points in the direction of fastest policy improvement. Assuming that $g^{(t)}$ is a good estimate of the gradient direction, then

a simple attack strategy is to try to perturb $g^{(t)}$ to point in the $-g^{(t)}$ direction, in which case the policy, rather than improving, will deteriorate as learning proceed. A straightforward way to achieve this is to flip the rewards and multiply them by a big constant δ in the adversarial episodes. In the linear regression subproblem of Alg. 2, this would result in a set of (x, y) pairs whose y becomes $-\delta y$. This in expectation will make the best linear regressor w point to the opposite direction, which is precisely what we want.

This attack strategy is therefore parameterized by a single parameter δ, which guides the magnitude of the attack, and is **adaptively tuned** against each learning algorithm in the experiments: Throughout the experiment, we set the contamination level $\epsilon = 0.01$, and tune δ among the values of $[1, 2, 4, 8, 16, 32, 64]$ to find the most effective magnitude against each learning algorithm. All experiments are repeated with 3 random seeds and the mean and standard deviations are plotted in the figures.

Results: The experiment results are shown in Figure 4.1. Consistent patterns can be observed across all environments: vanilla `TRPO` performs well without attack but fails completely under the adaptive attack (which choose $\delta = 64$ in all environments). `FPG`, on the other hand, matches the performance of vanilla `TRPO` with or without attack. Figure 3.2 showcase two half-cheetah control policies learned by `TRPO` and `FPG` under attack with $\delta = 100$. Interestingly, due to the large negative adversarial rewards, TRPO actually learns the "running backward" policy, showing that our attack strategy indeed achieves what it's designed for. In contrast, `FPG` is still able to learn the "running forward" policy despite the attack.

Figure 3.3 shows the detailed performances of `TRPO` and `FPG` across different δ's on the hardest *Humanoid* environment. One can observe that `TRPO` actually learns robustly under attacks of small magnitude ($\delta = 1, 2, 4$) and achieves similar performances to itself in clean environments, verifying our theoretical result in Theorem 3.4.1. In contrast, `FPG` remains robust across all values of δ's. Figure 3.3c shows the proportion of adversary data detected and removed by `FPG`'s filtering subroutine throughout the learning process. One can observe that as the attack norm δ increases, the filtering algorithm also does a better job detecting the adversarial data

and thus protect the algorithm from getting inaccurate gradient estimates. Similar patterns can be observed in all the other environments, and we defer the additional figures to the appendix.

3.7 Discussions

To summarize, in this work we present a robust policy gradient algorithm `FPG`, and show theoretically and empirically that it can learn in the presence of strong data corruption. Despite our results, many open questions remain unclear:

1. `FPG` does not handle exploration and relies on an exploratory initial distribution. Can we design algorithms that achieve the same *dimension-free* robustness guarantee without such assumptions?

2. Our $O(\epsilon^{1/4})$ upper-bound and $O(\epsilon)$ lower-bound are not tight. Information theoretically, what is the best robustness guarantee one can achieve under ϵ-contamination?

3. The `SEVER` algorithm requires computing the top eigenvalue of an $n \times d$ matrix, which is memory and time consuming when using large neural networks (large d). More computationally efficient robust learning method will be extremely valuable to make FPG truly scale.

4. In the experiment, we focus on TRPO as the closest variant of NPG. Can other policy gradient algorithm, such as PPO and SAC, be robustified in similar fashions and achieve strong empirical performance?

We believe that answering these questions will be important steps towards robust reinforcement learning.

4 ROBUST OFFLINE REINFORCEMENT LEARNING

In this chapter, we study reinforcement learning in the offline setting. Unlike the online setting, the learner learns from a corrupted offline dataset but no further interaction with the environment is allowed. By utilizing robust supervised learning oracles, we propose robust variants of the Least-Square Value Iteration (LSVI) algorithm. Furthermore, our method achieves near-matching performances in cases both with and without global data coverage.

This Chapter is joint work with Xuezhou Zhang, Xiaojin Zhu and Wen Sun. The author Yiding Chen contributed to part of the theoretical analysis.

4.1 Introduction

Offline Reinforcement Learning (RL) (Lange et al., 2012; Levine et al., 2020) has received increasing attention recently due to its appealing property of avoiding online experimentation and making use of offline historical data. In applications such as assistive medical diagnosis and autonomous driving, historical data is abundant and keeps getting generated by high-performing policies (from human doctors/drivers). However, it is often unethical or expensive to allow an online RL algorithm to freely experiment with potentially suboptimal policies, as often human lives are at stake. Offline RL provides a powerful framework aiming to find a good policy based on historical data alone. Exciting advances have been made in designing stable and high-performing empirical offline RL algorithms (Fujimoto et al., 2019; Laroche et al., 2019; Wu et al., 2019; Kumar et al., 2019, 2020; Agarwal et al., 2020d; Kidambi et al., 2020; Siegel et al., 2020; Liu et al., 2020; Yang and Nachum, 2021; Yu et al., 2021). On the theoretical front, recent works have proposed efficient algorithms with theoretical guarantees, based on the principle of *pessimism in face of uncertainty* (Liu et al., 2020; Buckman et al., 2020; Yu et al., 2020; Jin et al., 2020c; Rashidinejad et al., 2021), or variance reduction (Yin et al., 2020, 2021).

In this work, however, we investigate a different aspect of the offline RL framework, namely the statistical robustness in the presence of data corruption. Data corruption is one of the main security threats against modern ML systems: autonomous vehicles can misread traffic signs contaminated by adversarial stickers (Eykholt et al., 2018); chatbots were misguided by tweeter users to make misogynistic and racist remarks (Neff, 2016); recommendation systems are fooled by fake reviews/comments to produce incorrect rankings. Despite the many vulnerabilities, robustness against data corruption has not been extensively studied in RL until recently. To the best of our knowledge, *all* prior works on corruption-robust RL study the online RL setting. As direct extensions to the setting of adversarial bandits, earlier works focus on designing robust algorithms in *fully adversarial* environments, i.e. the reward functions at all rounds are adversarially generated, and show that $O(\sqrt{T})$ regret is achievable (Even-Dar et al., 2009; Neu et al., 2010, 2012; Zimin and Neu, 2013; Rosenberg and Mansour, 2019; Jin et al., 2020a). While such setting might appear certain game-theoretical situations, in most practical scenarios, such as the ones described above, only a small fraction of the data are actually adversarial while the majority of the data are benign.

Recent works start to study the *Huber's contamination* setting (Lykouris et al., 2019; Chen et al., 2021b), where both rewards and transitions can be contaminated but only in ϵ fraction of all episodes. This setting turns out to be significantly harder, and both works can only tolerate at most $\epsilon \leq O(1/\sqrt{T})$ fraction of corruptions even against oblivious adversaries. Zhang et al. (2021b) recently proposes the first online RL algorithm that can be robust against a constant fraction (i.e. $\epsilon \geq \Omega(1)$) of adaptive corruption on both rewards and transitions while being agnostic to the value of ϵ, albeit requiring the help of an exploration policy with finite relative condition number.

In this work, we extend the study of robust RL to the offline setting. Following (Lykouris et al., 2019; Chen et al., 2021b; Zhang et al., 2021b), we study the *Huber's contamination model* in offline reinforcement learning, formally defined in Assumption 4.2.2. Huber's contamination model is a classic model for studying sparse data contamination, and is widely used in the traditional literature of robust

statistics (Huber et al., 1967). We refer interesting readers to a comprehensive survey (Diakonikolas and Kane, 2019) of recent advances along these directions. Motivated by these prior works, in this paper we ask the following question:

Given an offline RL dataset with ϵ-fraction of corrupted data, what is the information-theoretic limit of robust identification of the optimal policy?

Towards answering this question, we summarize the following contributions of this work:

1. We provide the formal definition of ϵ-contamination model in offline RL, and establish an information-theoretical lower-bound of $\Omega(Hd\epsilon)$ in the setting of linear MDP with dimension d.

2. We design a robust variant of the Least-Square Value Iteration (LSVI) algorithm utilizing robust supervised learning oracles with a novel pessimism bonus term, and show that it achieves near-optimal performance in cases with (Theorem 4.3.2) or without global data coverage (Theorem 4.3.3).

3. In the without coverage case, we establish a sufficient condition for learning based on the relative condition number with respect to any comparator policy — not necessary the optimal one. When specialized to offline RL without corruption, our partial coverage assumption is much weaker than the full coverage assumption in (Jin et al., 2020c) for linear MDP.

4. In contrast to (Zhang et al., 2021b), we show that agnostic learning, i.e. learning without the knowledge of ϵ, is generally impossible in the offline RL setting, establishing a separation in hardness between online and offline RL in face of data corruption.

While our paper's main contributions are on corruption robust offline RL, it is worth noting when specialized to the clean offline RL setting, i.e., $\epsilon = 0$, our work also gives two improved results: (1) under the linear MDP setting, we achieve an optimality gap with respect to any comparator policy (not necessarily the optimal one) in the order of $O(d^{3/2}/\sqrt{N})$ with N being the number of offline samples, saving a $\sqrt{d}$

factor over previously best-known results. (2) our analysis works for the setting where offline data only has partial coverage which is formalized using the concept of relative condition number with respect to the comparator policy[1].

4.2 Preliminaries

To begin with, let us formally introduce the episodic linear MDP setup we will be working with, the data collection and contamination protocol, as well as the robust linear regression oracle.

Environment. We consider an episodic finite-horizon Markov decision process (MDP), $\mathcal{M}(\mathcal{S}, \mathcal{A}, P, R, H, \mu_0)$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, $P : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$ is the transition function, such that $P(\cdot|s, a)$ gives the distribution over the next state if action a is taken from state s, $R : \mathcal{S} \times \mathcal{A} \to \Delta(\mathbb{R})$ is a stochastic and potentially unbounded reward function, H is the time horizon, and $\mu_0 \in \Delta_{\mathcal{S}}$ is an initial state distribution. The value functions $V_h^\pi : \mathcal{S} \to \mathbb{R}$ is the expected sum of future rewards, starting at time h in state s and executing π, i.e. $V_h^\pi(s) := \mathbb{E}\left[\sum_{t=h}^{H} R(s_t, a_t)|\pi, s_0 = s\right]$, where the expectation is taken with respect to the randomness of the policy and environment $\mathcal{M}$. Similarly, the *state-action* value function $Q_h^\pi : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is defined as $Q_h^\pi(s, a) := \mathbb{E}\left[\sum_{t=h}^{H} R(s_t, a_t)|\pi, s_0 = s, a_0 = a\right]$. We use π_h^*, Q_h^*, V_h^* to denote the optimal policy, Q-function and value function, respectively. For any function $f : \mathcal{S} \to \mathbb{R}$, we define the Bellman operator as

$$(\mathbf{B}f)(s, a) = \mathbb{E}_{s' \sim P(\cdot|s,a)}[R(s, a) + f(s')]. \tag{4.1}$$

We then have the Bellman equation

$$V_h^\pi(s) = \langle Q_h^\pi(s, \cdot), \pi_h(\cdot|s)\rangle_{\mathcal{A}},\ Q_h^\pi(s, a) = (\mathbf{B}V_{h+1}^\pi)(s, a)$$

[1]Contemporary to ours, Jin et al. (2020c) added a new Corollary 4.5 in the latest arXiv version of their paper that matches with our results.

and the Bellman optimality equation

$$V_h^*(s) = \max_a Q_h^*(s,a), \quad Q_h^*(s,a) = (\mathbf{B}V_{h+1}^*)(s,a)$$

We define the averaged state-action distribution d^π of a policy π: $d^\pi(s,a) := \frac{1}{H}\sum_{h=1}^H \Pr^\pi(s_t = s, a_t = a | s_0 \sim \mu_0)$. We aim to learn a policy that maximizes the expected cumulative reward and thus define the performance metric as the suboptimality of the learned policy π compared to a *comparator policy* $\tilde{\pi}$:

$$\mathrm{SubOpt}(\pi, \tilde{\pi}) = \mathbb{E}_{s\sim\mu_0}[V_1^{\tilde{\pi}}(s) - V_1^{\pi}(s)]. \tag{4.2}$$

Notice that $\tilde{\pi}$ doesn't necessarily have to be the optimal policy π^*, in contrast to most recent results in pessimistic offline RL, such as (Jin et al., 2020c; Buckman et al., 2020).

For the majority of this work, we focus on the linear MDP setting (Yang and Wang, 2019a; Jin et al., 2020b).

Assumption 4.2.1 (Linear MDP)**.** *There exists a known feature map $\phi : \mathcal{S}\times\mathcal{A} \to \mathbb{R}^d$, d unknown signed measures $\mu = (\mu^{(1)}, ..., \mu^{(d)})$ over $\mathcal{S}$ and an unknown vector $\theta \in \mathbb{R}^d$, such that for all $(s,a,s') \in \mathcal{S}\times\mathcal{A}\times\mathcal{S}$,*

$$P(s'|s,a) = \phi(s,a)^\top \mu(s'),\ R(s,a) = \phi(s,a)^\top\theta + \omega$$

where ω is a zero-mean and σ^2-subgaussian distribution. Here we also assume that the parameters are bounded, i.e.$\|\phi(s,a)\| \leq 1$, $\mathbb{E}[R(s,a)] \in [0,1]$ for all $(s,a) \in \mathcal{S}\times\mathcal{A}$ and $\max(\|\mu(\mathcal{S})\|, \|\theta\|) \leq \sqrt{d}$.

Clean Data Collection. We consider the offline setting, where a clean dataset $\tilde{D} = \{(\tilde{s}_i, \tilde{a}_i, \tilde{r}_i, \tilde{s}_i')\}_{i=1:N}$ of transitions is collected a priori by an unknown experimenter. In this work, we assume the stochasticity of the clean data collecting process, i.e. there exists an offline state-action distribution $\nu \in \Delta(\mathcal{S}\times\mathcal{A})$, s.t. $(\tilde{s}_i, \tilde{a}_i) \sim \nu(s,a)$, $\tilde{r}_i \sim R(\tilde{s}_i, \tilde{a}_i)$ and $\tilde{s}_i' \sim P(\tilde{s}_i, \tilde{a}_i)$. When there is no corruption, $\tilde{D}$ will be observed

by the learner. However, in this work, we study the setting where the data is contaminated by an adversary before revealed to the learner.

Contamination model. We define an adversarial model that can be viewed as a direct extension to the *ϵ-contamination model* studied in the traditional robust statistics literature.

Assumption 4.2.2 (ϵ-Contamination in offline RL)**.** *Given $\epsilon \in [0, 1]$ and a set of clean tuples $\tilde{D} = \{(\tilde{s}_i, \tilde{a}_i, \tilde{r}_i, \tilde{s}'_i)\}_{i=1:N}$, the adversary is allowed to inspect the tuples and replace any ϵN of them with arbitrary transition tuples $(s, a, r, s') \in \mathcal{S} \times \mathcal{A} \times \mathbb{R} \times \mathcal{S}$. The resulting set of transitions is then revealed to the learner. We will call such a set of samples ϵ-corrupted, and denote the contaminated dataset as $D = \{(s_i, a_i, r_i, s'_i)\}_{i=1:N}$. In other words, there are at most ϵN number of indices i, on which $(\tilde{s}_i, \tilde{a}_i, \tilde{r}_i, \tilde{s}'_i) \neq (s_i, a_i, r_i, s'_i)$.*

Under ϵ-contamination, we assume access to a robust linear regression oracle.

Assumption 4.2.3 (Robust least-square oracle (RLS))**.** *Given a set of ϵ-contaminated samples $S = \{(x_i, y_i)\}_{1:N}$, where the clean data is generated as: $\tilde{x}_i \sim \nu$, $P(\|x\| \leq 1) = 1$, $\tilde{y}_i = \tilde{x}_i \top w^* + \gamma_i$, where γ_i's are subgaussian noise with zero-mean and γ^2-variance. Then, a robust least-square oracle returns an estimator $\hat{w}$, such that*

1. *If $\mathbb{E}_\nu[xx^\top] \succeq \xi$, then with probability at least $1-\delta$, $\|\hat{w}-w^*\|_2 \leq c_1(\delta) \cdot \left(\sqrt{\frac{\gamma^2 \mathrm{poly}(d)}{\xi^2 N}} + \frac{\gamma}{\xi}\epsilon\right)$*
2. *With probability at least $1-\delta$, $\mathbb{E}_\nu\left(\|x^\top(\hat{w}-w^*)\|_2^2\right) \leq c_2(\delta) \cdot \left(\frac{\gamma^2 \mathrm{poly}(d)}{N} + \gamma^2\epsilon\right)$*

where c_1 and c_2 hide absolute constants and polylog$(1/\delta)$.

Such guarantees are common in the robust statistics literature, see e.g. (Bakshi and Prasad, 2020; Pensia et al., 2020; Klivans et al., 2018). In particular, in the simpler setting of bounded reward, i.e. $r_i \in [0, 1]$ for all i, Regular Least Square (RLS) already satisfies Assumption 4.2.3 with $\mathrm{poly}d = O(d)$, see e.g. Appendix F of (Lykouris et al., 2019). We note that while we focus on oracles with such guarantees, our algorithm and analysis are modular and allow one to easily plug in oracles with stronger or weaker guarantees.

4.3 Algorithms and Main Results

In this work, we focus on a Robust variant of Least-Squares Value Iteration (LSVI)-style algorithms (Jin et al., 2020c), which directly calls a robust least-square oracle to estimate the Bellman operator $\hat{\mathbf{B}}\hat{V}_h(s,a)$. Optionally, it may also subtract a pessimistic bonus $\Gamma_h(s,a)$ during the Bellman update. A template of such an algorithm is defined in Algorithm 4. In section 4.3 and 4.3, we present two variants of the LSVI algorithm designed for two different settings, depending on whether the data has full coverage over the whole state-action space or not. However, before that, we first present an algorithm-independent minimax lower-bound that illustrates the hardness of the robust learning problem in offline RL, in contrast to classic results in statistical estimation and supervised learning.

Minimax Lower-bound

Theorem 4.3.1 (Minimax Lower bound)**.** *Under assumptions 4.2.1 (linear MDP) and 4.2.2 (ϵ-contamination), for any fixed data-collecting distribution ν, no algorithm $L : (\mathcal{S} \times \mathcal{A} \times \mathbb{R} \times \mathcal{A})^N \to \Pi$ can find a better than $O(dH\epsilon)$-optimal policy with probability more than $1/4$ on all MDPs. Specifically,*

$$\min_{L,\nu} \max_{\mathcal{M}, f_c} \text{SubOpt}(\hat{\pi}, \pi^*) = \Omega\left(dH\epsilon\right) \tag{4.3}$$

where f_c denotes an ϵ-contamination strategy that corrupts the data based on the MDP $\mathcal{M}$ and clean data $\tilde{D}$ and returns a contaminated dataset, and L denotes an algorithm that takes the contaminated dataset and return a policy $\hat{\pi}$, i.e. $\hat{\pi} = L(f_c(\mathcal{M}, \tilde{D}))$.

The detailed proof is presented in appendix B.2, but the high-level idea is simple. Consider the tabular MDP setting which is a special case of linear MDP with $d = SA$. For any data generating distribution ν, by the pigeonhole principle, there must exists a least-sampled (s,a) pair, for which $\nu(s,a) \leq 1/SA$. If the adversary *concentrate* all its attack budget on this least sampled (s,a) pair, it can perturb the empirical reward on this (s,a) pair to be as much as $\hat{r}(s,a) = r(s,a) + SA\epsilon$. Further more, assume

that there exists another (s^*, a^*) such that $r(s^*, a^*) = r(s, a) + SA\epsilon/2$. Then, the learner has no way to tell if truly $r(s, a) > r(s^*, a^*)$ (i.e., the learner believes what she observes and believes there is no contamination) or if the data is contaminated and in fact $r(s, a) < r(s^*, a^*)$. Either could be true and whichever alternative the learner chooses to believe, it will suffer at least $SAH\epsilon/2$ optimality gap in one of the two scenarios.

Remark 4.3.1 (dimension scaling)**.** Theorem 4.3.1 says that even if the algorithm has control over the data collecting distribution ν (without knowing $\mathcal{M}$ a priori), it can still do no better than $\Omega(dH\epsilon)$ in the worst-case, which implies that robustness is fundamentally impossible in high-dimensional problems where $d \gtrsim 1/\epsilon$. This is in sharp contrast to the classic results in the robust statistics literature, where estimation errors are found to not scale with the problem dimension, in settings such as robust mean estimation (Diakonikolas et al., 2016; Lai et al., 2016) and robust supervised learning (Charikar et al., 2017; Diakonikolas et al., 2019b). From the construction we can see that the dimension scaling appears fundamentally due to a *multi-task learning* effect: the learner must perform SA separate reward mean estimation problems for each (s, a) pair, while the data is provided as a mixture for all these tasks. As a result, the adversary can concentrate on one particular task, raising the contamination level to effectively $d\epsilon$.

Remark 4.3.2 (Offline vs. Online RL)**.** We note that the construction in Theorem 4.3.1 remains valid even if the adversary only contaminates the rewards, and if the adversary is oblivious and perform the contamination based only on the data generating distribution ν rather than the instantiated dataset $\tilde{D}$. In contrast, the best-known lower-bound for robust online RL is $\Omega(H\epsilon)$ (Zhang et al., 2021b). It remains unknown whether $\Omega(H\epsilon)$ is tight, as no algorithm yet can achieve a matching upper-bound without additional information. We will come back to this discussion in section 4.3.

In what follows, we show that the above lower-bound is tight in both d and ϵ, by presenting two upper-bound results nearly matching the lower-bound.

Algorithm 4 Robust Least-Square Value Iteration (R-LSVI)

1: Input: Dataset $D = \{(s_i, a_i, r_i, s_i')\}_{1:N}$; pessimism bonus $\Gamma_h(s,a) \geq 0$, robust least-squares Oracle: $RLS(\cdot)$.
2: Split the dataset randomly into H subset: $D_h = \{(s_i^h, a_i^h, r_i^h, s_i'^h)\}_{1:(N/H)}$, for $h \in [H]$.
3: Initialization: Set $\hat{V}_{H+1}(s) \leftarrow 0$.
4: **for** step $h = H, H-1, \ldots, 1$ **do**
5: Set $\hat{w}_h \leftarrow RLS\left(\left\{(\phi(s_i^h, a_i^h), y_i^h)\right\}_{i \in D_h}\right)$, where $y_i^h = r_i^h + \hat{V}_{h+1}(s_i'^h)$.
6: Set $\hat{Q}_h(s,a) \leftarrow \phi(s,a)^\top \hat{w}_h - \Gamma_h(s,a)$, clipped within $[0, H-h+1]$.
7: Set $\hat{\pi}_h(a|s) \leftarrow \mathrm{argmax}_a \hat{Q}_h(s,a)$ and $\hat{V}_h(s) \leftarrow \max_a \hat{Q}_h(s,a)$.
8: **end for**
9: Output: $\{\hat{\pi}_h\}_{h=1}^H$.

Robust Learning with Data Coverage

To begin with, we study the simple setting where the offline data has sufficient coverage over the whole state-action distribution. This is often considered as a strong assumption. However, results in this setting will establish meaningful comparison to the above lower-bound and the no-coverage results later. In the context of linear MDP, we say that a data generating distribution has coverage if it satisfies the following assumption.

Assumption 4.3.1 (Uniform Coverage)**.** *Under assumption 4.2.1, define*

$$\Sigma_\nu \triangleq \mathbb{E}_\nu[\phi(s,a)\phi(s,a)^\top]$$

as the covariance matrix of ν. We say that the data generating distribution ξ-covers the state-action space for $\xi > 0$, if $\Sigma_\nu \succeq \xi I$ i.e. the smallest eigenvalue of Σ_ν is strictly positive and at least ξ.

Under such an assumption, we show that the R-LSVI without pessimism bonus can already be robust to data contamination.

Theorem 4.3.2 (Robust Learning under ξ-Coverage)**.** *Under assumption 4.2.1, 4.2.2 and 4.3.1, for any $\xi, \epsilon > 0$, given a dataset of size N, Algorithm 4 with bonus $\Gamma_h(s,a) = 0$ achieves*

$$\mathrm{SubOpt}(\hat{\pi}, \pi^*) \leq \tilde{O}\left(\sqrt{\frac{(\sigma+H)^2 H^3 \mathrm{poly}(d)}{\xi^2 N}} + \frac{(\sigma+H)H^2}{\xi}\epsilon\right) \tag{4.4}$$

with probability at least $1-\delta$.

The proof of Theorem 4.3.2 follows readily from the standard analysis of approximated value iterations and rely on the following classic result connecting the Bellman error to the suboptimality of the learned policy, see e.g. Section 2.3 of (Jiang, 2020).

Lemma 4.3.1 (Optimality gap of VI)**.** *Under assumption 4.2.1, Algorithm 4 with $\Gamma_h(s,a) = 0$ satisfies*

$$\begin{aligned}\mathrm{SubOpt}(\hat{\pi}, \pi^*) &\leq 2H \max_{s,a,h} |\hat{Q}_h(s,a) - (\mathbf{B}_h \hat{V}_{h+1})(s,a)| \\ &\leq 2H \max_{s,a,h} \|\phi(s,a)\|_2 \cdot \|\hat{w}_h - w_h^*\|_2 \end{aligned} \tag{4.5}$$

where $w_h^* \triangleq \theta + \int_{\mathcal{S}} \hat{V}_{h+1}(s')\mu_h(s')ds'$ is the best linear predictor.

The result then follows immediately using property 1 of the robust least-square oracle and the fact that $\mathbb{E}[((r(s,a) + \hat{V}(s')) - (\mathbf{B}_h \hat{V})(s,a))^2 | s,a] \leq (\sigma+H)^2$ (Lemma B.1.2).

Remark 4.3.3 (Data Splitting and tighter d-dependency)**.** The data splitting in step 2 of Algorithm 4 is mainly for the sake of theoretical analysis and is not required for practical implementations. Nevertheless, it directly contributes to our tighter bounds. Specifically, the data splitting makes $\hat{V}_{h+1}$, which is learned based on D_{h+1}, independent from D_h, at the cost of an additional H multiplicative factor. In contrast, the typical covering argument used in online RL will introduce another $O(d^{1/2})$ multiplicative factor, and naively applying it to the offline RL setting will make the finally sample complexity scales as $O(d^{3/2})$, see e.g. Corollary 4.5 of (Jin

et al., 2020c). Our result above, when specialized to offline RL without corruption (i.e., $\epsilon = 0$), achieves the following results.

Corollary 4.3.1 (Uncorrupted Learning under ξ-Coverage)**.** *Under assumption 4.2.1 and 4.3.1, for any $\xi > 0$, given a clean dataset of size N, with bonus $\Gamma_h(s,a) = 0$ and ridge regression with regularizer coefficient $\lambda = 1$ as the RLS solver, Algorithm 4 achieves with probability at least $1-\delta$*

$$\mathrm{SubOpt}(\hat{\pi}, \pi^*) \leq \tilde{O}\left(\frac{H^3 d}{\xi\sqrt{N}}\right). \tag{4.6}$$

Remark 4.3.4 (Tolerable ϵ)**.** Notice that Theorem 4.3.2 requires $\epsilon \leq \xi$ to provide a non-vacuous bound. This is because if $\epsilon > \xi$, then similar to the lower-bound construction in Theorem 4.3.1, the adversary can corrupt all the data along the eigenvector direction corresponding to the smallest eigenvalue, in which case the empically estimated reward along that direction can be arbitrarily far away from the true reward even with a robust mean estimator, and thus the estimation error becomes vacuous.

Remark 4.3.5 (Unimprovable gap)**.** Notice in contrast to classic RL results, Theorem 4.3.2 implies that in the presence of data contamination, there exists an unimprovable optimality gap $(\sigma + H)H^2\epsilon/\xi$ for the proposed algorithm, even if the learner has access to infinite data. Also note that because $\|\phi(s,a)\| \leq 1$, ξ is at most $1/d$. This implies that asymptotically, $V^* - V^{\hat{\pi}} \leq O(H^3 d\epsilon)$ when ξ is on the order of $1/d$, matching the lower-bound upto H factors.

Remark 4.3.6 (Agnosticity to problem parameters)**.** It is worth noting that in theorem 4.3.2, the algorithm does not require the knowledge of ϵ or ξ, and thus works in the agnostic setting where these parameters are not available to the learner (given that the robust least-square oracle is agnostic). In other words, the algorithm and the bound are adaptive to both ϵ and ξ. This point will be revisited in the next section.

Robust Learning without Coverage

Next, we consider the harder setting where assumption 4.3.1 does not hold, as often in practice, the offline data will not cover the whole state-action space. Instead, we provide a much weaker sufficient condition under which offline RL is possible.

Assumption 4.3.2 (relative condition number)**.** *For any given comparator policy $\tilde{\pi}$, under assumption 4.2.1 and 4.2.2, define the relative condition number as*

$$\kappa = \sup_{w} \frac{w^\top \tilde{\Sigma} w}{w^\top \Sigma_\nu w} \tag{4.7}$$

where $\tilde{\Sigma}$ denotes $\Sigma_{d^{\tilde{\pi}}}$ and we take the convention that $\frac{0}{0} = 0$. We assume that $\kappa < \infty$.

The relative condition number is recently introduced in the policy gradient literature (Agarwal et al., 2019a; Zhang et al., 2021b). Intuitively, the relative condition number measures the worst-case density ratio between the occupancy distribution of comparator policy and the data generating distribution. For example, in a tabular MDP, $\kappa = \max_{s,a} \frac{d^{\tilde{\pi}}(s,a)}{\nu(s,a)}$. Here, we show that a finite relative condition number with respect to an *arbitrary* comparator policy is already sufficient for offline RL, for both clean and contaminated setting.

Without data coverage, we now rely on pessimism to retain reasonable behavior. However, the challenge, in this case, is to design a valid confidence bonus using only the corrupted data. We now present our constructed pessimism bonus that allows Algorithm 4 to handle ϵ-corruption, albeit requiring the knowledge of ϵ.

Theorem 4.3.3 (Robust Learning without Coverage)**.** *Under assumption 4.2.1, 4.2.2 and 4.3.2, with $\epsilon > 0$, given any comparator policy $\tilde{\pi}$ with $\kappa < \infty$, define the ϵ-robust empirical covariance as*

$$\Lambda_h = \frac{3}{5}\left(\frac{H}{N}\sum_{i \in \mathbb{D}_h} \phi(s_i^h, a_i^h)\phi(s_i^h, a_i^h)^\top + (\epsilon + \lambda)\cdot I\right), \tag{4.8}$$
$$\lambda = c' \cdot dH \log(N/\delta)/N$$

where D_h denotes the data for step h and c' is an absolute constant. Then, Algorithm 4 with pessimism bonus

$$\Gamma_h(s,a) = \left(\frac{(\sigma+H)\sqrt{H}\text{poly}(d)}{\sqrt{N}} + ((\sigma+H) + 2H\sqrt{d})\sqrt{\epsilon} + \sqrt{d\lambda}\right)\sqrt{c_2(\delta/H)}\|\phi(s,a)\|_{\Lambda_h^{-1}} \tag{4.9}$$

will with probability at least $1-\delta$ achieve

$$\text{SubOpt}(\hat{\pi},\tilde{\pi}) \leq \tilde{O}\left(\frac{(\sigma+H)\sqrt{H^3\kappa}\text{poly}(d)}{\sqrt{N}} + ((\sigma+H)H + H^2\sqrt{d})\sqrt{d\kappa\epsilon}\right) \tag{4.10}$$

Remark 4.3.7 (Arbitrary comparator policy)**.** Notice that in comparison to Theorem 4.2 of (Jin et al., 2020c), Lemma C.4.1 allows the comparator policy to be arbitrary, and the implication is profound. Specifically, Lemma C.4.1 indicates that a pessimism-style algorithm *always* retains reasonable behavior, in the sense that, given enough data, it will eventually find the best policy among all the policies covered by the data generating distribution, i.e. $\text{argmax}_\pi V^\pi(\mu)$, s.t. $\kappa(\pi) < \infty$. Similar to the ξ-coverage, when specialized to standard offline RL, our analysis provides a tighter bound.

Corollary 4.3.2 (Uncorrupted Learning without Coverage)**.** *Under assumption 4.2.1 and 4.3.2, given any comparator policy $\tilde{\pi}$ with $\kappa < \infty$, define the empirical covariance as*

$$\Lambda_h = \frac{H}{N}\sum_{i=1}^{N/H}\phi(s_i^h,a_i^h)\phi(s_i^h,a_i^h)^\top + \lambda\cdot I \tag{4.11}$$
$$\lambda = c'\cdot dH\log(N/\delta)/N$$

where c' is an absolute constant. Then, with pessimism bonus

$$\Gamma_h(s,a) = H\left(\sqrt{d\cdot\lambda} + \sqrt{\frac{Hd\log(N/\delta\lambda)}{N}}\right)\cdot\|\phi(s,a)\|_{\Lambda_h^{-1}}$$

and ridge regression with regularizer coefficient λ as the RLS solver, Algorithm 4 will with probability at least $1-\delta$ achieve

$$\mathrm{SubOpt}(\hat{\pi},\tilde{\pi}) \leq \tilde{O}\left(\left(H^2 d + H^{2.5}\sqrt{d}\right)\sqrt{\frac{d\kappa}{N}}\right) \tag{4.12}$$

We note that the leading term (first term) $O(d^{3/2})$ is directly due to the assumption that the linear MDP parameter $\max(\|\mu(\mathcal{S})\|, \|\theta\|) \leq \sqrt{d}$. If instead $\max(\|\mu(\mathcal{S})\|, \|\theta\|) \leq \rho$ for some ρ indepdent of d, then the above bound will become linear in d. In contrast, the covering-number style analysis will generate $d^{3/2}$ regardless of the parameter norm, since its second term will become $O(d^{3/2})$ and dominate (as one needs to perform a covering argument to cover the quadratic penalty term $\Gamma_h(s,a)$).

The proof of Theorem 4.3.3 is technical but largely follows the analysis framework of pessimism-based offline RL and consists of two main steps. The first step establishes $\Gamma_h(s,a)$ as a valid bonus by showing

$$|\hat{Q}_h(s,a) - (\mathbf{B}_h\hat{V}_{h+1})(s,a)| \leq \Gamma_h(s,a), \text{ w.p. } 1-\delta/H. \tag{4.13}$$

The second step applies the following Lemma connectingthe optimality gap with the expectation of $\Gamma_h(s,a)$ under visitation distribution of the comparator policy.

Lemma 4.3.2 (Suboptimality for Pessimistic Value Iteration)**.** *Under assumption 4.2.1, and under the event $\mathcal{E}$ that the $\Gamma_h(s,a)$ satisfies the required property of bounding the Bellman error, i.e. $|\hat{Q}_h(s,a) - (\mathbf{B}_h\hat{V}_{h+1})(s,a)| \leq \Gamma_h(s,a), \forall h \in [H]$, then against any comparator policy $\tilde{\pi}$, it achieves*

$$\mathrm{SubOpt}(\hat{\pi},\tilde{\pi}) \leq 2\sum_{h=1}^{H}\mathbb{E}_{d^{\tilde{\pi}}}[\Gamma_h(s,a)] \tag{4.14}$$

We then further upper-bound the expectation through the following inequality, which bounds the distribution shift effect using the relative condition number κ:

$$\mathbb{E}_{d^{\tilde{\pi}}}\left[\sqrt{\phi(s,a)^\top\Lambda^{-1}\phi(s,a)}\right] \leq \sqrt{5d\kappa} \tag{4.15}$$

The detailed proof can be found in Appendix B.3. Note that the prior work (Jin et al., 2020c) only establishes results in terms of the suboptimality comparing with the optimal policy, and when specializes to linear MDPs, they assume the offline data has global full coverage. We replace these redundant assumptions with a single assumption of partial coverage with respect to any comparator policy, in the form of a finite relative condition number.

Remark 4.3.8 (Novel bonus term)**.** One of our main algorithmic contributions is the new bonus term that upper-bound the effect of data contamination on the Bellman error. Ignoring ϵ-independent additive terms and absolute constants, our bonus term has the form

$$H\sqrt{\epsilon} \cdot \sqrt{\phi(s,a)^\top \Lambda^{-1} \phi(s,a)}. \tag{4.16}$$

In comparison, below is the one used in (Lykouris et al., 2019) for online corruption-robust RL:

$$H\epsilon \cdot \sqrt{\phi(s,a)^\top \Lambda^{-2} \phi(s,a)}. \tag{4.17}$$

In the tabular case, (4.17) evaluates to $H\epsilon/\nu(s,a)$ and (4.16) evaluates to $H\sqrt{\epsilon/\nu(s,a)}$, and thus (4.17) is actually tighter than (4.16) for $\nu(s,a) \geq \epsilon$. However, in the linear MDP case, the relation between the two is less obvious. As we shall see, when offline distribution has good coverage, i.e. Λ is well-conditioned, (4.17) appears to be tighter. However, as the smallest eigenvalue of Λ goes to zero, a.k.a. lack of coverage, (4.17) actually blows up rapidly, whereas both (4.16) and the actual achievable gap remain bounded.

We demonstrate these behaviors with a numerical simulation, shown in Figure 4.1. In the simulation, we compare the size of three terms

$$\text{maximum possible gap} = \tag{4.18}$$

$$\max_{\|y\|_\infty \leq 2H, \|y\|_0 \leq \epsilon N} \phi(s,a)^\top \Lambda^{-1} \left(\frac{1}{N} \sum_{i=1}^{N} \phi(s_i, a_i) \cdot y_i \right) \tag{4.19}$$

$$\text{bonus } 1 = H\epsilon \cdot \sqrt{\phi(s,a)^\top \Lambda^{-2} \phi(s,a)}$$

$$\text{bonus } 2 = H\sqrt{\epsilon} \cdot \sqrt{\phi(s,a)^\top \Lambda^{-1} \phi(s,a)}$$

The maximum possible gap is defined as above since for any (s, a) pair and in any step h, the bias introduced to its Bellman update due to corruption takes the form of

$$\phi(s,a)^\top \Lambda^{-1} \left(\frac{1}{N} \sum_{i=1}^{N} \phi_i \left(\tilde{y}_i - y_i \right) \right) \tag{4.20}$$

where $\tilde{y}_i = \tilde{r}_i + \hat{V}_{h+1}(\tilde{s}'_i)$ and $y_i = r_i + \hat{V}_{h+1}(s'_i)$, in which $\tilde{r}_i$ and $\tilde{x}'_i$ are the clean reward and transitions. For the sake of clarity, here we assume that the adversary only contaminates the reward and transitions in a bounded fashion while keeping the current (s, a)-pairs unchanged. (4.20) can then be upper-bounded by (4.19), because there are at most ϵN tuples on which $\tilde{r}_i \neq r_i$ or $\tilde{s}'_i \neq s'_i$, and for any such tuple $(\tilde{r}_i + \hat{V}_{h+1}(\tilde{s}'_i)) - (r_i + \hat{V}_{h+1}(s'_i)) \leq 2H$.

In the simulation, we set $H = 1$ to ignore the scaling on time horizon and let $\lambda = 1$; We let both the test data $\phi(s, a)$ and the training data $\phi(s_i, a_i)$ to be sampled from a truncated standard Gaussian distribution in $\mathbb{R}^3$, denoted by ν, with mean 0, and covariance eigenvalues $1, 1, \lambda_{\min}$. We set the training data size set to $N = 10^6$ and contamination level set to $\epsilon = 0.01$. The x-axis tracks $-\log(\lambda_{\min})$, while the y-axis tracks $\mathbb{E}_{s,a\sim\nu}\text{bonus}(s, a)$, with expectation being approximated by 1000 test samples from ν. It can be seen that bonus 1 starts off closely upper-bounding the maximum possible gap when the data has good coverage, but increases rapidly as $\lambda_{\min}$ decreases. Note that for a fixed N, bonus 1 will eventually plateau at $HN\epsilon/\lambda$, but this term scales with N, so the error blows up as the number of samples grows, which certainly is not desirable. Bonus 2, on the other hand, is not as tight as bonus 1 when there is good data coverage, but remains intact regardless of the value of $\lambda_{\min}$, which is essential for the more challenging setting with poor data coverage.

This new bonus term can be of independent interest in other robust RL contexts. For example, in the online corruption-robust RL problem, as a result of using the looser bonus term (4.16), the algorithm in (Lykouris et al., 2019) can only handle $\epsilon = T^{-3/4}$ amount of corruptions in the linear MDP setting, while being able to handle $\epsilon = T^{-1/2}$ amount of corruptions in the tabular setting, due to the tabular bonus being tighter. Our bonus term can be directly plugged into their algorithm, allowing it to handle up to $\epsilon = T^{-1/2}$ amount of corruption even in the linear MDP

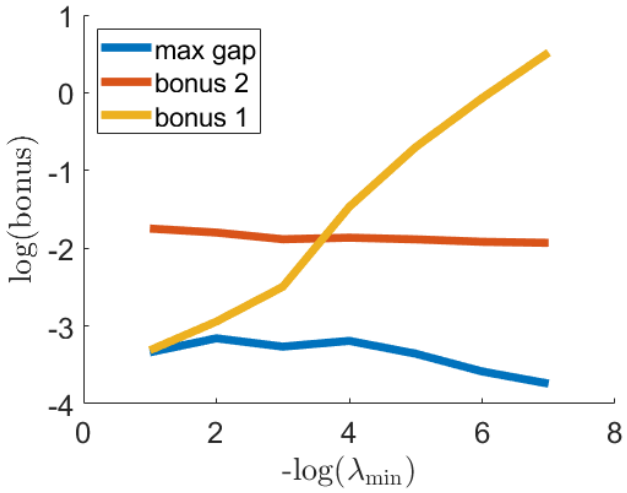


Figure 4.1: bonus size simulation

setting, achieving an immediate improvement over previous results.[2]

Note that our algorithm and theorem are adaptive to the unknown relative coverage κ, but is not adaptive to the level of contamination ε (i.e., algorithm requires knowing ε or a tight upper bound of ε). One may ask whether there exists an agnostic result, similar to Theorem 4.3.2, where an algorithm can be adaptive simultaneously to unknown values of ϵ and coverage parameter κ. Our last result shows that this is unfortunately not possible without full data coverage. In particular, we show that no algorithm can achieve a best-of-both-worlds guarantee in both clean and ϵ-corrupted environments. More specifically, in this setting, κ is still unknown to the learner, and the adversary either corrupt ε amount of tuples (ϵ is known) or does not corrupt at all—but the learner does not know which situation it is.

Theorem 4.3.4 (Agnostic learning is impossible without full coverage)**.** *Under assumption 4.2.1 and 4.3.2, for any algorithm* $L : (\mathcal{S} \times \mathcal{A} \times \mathbb{R} \times \mathcal{A})^N \to \Pi$ *that achieves diminishing suboptimality in clean environment, i.e., for any clean dataset* $\tilde{\mathcal{D}}$ *it achieves*

[2]Though our bound improve their result, the tolerable corruption amount is still sublinear, which is due to the multi-layer scheduling procedure used in their algorithm.

$\mathrm{SubOpt}(L(\tilde{\mathcal{D}})) = g(N)$ *for some positive function* g *such that* $\lim_{N\to\infty} g(N) = 0$, *we have that for any* $\epsilon \in (0, 1/2]$, *there exists an MDP* $\mathcal{M}^\dagger$ *such that with probability at least* $1/4$, $\max_{f_c} \mathrm{SubOpt}(\hat{\pi}, \tilde{\pi}) \geq 1/2$.

Intuitively, the logic behind this result is that to achieve vanishing errors in the clean environment, the learner has no choice but to *trust* all data as clean. However, it is also possible that the same dataset could be generated under some adversarial corruption from another MDP with a very different optimal policy—thus the learner cannot be robust to corruption under that MDP.

Specifically, consider a 2-arm bandit problem. The learner observes a dataset of N data points of arm-reward pairs, of which p fraction is arm a_1 and $(1-p)$ fraction is arm a_2. For simplicity, we assume that N is large enough such that the empirical distribution converges to the underlying sampling distribution. Assume further that the average reward observed for a_1 is $\hat{r}_1 = \frac{1}{2} + \frac{\epsilon}{2p}$, for some $\epsilon \leq p$, and the average reward observed for a_2 is $\frac{1}{2}$. Given such a dataset, two data generating processes can generate such a dataset with equal likelihood and thus indistinguishable based only on the data:

1. There is no contamination. The MDP has a reward setting where a_1 indeed has reward $r_1 = \text{Bernoulli}(\frac{1}{2} + \frac{\epsilon}{2p})$ and a_2 has $r_2 = \text{Bernoulli}(\frac{1}{2})$. Since there is no corruption, $\kappa = 1/p$ in this MDP.

2. The data is ϵ-corrupted. In particular, in this MDP, the actual reward of a_1 is $r_1 = \text{Bernoulli}(\frac{1}{2} - \frac{\epsilon}{2p})$, and the adversary is able to increase empirical mean by ϵ/p via changing ϵN number of data points from $(a_1, 0)$ to $(a_1, 1)$. One can show that this can be achieved by the adversary with probability at least $1/2$ (which is where the probability $1/2$ in the theorem statement comes from). In this MDP, we have $\kappa = 1/(1-p)$.

Now, since the algorithm achieves a diminishing suboptimal gap in all clean environments, it must return a_1 with high probability given such a dataset, due to the possibility of the learner facing the data generation process 1. However, committing

to action a_1 will incur $\epsilon/2p$ suboptimal gap in the second MDP with the data generation process 2. On the other hand, note that the relative condition number in the second MDP is bounded, i.e. $\frac{1}{1-p} \leq 2$ for $\epsilon \leq p \leq 1/2$. Therefore, for any $\epsilon \in (0, 1/2]$, one can construct such an instance with $p = \epsilon$, such that the relative condition number for the second MDP is $\frac{1}{1-p} \leq 2$ and the relative condition number for the first MDP is $\frac{1}{\epsilon} < \infty$, while the learner would always suffer $\epsilon/2p = 1/2$ suboptimality gap in the second MDP if she had to commit to a_1 under the first MDP where data is clean.

Remark 4.3.9 (Offline vs. Online RL: Agnostic Learning)**.** Theorem 4.3.4 shows that no algorithm can simultaneously achieve good performance in both clean and corrupted environments without knowing which one it is currently experiencing. This is in sharp contrast to the recent result in (Zhang et al., 2021b), which shows that in the online RL setting, natural policy gradient (NPG) algorithm can find an $O(\sqrt{\kappa\epsilon})$-optimal policy for any unknown contamination level ϵ with the help of an exploration policy with finite relative condition number. Without such a helper policy, however, robust RL is much harder, and the best-known result (Lykouris et al., 2019) can only handle $\epsilon \leq O(1/\sqrt{T})$ corruption, but still does not require the knowledge of ϵ. Intuitively, such adaptivity is lost in the offline setting, because the learner is no longer able to evaluate the current policy by collecting on-policy data. In the online setting, the construction in Theorem 4.3.4 will not work. Our construction heavily relies on the fact that ν has ε probability of sampling a_1, which allows adversary in the second MDP to concentrate its corruption budget all on a_1. In the online setting, one can simply uniform randomly try a_1 and a_2 to significantly increase the probability of sampling a_1 which in turn makes the estimation of r_1 accurate (up to $O(\varepsilon)$ in the corrupted data generation process).

4.4 Discussions and Conclusion

In this paper, we studied corruption-robust RL in the offline setting. We provided an information-theoretical lower bound and two near-matching upper-bounds

for cases with or without full data coverage, respectively. We also establish an impossibility result, showing that an agnostic algorithm is impossible in corruption-robust offline RL and distincting the offline setting from the online counterpart. Finally, when specialized to the uncorrupted setting, our algorithm and analysis also obtained tighter bounds than prior works.

5 BYZANTINE-ROBUST REINFORCEMENT LEARNING

In the last two chapters, we study online and offline reinforcement with an adversary who can change a constant fraction of the data. When applying robust learning method, the corruption results in a bias term in the suboptimality gap. However, when the data corruption has some special structure, one may aim to achieve a better robustness guarantee. In this chapter, we consider a distributed reinforcement learning setting where multiple agents separately explore the environment and communicate their experiences through a central server. However, a portion of agents are adversarial and can report arbitrary fake information. This means the clean and corrupted data form clean and corrupted data batches. By utilizing the batch structure, we show that our algorithms achieve sublinear regret in the online setting and diminishing bound on suboptimality gap in the offline setting.

5.1 Introduction

Distributed learning systems have been one of the main driving forces to recent successes of deep learning (Verbraeken et al., 2020; Goyal et al., 2017; Abadi et al., 2016). Advances in designing efficient distributed optimization algorithms(Horgan et al., 2018) and deep learning infrastructures (Espeholt et al., 2018) have enabled the training of powerful models with hundreds of billions of parameters (Brown et al., 2020). However, new challenges emerge with the outsourcing of computation and data collection. In particular, distributed systems have been found vulnerable to Byzantine failure (LAMPORT et al., 1982), meaning there could be agents with failure that may send arbitrary information to the central server. Even a small number of Byzantine machines that send out moderately corrupted data can lead to a significant loss in performance (Yin et al., 2018; Ma et al., 2019; Zhang et al., 2020a), which raise security concern in real-world applications such as chatbot (Neff and Nagy, 2016) and autonomous vehicles (Eykholt et al., 2018; Ma et al., 2021). In addition, other desired properties are chased after, such as protecting the

data privacy of individual data contributors (Sakuma et al., 2008; Liu et al., 2019) and reducing communication cost (Dubey and Pentland, 2021). These challenges require new algorithmic design on the server side, which is the main focus of this paper.

When it comes to reinforcement learning (RL), distributed learning has been prevalent in many large-scale decision-making problems even before the deep learning era, such as cooperative learning in robotics systems (Ding et al., 2020a), power grids optimization (Yu et al., 2014) and automatic traffic control (Bazzan, 2009). Unlike supervised learning, where the data distribution of interest is often fixed prior, reinforcement learning requires active exploration on the agent's side to discover the optimal policy for the current task, thus creating new challenges in achieving the above desiderata while exploring in an unknown environment.

This paper studies this exact problem:

Can we design a distributed RL algorithm that is sample efficient and robust to Byzantine agents while having small communication costs and promoting data privacy?

We study Byzantine-robust RL in both the online and offline settings: In the online setting, a central server is designed to outsource exploration tasks to m agents iteratively, the agents collect experiences and send them back to the server, and the server uses the data to update its policy; In the offline setting, a central server collects logged data from m agents and uses the data to identify a good policy without additional interaction with the environment. However, among the m agents, an α-fraction is Byzantine, meaning they can send arbitrary data in both the online and offline settings. We summarize our contributions as follows:

1. We design COW, a robust mean estimation algorithm for learning from batches. By utilizing the batch structure, the estimation error of our algorithm vanishes with more data. Compared to prior works (Qiao and Valiant, 2017; Chen et al., 2020; Jain and Orlitsky, 2021; Yin et al., 2018), our algorithm adapts to arbitrary batch sizes, which is desired in many applications of interest.

2. We design Byzan-UCBVI, a Byzantine-Robust variant of optimistic value iteration for online RL, by calling COW as a subroutine. We show that Byzan-UCBVI achieves near-optimal regret with α-fraction Byzantine agents. Meanwhile, Byzan-UCBVI also enjoys a logarithmic communication cost and switching cost (Bai et al., 2019; Zhang et al., 2020b; Gao et al., 2021), and preserves data privacy of individual agents.
3. We design Byzan-PEVI, a Byzantine-Robust variant of pessimistic value iteration for offline RL, again utilizing COW as a subroutine. Despite the presence of Byzantine agents, we show that Byzan-PEVI can learn a near-optimal policy with a polynomial number of samples when certain coverage conditions are satisfied (Zhang et al., 2021a).

5.2 Related Work

Reinforcement Learning: Reinforcement learning aims to find the optimal policy in a Markov Decision Process (MDP) (Sutton and Barto, 2018). Here we mainly survey prior works that introduce ideas and theoretical tools that inspire our work. (Azar et al., 2017; Dann et al., 2017) show that UCB-style algorithms achieve minimax regret bound in tabular MDPs. Recent work extends the theoretical understanding to RL with function approximation (Jin et al., 2020b; Yang and Wang, 2019a, 2020). Our analysis for the online RL algorithm follows the theoretical framework of *optimism in the face of uncertainty*, yet the technical steps differ significantly from the above works. (Jin et al., 2021; Rashidinejad et al., 2021) use a pessimistic strategy to efficiently learn a near-optimal policy in the offline setting. The same principle is utilized in the design of our offline RL algorithm. Recently, (Bai et al., 2019; Zhang et al., 2020b; Gao et al., 2021) study low switching-cost RL algorithm, meaning the learning agent only performs a small number of policy changes. Our algorithm borrows ideas from these works to simultaneously achieve small communication costs and statistical robustness.

Distributed Reinforcement Learning: Parallel RL deploys large-scale models in distributed system (Kretchmar, 2002). (Horgan et al., 2018; Espeholt et al., 2018) provide distributed architecture for deep reinforcement learning by parallelizing the data-generating process. (Dubey and Pentland, 2021; Agarwal et al., 2021; Chen et al., 2021a) provide the first sets of theoretical guarantees for performance and communication cost in parallel RL. We take a step further to study the Byzantine-robust problem in distributed RL.

Robust Statistics: Robust statistics studies learning with corrupted datasets and has a long history (Huber, 1992; Tukey, 1960). In modern machine learning, models are high-dimensional. Recent work provides sample and computationally efficient algorithms for robust mean and covariance estimation in high dimension (Diakonikolas et al., 2016, 2017; Lai et al., 2016). Shortly after, those robust mean estimators are applied to robust supervised learning (Diakonikolas et al., 2019b; Prasad et al., 2018) and RL (Zhang et al., 2021a,b). A line of work of particular interest to us studies robust learning from data batches (Qiao and Valiant, 2017; Chen et al., 2020; Jain and Orlitsky, 2021; Yin et al., 2018). They consider a setting where the data is collected from many distinct data sources, and a fraction of the data sources is corrupted. By exploiting the batch structure of the data, these algorithms can achieve significantly higher accuracy than in the non-batch setting (Diakonikolas et al., 2016). However, to our best knowledge, all of these works study batches with equal sizes, which does not often capture situations in practice. In contrast, our algorithm in page 54 works for arbitrarily different batch sizes and achieves a near-optimal rate *adaptively*.

Byzantine-Robust Distributed Learning: Byzantine-Robust learning algorithm studies learning under Byzantine failure (LAMPORT et al., 1982). (Chen et al., 2017) provides a Byzantine gradient descent via the geometric median of mean estimation for the gradients. (Yin et al., 2018) provides robust distributed gradient descent algorithms with optimal statistics rates. These works also restrict to a setting where each worker handles the same number of gradient computations. As we will

show later, their algorithm and rate will no longer be optimal when the batch sizes differ.

Corruption-Robust RL And Byzantine-Robust RL: There is a line of work studying adversarial attack against reinforcement learning (Ma et al., 2019; Zhang et al., 2020a; Huang et al., 2017), and corruption robust reinforcement RL for online (Zhang et al., 2021b; Lykouris et al., 2021) and offline (Zhang et al., 2021a) settings. (Jadbabaie et al., 2022) studies Byzantine-Robust linear bandits in the federated setting. Unlike our setting, they allow different agents to be subject to Byzantine attacks in different episodes. Our algorithm enjoys a better regret bound and communication cost. (Fan et al., 2021) provides a Byzantine-robust policy gradient algorithm that is guaranteed to converge to an approximately stationary point, whereas our algorithm guarantees to find an approximately optimal policy. (Dubey and Pentland, 2020) studies Byzantine-Robust multi-armed bandit, where the corruption can only come from a fixed distribution. We study a more difficult MDP setting and allow the corruption to be arbitrary.

5.3 Robust Mean Estimation From Untruthful Batches

To prepare for our discussion of byzantine-robust RL, we first discuss an important subproblem called *robust mean estimation from batches*, which captures many of the unique properties and challenges byzantine-robust RL faces. Indeed, our byzantine-robust RL algorithms will crucially be built upon the algorithm we design for this preliminary problem.

Definition 5.3.1 (Robust mean estimation from batches)**.** There are m data providers indexed by $\{1, 2, \ldots, m\} =: [m]$. Among these providers, we denote the indices of uncorrupted (good) providers by $\mathcal{G} \subseteq [m]$ and the indices of corrupted (bad) providers by $\mathcal{B} = [m] \backslash \mathcal{G}$, where $|\mathcal{B}| = \alpha m$. Each provider $j \in [m]$ sends a data batch $x_j^{[n_j]} := \{x_j^1, \ldots, x_j^{n_j}\}$ to the server, where the batch size n_j can be arbitrary. For

$j \in \mathcal{G}$, its batch consists of i.i.d. samples drawn from the same σ-subGaussian distribution $\mathcal{D}$ with mean μ (i.e. $\mathbb{E}_{X\sim\mathcal{D}}[X] = \mu$ and $\mathbb{E}_{X\sim\mathcal{D}}\left[\exp\left(s\left(X-\mu\right)\right)\right] \leq \exp\left(\sigma^2 s^2/2\right)$, $\forall s \in \mathbb{R}$.). For $j \in \mathcal{B}$, $x_j^{[n_j]}$ can be arbitrary.

page 54 considers a robust learning problem from batches where we allow arbitrarily different batch sizes. The corruption level α is the fraction of bad providers not data points; it is possible that a bad provider j has an overwhelmingly large n_j compared to other providers. In contrast, prior works (Qiao and Valiant, 2017; Chen et al., 2020; Jain and Orlitsky, 2021) have only studied the setting with (roughly) equal batch sizes. In many real-world crowd-sourcing applications, large and small data providers can differ drastically in the amount of data they provide, so our framework above captures broader application scenarios than prior works.

For this problem, we propose the COW (clique-overweight) algorithm (page 56). Given the empirical means of the batches $\hat{\mu}_j := \frac{1}{n_j}\sum_{i=1}^{n_j} x_j^i$, $j = 1, \ldots, m$, batch sizes $n_1, \ldots, n_m$, subGaussian parameter σ, corruption level $\alpha < 1/2$, and confidence level $\delta > 0$, COW first constructs a confidence interval I_j for the true mean μ on page 56 using each batch j, where $I_j = \mathbb{R}$ if $n_j = 0$. With large probability, all good providers' intervals I_j should intersect because they contain μ. Define an undirected graph with nodes $I_1 \ldots I_m$, and I_i, I_j is connected by an edge if and only if $I_i \cap I_j \neq \varnothing$. Then we anticipate the good providers to form a large clique of size $(1-\alpha)m$. Accordingly, the algorithm finds the maximum clique in this graph. Of course, the maximum clique may contain some bad providers and miss some good providers. The second part of the algorithm reduces the influence of any "overweight" providers by cutting their effective batch size on page 56, thus preventing bad providers in the clique to overwhelm the final mean estimate on page 56.

There can be multiple maximum cliques in page 56; we break ties arbitrarily. A maximum clique can be computed efficiently.

We show that page 56 achieves the following guarantee.

Theorem 5.3.1. *Under page 54, if $n^{\text{cut}} > 0$ and $\alpha < \frac{1}{2}$, then with probability at least $1-\delta$,*

Algorithm 5 COW

Require: Batch empirical means: $\hat{\mu}_1, \dots, \hat{\mu}_m$; batch sizes: $n_1, \dots, n_m$; subGaussian parameter σ; corruption level α; confidence level δ

1: $I_j \leftarrow \left[\hat{\mu}_j - \frac{\sigma}{\sqrt{n_j}}\sqrt{2\log\frac{2m}{\delta}}, \hat{\mu}_j + \frac{\sigma}{\sqrt{n_j}}\sqrt{2\log\frac{2m}{\delta}}\right], \forall j \in [m]$

2: $C^* \leftarrow \operatorname{argmax}_{C\subseteq[m]:\bigcap_{j\in C} I_j \neq \varnothing} |C|$

3: $n^{\text{cut}} \leftarrow$ the $(2\alpha m + 1)$-th largest batch size

4: $\tilde{n}_j \leftarrow \min(n_j, n^{\text{cut}}), \forall j \in [m]$

5: **return** $\hat{\mu} \leftarrow \frac{1}{\sum_{j\in C^*}\tilde{n}_j}\sum_{j\in C^*}\tilde{n}_j\hat{\mu}_j$, Error $\leftarrow$ page 56

the estimation error $|\hat{\mu} - \mu|$ of $\hat{\mu}$ returned by page 56 satisfies:

$$\frac{2}{\sqrt{\sum_{j\in[m]}\tilde{n}_j}}\sigma\sqrt{2\log\frac{2}{\delta}} + \frac{8\alpha m\sqrt{n^{\text{cut}}}}{\sum_{j\in[m]}\tilde{n}_j}\sigma\sqrt{2\log\frac{2m}{\delta}} \tag{5.1}$$

where n^{cut} and $\tilde{n}_j$'s are defined in page 56 and page 56 in page 56.

A few remarks are in order.

Remark 5.3.1 (Compare to prior work)**.** Note that compared to prior works (Yin et al., 2018), our algorithm allows arbitrary batch sizes. Even if some agents report $n_j = 0$, as long as $n^{\text{cut}} > 0$, i.e. there are at least $2\alpha m + 1$ agents reporting non-zero n_j's, our estimator will return a well-behaved estimator. In contrast, algorithms designed for equal batches will provably fail if the batches are imbalanced. (Yin et al., 2018) calculates the trimmed-mean of the empirical means of each batch. Suppose the clean data distribution is Gaussian $N(\mu, 1)$ and $3\alpha m$ batches have size $n^* >> m > 1$ while the rest of the batches have size 1, then the error of trimmed-mean is $\tilde{O}\left(\frac{1}{\sqrt{m}} + \alpha\right)$, Importantly, $O\left(\frac{1}{\sqrt{m}}\right)$ is much larger than $\tilde{O}\left(\frac{1}{\sqrt{m+\alpha mn^*}}\right)$, the optimal statistical rate without data corruption. On the contrary, page 56 returns an estimation with error $\tilde{O}\left(\frac{1}{\sqrt{m+\alpha mn^*}} + \frac{\alpha mn^*}{m+\alpha mn^*}\frac{1}{\sqrt{n^*}}\right) \leq \tilde{O}\left(\frac{1}{\sqrt{n^*}}\right) << \tilde{O}\left(\frac{1}{\sqrt{m}}\right)$.

Remark 5.3.2 (Equal batch size case)**.** On the other hand, in case of equal batch sizes, i.e. $n_1 = \cdots = n_m = n$, page 56 becomes $O\left(\frac{\sigma}{\sqrt{n}}\left(\frac{1}{\sqrt{m}} + \alpha\sqrt{\log m}\right)\right)$. This recovers the rate in (Yin et al., 2018), which is optimal (up to logarithmic factors).

Therefore, our result strictly generalizes prior works on robust estimation from batches.

Remark 5.3.3 (Robust mean estimation v.s. robust mean estimation from batches)**.** In classical robust mean estimation setting (Huber, 1992; Diakonikolas et al., 2016), the optimal error rate is $O\left(\sigma\left(\alpha+\frac{1}{\sqrt{m}}\right)\right)$ given m total samples and α fraction corrupted samples. In contrast, due to having access to the data source ID, i.e. the batch indices, the learner can achieve significantly improved robustness. To see this, notice that the equal batch setting can be viewed as robust mean estimation from m data points $\hat{x}_j$'s. When the batch size n becomes larger, $\hat{x}_j$ has a smaller variance $\frac{\sigma^2}{n}$ and thus the error of robust mean estimation becomes $O\left(\frac{\sigma}{\sqrt{n}}\left(\alpha+\frac{1}{\sqrt{m}}\right)\right)$, which matches the above rate (up to logarithmic factors).

Remark 5.3.4 (Dependency on the largest batches)**.** Our bound in page 56 does not depend on the largest $2\alpha m$ n_j's. This implies that even if some clean agents have infinite samples, the algorithm cannot achieve an error that diminishes to zero. This might not look ideal at first glance, but we show this is inevitable information-theoretically. Interested readers are referred to page 196.

Remark 5.3.5 (Technical extensions)**.** When the good data batch is subject to point-wise perturbation of magnitude at most ϵ, a variant of Algorithm 5 (page 197 Pert-COW, see page 197) suffers at most a 2ϵ term in the error upper bound in addition to (5.1). In addition, page 56 does not require the exact dataset as input, but only the empirical mean and batch sizes of each data batch. As we shall see next, these two properties allow us to use Pert-COW in our byzantine-robust online RL algorithm to achieve low communication costs and preserve data privacy.

5.4 Byzantine-Robust RL in Parallel MDPS

Now, we are ready to study the problem of Byzantine-robust reinforcement learning in parallel *Markov Decision Processes* (MDPs). We consider a setting with one central server and m agents, α fraction of which may be adversarial. We postpone the precise interaction protocols between the server and agents to page 59 and page 62.

In both online and offline settings, we focus on finite horizon episodic tabular MDPs $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, H, \mu_1)$. Where $\mathcal{S}$ is the finite state space with $|\mathcal{S}| = S$; $\mathcal{A}$ is the finite action space with $|\mathcal{A}| = A$; $\mathcal{P} = \{P_h\}_{h=1}^H$ is the sequence of transition probability matrix, meaning $\forall h \in [H]$, $P_h : \mathcal{S} \times \mathcal{A} \mapsto \Delta(\mathcal{S})$ and $P_h(\cdot|s,a)$ specifies the state distribution in step $h+1$ if action a is taken from state s at step h; $\mathcal{R} = \{R_h\}_{h=1}^H$ is the sequence of bounded stochastic reward function, meaning $\forall h \in [H]$, $R_h(s,a)$ is the stochastic reward bounded in $[0,1]$ associated with taking action a in state s at step h; H is the time horizon; μ_1 is the initial state distribution. For simplicity, we assume μ_1 is deterministic and has probability mass 1 on state s_1.

Within each episode, the MDP starts at state s_1. At each step h, the agent observes the current state s_h and takes an action a_h and receives a stochastic reward $R_h(s_h, a_h)$. After that, the MDP transits to the next state s_{h+1}, which is drawn from $P_h(\cdot|s,a)$. The episode terminates after the agent takes action a_H in state s_H and receives reward $R_H(s_H, a_H)$ at step H.

A policy π is a sequence of functions $\{\pi_1, \ldots, \pi_H\}$, each maps from state space $\mathcal{S}$ to action space $\mathcal{A}$. The value function $V_h^\pi : \mathcal{S} \mapsto [0, H-h+1]$, is the expected sum of future rewards by taking action according to policy π, i.e. $V_h^\pi(s) := \mathbb{E}\left[\sum_{t=h}^H R_t(s_t, \pi_t(s_t)) \middle| s_h = s\right]$, where the expectation is w.r.t. to the stochasticity of state transition and reward in the MDP. Similarly, we define the state-action value function $Q_h^\pi : \mathcal{S} \times \mathcal{A} \mapsto [0, H-h+1]$:

$$Q_h^\pi(s,a) := \mathbb{E}\left[R_h(s,a)\right] + \mathbb{E}\left[\sum_{t=h+1}^H R_t(s_t, \pi_t(s_t)) \middle| s_h = s, a_h = a\right]$$

Let $\pi^* = \left\{\pi^h\right\}$ be an optimal policy and let $V_h^*(s) := V_h^{\pi^*}(s,a)$, $Q_h^*(s) := Q_h^{\pi^*}(s,a)$, $\forall h, s, a$.

For any $f : \mathcal{S} \mapsto [0, H]$, We define the Bellman operator by: $\left(\mathbb{B}_h f\right)(s,a) = \mathbb{E}\left[R_h(s,a)\right] + \mathbb{E}_{s' \sim P_h(\cdot|s,a)}[f(s')]$ Then the Bellman equation is given by:

$$V_h^\pi(s) = Q_h^\pi(s, \pi_h(s)) \tag{5.2}$$

$$Q_h^\pi(s,a) = \left(\mathbb{B}_h V_{h+1}^\pi\right)(s,a) \tag{5.3}$$

$$V_{H+1}^{\pi}(s) = 0. \tag{5.4}$$

The Bellman optimality equation is given by:

$$V_h^*(s) = \max_{a \in \mathcal{A}} Q_h^*(s,a) \tag{5.5}$$

$$Q_h^*(s,a) = \left(\mathbb{B}_h V_{h+1}^*\right)(s,a) \tag{5.6}$$

$$V_{H+1}^*(s) = 0. \tag{5.7}$$

We define the state distribution at step h by following policy π as $d_h^\pi(s) := P_h^\pi(s_h = s)$, and the state trajectory distribution of π as: $d^\pi := \{d_h^\pi\}_{h=1}^H$. The goal is to find a policy that maximizes the reward, i.e. find a $\hat{\pi}$, s.t. $V_1^{\hat{\pi}}(s_1) = V_1^*(s_1) = \max_\pi V_1^\pi(s_1)$. To measure the performance of our RL algorithms, we use suboptimality as our performance metric for offline settings and use regret as our performance metric for online settings. We formalize these two measures in their corresponding sections below.

5.5 Byzantine-Robust Online RL

In the online setting, we assume that a central server and m agents aim to collaboratively minimize their total regrets. The agents and server collaborate by following a communication protocol to decide when to synchronize and what information to communicate. Unlike the standard distributed RL setting, we assume α-fraction of the agents are Byzantine:

Definition 5.5.1 (Distributed online RL with Byzantine corruption)**.** There are m agents consisting of two types:

- $(1-\alpha)m$ **good agents, denoted by** $\mathcal{G}$: Each of the good agents interacts with a copy of $\mathcal{M}$ and communicates its observations to the server following the interaction protocol;

- αm **bad agents, denoted by** $\mathcal{B}$: The bad agents are allowed to communicate arbitrarily.

Because the server has no control over the bad agents, we only seek to minimize the error incurred by the good agents. Formally, we use regret as our performance measure for the online RL algorithm:

$$\text{Regret}(K) = \sum_{k=1}^{K} \sum_{j \in \mathcal{G}} \left(V_1^*(s_1) - V_1^{\pi_k^j}(s_1) \right), \tag{5.8}$$

where π_k^j is the policy used by agent j in episode k. At the same time, because of the distributed nature of our problem, we want to synchronize between the servers and agents only if necessary to reduce the communication cost.

Based on these considerations, we propose the Byzan-UCBVI algorithm (page 68). We highlight the following key features of Byzan-UCBVI:

1. **Low-switching-cost algorithm design**: the server will check the synchronization criteria in page 68 when receiving agent requests. Each good agent will request synchronization if and only if any of their own (s, a, h) counts doubles (page 68). Importantly, our agents do not need to know other agents' (s, a, h) counts to decide if synchronization is necessary. This design choice reduces the number of policy switches, synchronization rounds, and total communication costs, all from $O(K)$ to $O(\log K)$. Compared to the $O(\sqrt{K})$ communication steps in (Jadbabaie et al., 2022), ours is much lower. Unlike (Dubey and Pentland, 2021), our agents do not need to know other agents' transition counts to decide whether to synchronize.
2. **Homogeneous policy execution**: In any episode k, our algorithm is designed so that all good agents are running the same policy π_k. This ensures that the robust mean estimation achieves the smallest estimation error. Recall that the samples in the large batches are wasted if the batch sizes are severely imbalanced (cf. page 54).

3. **Robust UCBVI updates**: During synchronization, the central server performs policy update using a variant of the UCBVI algorithm (Azar et al., 2017): for $h = H, H-1, \ldots, 1$, compute:

$$\bar{Q}_h(\cdot,\cdot) = \left(\hat{\mathbb{B}}_h \hat{V}_{h+1}\right)(\cdot,\cdot) + \Gamma_h(\cdot,\cdot) \tag{5.9}$$

$$\hat{Q}_h(\cdot,\cdot) = \min\left\{\bar{Q}_h(\cdot,\cdot), H-h+1\right\}^+ \tag{5.10}$$

$$\hat{\pi}_h(\cdot) = \operatorname*{argmax}_a \hat{Q}_h(\cdot,a) \tag{5.11}$$

$$\hat{V}_h(\cdot) = \max_a \hat{Q}_h(\cdot,a). \tag{5.12}$$

The main difference lies in page 68, where we replace the standard mean and confidence interval estimation with our Pert-COW algorithm (page 197). Instead of estimating the transition matrix and reward function, we directly estimate the Bellman operator given an estimated value function $\hat{V}_{h+1}$. The server gathers sufficient statistics from agents in page 68. According to page 197, when $n^{\text{cut}} \leq 0$, the $\Gamma_h(s,a)$ in page 68 is set to be ∞ as a trivial error bound. page 68 adjust the bonus to be the range of the value function. The an additional ϵ is an adjustment for ϵ-cover argument in the proof of page 61.

We are now ready to present the following regret bound for Byzan-UCBVI.

Theorem 5.5.1 (Regret bound)**.** *Under page 59, if $\alpha \leq \frac{1}{3}\left(1-\frac{1}{m}\right)$, for all $\delta < \frac{1}{4}$, with probability at least $1-3\delta$, the total regret of page 68 is at most*

$$\sum_{k=1}^{K}\sum_{j\in\mathcal{G}}\left(V_1^*(s_1) - V_1^{\hat{\pi}_k^j}(s_1)\right) \tag{5.13}$$

$$= \tilde{O}\left((1+\alpha\sqrt{m})H^2 S\sqrt{AmK\log(1/\delta)}\right). \tag{5.14}$$

Remark 5.5.1 (Understanding the regret bound)**.** In page 68, the good agents are using the same policy, and thus for all $j \in \mathcal{G}$, $\hat{\pi}_k^j = \hat{\pi}_k$, where $\hat{\pi}_k$ is the policy calculated by the server in k-th episode. By utilizing the batch structure, page 68 achieves a regret sublinear in K, even under Byzantine attacks. Our regret is only

$O(\sqrt{mK} + \alpha m\sqrt{K}))$ compared to the $O(m\sqrt{K} + m\alpha^{1/4}K^{3/4})$ regret in (Jadbabaie et al., 2022). When $\alpha \leq 1/\sqrt{m}$, the dominating term $\sqrt{mK}$ is optimal even in the clean setting (Azar et al., 2017).

Remark 5.5.2 (The Breakdown point)**.** We require α to be smaller than $\frac{1}{3}$ because we can show that with high probability, all of the good agents will have visitation on some (s, a) pair and simply restricting $\alpha \leq \frac{1}{3}$ ensures the n^{cut} in page 197 is greater than 0, which meets the requirement in page 55 and allows for a cleaner exposition of page 61.

Remark 5.5.3 (Communication cost)**.** Because each agent runs K episodes in total, the count of each of the (s, a, h) tuples doubles at most $\lfloor \log_2 K \rfloor$ times during training. Thus each good agent will send at most $SAH\lfloor \log_2 K \rfloor$ sync requests. The bad agents can only send a logarithmic number of effective requests because of the checking step in page 68. As a result, there will be at most $mSAH\lfloor \log_2 K \rfloor$ synchronization episodes in total. The communication inside one synchronization episode includes the following: at least one agent sends a sync request; inside the value iteration, the server will send estimated value functions at H steps to each agent; Each good agent will send the estimated Bellman operator for each (s, a) pair at H steps and the counts to the server. Importantly, the agents only need to send summary statistics instead of the raw dataset to the server. This preserves the data privacy of individual agents (Sakuma et al., 2008; Liu et al., 2019).

Remark 5.5.4 (Switching cost)**.** Switching cost measures the number of policy changes. Algorithms with low switching costs are favorable in real-world applications (Bai et al., 2019; Zhang et al., 2020b; Gao et al., 2021). page 68 only performs policy updates during synchronization episodes. Its switching cost is thus at most $mSAH\lfloor \log_2 K \rfloor$.

5.6 Byzantine-Robust Offline RL

In the offline setting, we assume the server has access to logged interaction data from many agents, among which some are adversarial. The goal of the server is to

find a nearly optimal policy using this collection of offline datasets without further interaction with the environment. Specifically:

Definition 5.6.1 (Distributed offline RL with Byzantine corruption)**.** The server has access to an offline data set with m data batches $\bigcup_{j\in[m]} D_j$, including $(1-\alpha)m$ good batches $\mathcal{G}$ and αm bad batches $\mathcal{B}$, where

$$D_j := \bigcup_{h\in[H]} D_j^h := \bigcup_{h\in[H]} \left\{\left(s_h^{j,k}, a_h^{j,k}, r_h^{j,k}, {s'}_h^{j,k}\right)\right\}_{k=1}^{K_j}.$$

We make an assumption on the data generating process similar to (Wang et al., 2020a). Specifically, for all $j \in \mathcal{G}$, D_j is drawn from an unknown distribution $\left\{\nu_h^j\right\}_{h=1}^H$, where for each $h \in [H]$, $\nu_h^j \in \Delta(\mathcal{S}\times\mathcal{A})$. For all h, j, k, $\left(s_h^{j,k}, a_h^{j,k}\right) \sim \nu_h^j$, ${s'}_h^{j,k} \sim P_h(\cdot|s_h^{j,k}, a_h^{j,k})$ and $r_h^{j,k}$ is an instantiation of $R_h\left(s_h^{j,k}, a_h^{j,k}\right)$. For any $j \in \mathcal{B}$ (i.e. bad batches), D_j can be arbitrary.

The performance is measured by the suboptimality w.r.t. a deterministic comparator policy $\tilde{\pi}$ (not necessarily an optimal policy):

$$\mathrm{SubOpt}(\pi, \tilde{\pi}) := V_1^{\tilde{\pi}}(s_1) - V_1^{\pi}(s_1). \tag{5.15}$$

In the offline setting, the server cannot interact with the MDP. So our result relies heavily on the quality of the dataset. As we will see in the analysis, the suboptimality gap page 63 can be upper bounded by the estimation error of the Bellman operator along the trajectory of $\tilde{\pi}$. As a result, we do not need full coverage over the whole state-action space. Instead, we only need the offline dataset to have proper coverage over $\{d_h^{\tilde{\pi}}\}_{h=1}^H$, the state distribution of policy $\tilde{\pi}$ at each step h. To characterize the data coverage, for any s, a, h, we define the counts on (s, a, h) tuples by:

$$N_h^j(s,a) := \sum_{k\in[K_j]} \mathbf{1}\left\{(s_h^{j,k}, a_h^{j,k}) = (s,a)\right\}, \quad \forall j \in [m]. \tag{5.16}$$

When calling page 56, the large data batches might be clipped in page 56. By definition, the clipping threshold is bounded between $N_h^{\mathcal{G},\mathrm{cut}_1}(s,a)$, the $(\alpha m+1)$-th

largest of $\left\{N_h^j(s,a)\right\}_{j\in\mathcal{G}}$ and $N_h^{\mathcal{G},\text{cut}_2}(s,a)$, the $(2\alpha m+1)$-th largest of $\left\{N_h^j(s,a)\right\}_{j\in\mathcal{G}}$. We define three quantities $p^{\mathcal{G},0}, \kappa, \kappa_{\text{even}}$ to characterize the quality of the offline dataset. The first quantity describes the density of $\tilde{\pi}$ trajectory that is not properly covered by the offline dataset:

Definition 5.6.2 (Measure of insufficient coverage)**.** We define $p^{\mathcal{G},0}$ as the probability of $\tilde{\pi}$ visiting an (s,h,a) tuple that is insufficiently covered by the logged data, namely

$$p^{\mathcal{G},0} := \sum_{h=1}^{H} \mathbb{E}_{d_h^{\tilde{\pi}}}\left[\mathbf{1}\left\{N_h^{\mathcal{G},\text{cut}_2}(s,\tilde{\pi}(s))=0\right\}\right]. \tag{5.17}$$

Recall that page 56 requires there are at least $(2\alpha m+1)$ non-empty data batches to make an informed decision. $p^{\mathcal{G},0}$ measures an upper bound on the total probability under $d^{\tilde{\pi}}$ to encounter an (s,h,a) on which COW cannot return a good mean estimator.

We now introduce κ, the density ratio between the $d^{\tilde{\pi}}$, and the empirical distribution of the uncorrupted offline dataset. κ quantifies the portion of useful data in the whole dataset and is commonly used in the offline RL literature (Rashidinejad et al., 2021; Zhang et al., 2021a). We only focus on the (s,a,h) tuples excluded by $p^{\mathcal{G},0}$ in page 64:

Definition 5.6.3 (density ratio)**.** We use $\{\mathcal{C}_h\}_{h=1}^H$ to denote the state space (in the support of $\{d_h^{\tilde{\pi}}\}_{h=1}^H$) that have proper clean agents coverage:

$$\mathcal{C}_h = \left\{s | N_h^{\mathcal{G},\text{cut}_2}(s,\tilde{\pi}(s)) > 0\right\}. \tag{5.18}$$

We use κ to denote the density ratio between the state distribution of policy $\tilde{\pi}$ and the empirical distribution over the uncorrupted offline dataset:

$$\kappa := \max_{h\in[H]} \max_{s\in\mathcal{C}_h} \frac{d_h^{\tilde{\pi}}(s)}{\sum_{j\in\mathcal{G}} N_h^j(s,\tilde{\pi}_h(s))/\sum_{j\in\mathcal{G}} K_j}. \tag{5.19}$$

As we can see in page 55, the accuracy of page 56 heavily depends on the evenness of the batches. We define the following quantity to measure the information

loss in the clipping step (page 56 in page 56):

Definition 5.6.4 (Unevenness of good agents coverage)**.**

$$\kappa_{\text{even}} := \max_{h\in[H]} \max_{s\in\mathcal{C}_h} \frac{\sum_{j\in\mathcal{G}} N_h^j(s,\tilde{\pi}_h(s))}{\sum_{j\in\mathcal{G}} \tilde{N}_h^{j,\text{cut}_2}(s,\tilde{\pi}_h(s))} \tag{5.20}$$

$$\cdot \frac{m(1-\alpha)N_h^{\mathcal{G},\text{cut}_1}(s,\tilde{\pi}_h(s))}{\sum_{j\in\mathcal{G}} \tilde{N}_h^{j,\text{cut}_2}(s,\tilde{\pi}_h(s))}, \tag{5.21}$$

where $\tilde{N}_h^{j,\text{cut}_2}(s,\tilde{\pi}_h(s)) = \max\left(N_h^{\mathcal{G},\text{cut}_2}(s,\tilde{\pi}_h(s)), N_h^j(s,\tilde{\pi}_h(s))\right)$.

Intuitively, κ_{even} describes the unevenness of good agent coverage. It takes into account both how much data in large batches are cut off by the clipping step and the unevenness of the batches after clipping. We include $N_h^{\mathcal{G},\text{cut}_1}(s,\tilde{\pi}_h(s))$ and $N_h^{\mathcal{G},\text{cut}_2}(s,\tilde{\pi}_h(s))$, instead of the true clipping threshold, meaning κ_{even} serves as an upper bound of the actual unevenness resulting from running the algorithm. For example, suppose $\alpha m > 1$: if for any s,a,h,j, $N_h^j(s,a) = n$, then $\kappa_{\text{even}} = 1$; if for any s,a,h, there is one good data batch with size Lm for some $L > 1$ while the others have size 1, then $N_h^{\mathcal{G},\text{cut}_1}(s,a) = N_h^{\mathcal{G},\text{cut}_2}(s,a) = 1$ and $\kappa_{\text{even}} = \frac{Lm+(1-\alpha)m-1}{(1-\alpha)m}\frac{(1-\alpha)m}{(1-\alpha)m} \approx L+1$, meaning κ_{even} increases as the batches become less even.

Remarkably, all three quantities defined above only depend on the (s,a,h) counts of the good data batches.

Given the above setup, we now present our second algorithm, Byzan-PEVI, a Byzantine-Robust variant of pessimistic value iteration (Jin et al., 2021). Similar to the online setting, we use our COW (without perturbation) algorithm to approximate the Bellman operator and use the estimation error to design the PESSIMISTIC bonus for the value iteration. Byzan-PEVI (page 69) runs pessimistic value iteration (page 65-page 66) and calls COW as a subroutine to robustly estimate the Bellman operator using offline dataset D:

$$\bar{Q}_h(\cdot,\cdot) = \left(\hat{\mathbb{B}}_h\hat{V}_{h+1}\right)(\cdot,\cdot) - \Gamma_h(\cdot,\cdot) \tag{5.22}$$

$$\hat{Q}_h(\cdot,\cdot) = \min\left\{\bar{Q}_h(\cdot,\cdot), H-h+1\right\}^+ \tag{5.23}$$

$$\hat{\pi}_h(\cdot) = \operatorname*{argmax}_a \hat{Q}_h(\cdot, a) \tag{5.24}$$

$$\hat{V}_h(\cdot) = \max_a \hat{Q}_h(\cdot, a). \tag{5.25}$$

Theorem 5.6.1. *Given any deterministic comparator policy $\tilde{\pi}$, under page 63, 5.6.2, 5.6.3 and 5.6.4: for any δ, $\alpha < \frac{1}{3}$, with probability at least $1-\delta$, page 69 outputs a policy $\hat{\pi}$ with:*

$$\begin{aligned} &\mathrm{SubOpt}\left(\hat{\pi}, \tilde{\pi}\right) \leq 2Hp^{\mathcal{G},0} \\ &+ O\left(\sqrt{\kappa\kappa_{even}} H^2 \sqrt{S} \frac{1+\sqrt{m}\alpha}{\sqrt{\sum_{j\in\mathcal{G}} K_j}} \sqrt{\log \frac{HSAm}{\delta}}\right). \end{aligned} \tag{5.26}$$

Remark 5.6.1 (Understanding the sub-optimality gap)**.** The sub-optimality gap page 66 depends on both the offline data distribution (characterized by $p^{\mathcal{G},0}$, κ and κ_{even}) and number of clear samples $\sum_{j\in\mathcal{G}} K_j$. The first term only depends on the coverage of the data distribution and will not shrink with a larger sample size. When for each (s,a,h), all agents visit the tuple for equal times, we have $\kappa_{\text{even}} = 1$. Furthermore, let $K_j = K$ for all $j \in [m]$, RHS of page 66 becomes:

$$\begin{aligned} &2Hp^{\mathcal{G},0} \\ &+O\left(\sqrt{\kappa} H^2 \sqrt{S} \frac{1}{\sqrt{mK}} \sqrt{\log \frac{HSAm}{\delta}}\right) \\ &+O\left(\sqrt{\kappa} H^2 \sqrt{S} \frac{\alpha}{\sqrt{K}} \sqrt{\log \frac{HSAm}{\delta}}\right), \end{aligned}$$

where the first term measures the effect of lack of coverage, the second term is the statistical error and the third term is the bias term due to the data corruption. Importantly, both the second and the third terms vanish as $K \to \infty$, whereas the first term is due to the lack of data coverage. On the contrary, (Zhang et al., 2021a) has a non-diminishing bias term due to data corruption. To the best of our knowledge, this is the first result for Byzantine-robust offline RL.

Remark 5.6.2 (Offline v.s. online RL)**.** Our offline RL results are more involved and notation-heavy due to the nature of the problem. In the offline RL setting, the learner cannot control the data-generating process, and each data source can be arbitrarily different. The agent can only passively rely on the robust mean estimator we designed and the pessimism principle to learn as well as the data permits. In contrast, the learner has complete control over the clean agents' data collection process in the online setting. Our algorithm Byzan-UCBVI enables the server to realize its full potential and obtain a tighter and cleaner sample complexity guarantee.

5.7 Conclusion

To summarize, in this work, we start by presenting COW, a robust mean estimation algorithm for learning from uneven batches. Building upon COW, we propose byzantine-robust online (Byzan-UCBVI) and the first byzantine-robust offline (Byzan-PEVI) reinforcement learning algorithms in the distributed setting. Several questions remain open: (1) Can we provide a complete characterization of the information-theoretical lower bound for robust mean estimation from uneven batches? (2) Can we extend our RL algorithms to the function approximation setting? Allowing function approximation is essential to apply our algorithm to empirical evaluations. However, this would require a computationally efficient high-dimensional robust mean estimator from uneven batches, which is highly nontrivial. Therefore, we defer the generalization to the function approximation setting and empirical evaluation of our framework as an important direction for future research.

Algorithm 6 BYZAN-UCBVI (K, δ, α)

[S]$\hat{V}_{H+1}(\cdot) \leftarrow 0$, $\hat{Q}_{H+1}(\cdot,\cdot) \leftarrow 0$, $\text{SyncCount}_j \leftarrow -1, \forall j \in [m]$, $\text{Sync}_j \leftarrow \text{TRUE}, \forall j \in [m]$ $\delta' \leftarrow \frac{\delta}{(SAHKm)^{3S}}$, $\epsilon \leftarrow \frac{1}{SAHKm}$ # We use [S] to denote the action of central server

[A]$N_h^j(s,a) \leftarrow 0$, $D_h^j \leftarrow \varnothing$, $\forall (j,h,s,a) \in \mathcal{G} \times [H] \times \mathcal{S} \times \mathcal{A}$ # We use [A] to denote the action of agents

for episode $k \in [K]$ **do**
  [S] Receive $\text{Sync}_1, \text{Sync}_2, \ldots, \text{Sync}_m$
  for agent $j \in [m]$ **do**
    if Sync_j and $\text{SyncCount}_j \leq SAH \log_2 K$ **then**
      [S] $\text{SyncCount}_j \leftarrow \text{SyncCount}_j + 1$
      [S] SYNCHRONIZE $\leftarrow$ TRUE
    end if
  end for
  if SYNCHRONIZE **then**
    [A] $N_{h,j}^{\text{old}}(s,a) \leftarrow N_h^j(s,a), \forall s,a,h,j$
    for $h = H, H-1, \ldots, 1$ **do**
      [S] Communicate $\hat{V}_{h+1}(\cdot)$ to each agent
      for $(s,a) \in \mathcal{S} \times \mathcal{A}$ **do**
        [A] send $x_j \leftarrow \frac{\sum_{(s,a,r,s') \in D_h^j} r + \hat{V}_{h+1}(s')}{N_h^j(s,a)}$, $n_j \leftarrow N_h^j(s,a)$ to Server, $\forall j \in \mathcal{G}$
        [S] $\left(\hat{\mathbb{B}}_h \hat{V}_{h+1}\right)(s,a), \Gamma_h(s,a) \leftarrow$ PERT-COW $\left(x_{[m]}, n_{[m]}, H-h+1, \alpha, \epsilon, \delta'\right)$
        $\Gamma_h(s,a) \leftarrow \min(H-h+1, \Gamma_h(s,a) + \epsilon)$
      end for
      [S] Compute $\bar{Q}_h, \hat{Q}_h, \hat{\pi}_h, \hat{V}_h$ as in page 61-page 61.
    end for
  end if
  [S] SYNCHRONIZE $\leftarrow$ FALSE
  for $j \in \mathcal{G}$ **do**
    [A] $\text{Sync}_j \leftarrow$ FALSE
    [A] Sample $\left\{(s_h^{j,k}, a_h^{j,k}, r_h^{j,k}, s_{h+1}^{j,k})\right\}_{h \in [H]}$ under $\{\hat{\pi}_h\}_{h=1}^H$
    [A] $\forall h$, $N_h^j(s_h^{j,k}, a_h^{j,k}) \leftarrow N_h^j(s_h^{j,k}, a_h^{j,k}) + 1$, $D_h^j \leftarrow D_h^j \cup \left\{(s_h^{j,k}, a_h^{j,k}, r_h^{j,k}, s_{h+1}^{j,k})\right\}$
    [A] Send Sync request to Server, if $\text{Sync}_j \leftarrow \mathbf{1}\left\{\max_{s,a,h} \frac{N_h^j(s,a)}{N_{h,j}^{\text{old}}(s,a)} \geq 2\right\}$ is TRUE.
  end for
end for
return $\{\hat{\pi}_h\}_{h=1}^H$

Algorithm 7 BYZAN-PEVI

Require: $D := \bigcup_{j\in[m]} D_j := \bigcup_{h\in[H]} D_j^h := \bigcup_{h\in[H]} \left\{\left(s_h^{j,k}, a_h^{j,k}, r_h^{j,k}, s_h'^{j,k}\right)\right\}_{k=1}^{K_j}$, α, δ

$\delta' \leftarrow \frac{\delta}{H|\mathcal{S}||\mathcal{A}|m}$
$\hat{V}_{H+1}(\cdot) \leftarrow 0$
for $h = H, H-1, \ldots, 1$ **do**
  $\sigma \leftarrow H - h + 1$
  for $(s,a) \in \mathcal{S} \times \mathcal{A}$ **do**
    for $j \in \mathcal{G}$ **do**
      $n_j \leftarrow \sum_{k\in[K_j]} \mathbf{1}\left\{(s_h^{j,k}, a_h^{j,k}) = (s,a)\right\}$
      $x_j \leftarrow \frac{1}{N_h^j(s,a)} \sum_{(s,a,r,s')\in D_h^j} \left(r + \hat{V}_{h+1}(s')\right)$
    end for
    if $|j \in [m] : n_j > 0| \geq 2\alpha m + 1$ **then**
      $\left(\hat{\mathbb{B}}_h \hat{V}_{h+1}\right)(s,a), \Gamma_h(s,a) \leftarrow \text{COW}\left(x_{[m]}, n_{[m]}, \sigma, \alpha, \delta'\right)$
    else
      $\left(\hat{\mathbb{B}}_h \hat{V}_{h+1}\right)(s,a) \leftarrow 0$, $\Gamma_h(s,a) \leftarrow H - h + 1$
    end if
  end for
  Compute $\bar{Q}_h, \hat{Q}_h, \hat{\pi}_h, \hat{V}_h$ as in page 65-page 66.
end for
return $\{\hat{\pi}_h\}_{h=1}^H$

We present suboptimality upper bounds in offline reinforcement learning as the worst-case analysis in the previous chapters. Those bounds hold universally regardless of the environments. However, those results can be overly conservative. In this chapter, we develop a refined instance-dependent analysis and show that under certain conditions, a generalized Pessimistic Value Iteration (PEVI) outputs the optimal policy even with the presence of data corruption and heavy-tailed reward distribution.

6.1 Introduction

Previous studies have primarily focused on problems under certain concentration assumptions, typically requiring that the rewards are bounded or follow a distribution with subGaussian tails Lattimore and Szepesvári (2020). However, there is growing evidence indicating that the subGaussianity assumption may not hold for many real-world scenarios Arnold (2014); Liebeherr et al. (2012); Borak et al. (2005), challenging the applicability of algorithms designed solely for sub-Gaussian settings.

In terms of data corruption in RL, prior work Zhang et al. (2022); Chen et al. (2022) showed that one can apply pessimistic value iteration (PEVI) with robust mean estimation to partially handle data corruption in offline RL, resulting in a policy $\hat{\pi}$ with suboptimality upper bound $\mathrm{SubOpt}(\hat{\pi}) \leq \tilde{O}\Big(\frac{\mathrm{poly}(H,\sigma)}{\sqrt{N}}\Big) + O(H\sigma\epsilon)$. Such an upper bound involves a term diminishing with sample size N and an irreducible bias term involves the corruption level ϵ. This implies that PEVI returns a suboptimal policy even with infinite data.

In this paper, we address the challenge of policy recovery in the presence of both heavy-tailed reward distributions and data corruption. We establish that Trimmed-mean estimation achieves the optimal error rate of $O\Big(\sigma\epsilon^{\frac{\gamma}{1+\gamma}} + \sigma N^{-\frac{\gamma}{1+\gamma}}\Big)$ for the robust mean estimation problem when confronted data corruption and

heavy-tailed distribution. When using Trimmed-mean estimation as a subroutine, PEVI generates a nearly optimal policy. In particular, by utilizing the property of action gap, we show that $O(H\sigma\epsilon^{\frac{\gamma}{1+\gamma}} + \epsilon H) < \Delta^{\mathcal{A}}_{\min}$ is sufficient for the policy to achieve the optimal value even under corruption. We summarize our contributions as follows:

1. We show that a modified version of Trimmed-Mean estimation achieves *minimax-optimal* error guarantee for robust mean estimation problems with heavy-tailed distribution and data corruption. Importantly, we only require the distribution to have *bounded* $(1+\gamma)$*-th centered moment* and allow the variance of the distribution to be infinite. Unlike the truncated empirical mean estimation in Bubeck et al. (2013), the trimmed mean estimator considered in our paper is both translation-invariant and robust to data corruption. As a result, we show that reward distribution with bounded $(1+\gamma)$-th moment is sufficient to ensure the success of policy learning, which is a much weaker concentration assumption than the subGaussian or bounded variance assumption typically used in the literature.
2. We present a generalized PEVI and derive an optimality condition based on the action gap. In the offline learning setting with *heavy-tailed reward* and *data corruption*, we plug in the trimmed mean estimation for reward estimation. We show that given sufficient samples, $O(H\sigma\epsilon^{\frac{\gamma}{1+\gamma}} + \epsilon H) < \Delta^{\mathcal{A}}_{\min}$ ensures that the learner takes an optimal action in each state visited by some optimal policy and thus achieves the *optimal value*.

6.2 Related Work

RL and adversarial attack against RL: Reinforcement learning aims to find the optimal strategy in a Markov Decision Process (MDP) Sutton and Barto (2018). In online RL, Azar et al. (2017); Dann et al. (2017) show that the UCB-style algorithm achieves minimax regret bound. In offline RL, Jin et al. (2021); Rashidinejad et al. (2021); Xie et al. (2021) use the pessimistic principle to design algorithms for offline policy learning. There are lines of work studying gap-dependent online Simchowitz

and Jamieson (2019); Xu et al. (2021a); Dann et al. (2021); Jonsson et al. (2020); Wagenmaker et al. (2022) and offline Wang et al. (2022); Hu et al. (2021) RL. Our paper is closely related to the work on offline gap-dependent RL. However, our main objective is to characterize sufficient conditions for optimality under data corruption instead of optimal sample complexity.

Heavy-tailed bandits: There is a significant body of research dedicated to studying bandit problems under weak moment assumptions. For instance, Bubeck et al. (2013) focused on the mean multi-armed bandit (MAB) problem with heavy-tailed rewards and utilized robust mean estimation to develop a UCB algorithm that achieves logarithmic regret. The pure-exploration problem for MAB with heavy-tailed distributions was investigated by Yu et al. (2018). Furthermore, Medina and Yang (2016); Shao et al. (2018) explored the linear bandit problem with heavy-tailed noise distributions and proposed algorithms with nearly-optimal regret guarantees. Dubey et al. (2020) examined this problem in the context of cooperative multi-agent settings.

Robust statistics: Robust statistics studies estimation with corrupted data Huber (1992); Tukey (1960). Recent advances Diakonikolas et al. (2019a); Lai et al. (2016) design efficient algorithms for high-dimensional robust statistics. These techniques are applied to more general machine learning tasks, including linear regression Diakonikolas et al. (2019c), supervised learning Diakonikolas et al. (2019b); Prasad et al. (2018) and RL Zhang et al. (2022, 2021b). Our work utilizes robust mean estimation to defend data corruption in offline RLs.

Adversarial RL and robust RL: RL is vulnerable to adversarial attacks Ma et al. (2019); Zhang et al. (2020a); Huang et al. (2017); Sun et al. (2020); Behzadan and Munir (2017). Corruption robust RL performs policy learning under data corruption Lykouris et al. (2021); Wei et al. (2022); Zhang et al. (2021b, 2022); Chen et al. (2022), which usually results in a bias term in the performance guarantee due to the data corruption. Niss and Tewari (2020); Kapoor et al. (2019) study multi-armed bandits under data corruption using robust statistics. They show that if the corruption level is not high enough to make the robust reward estimation of a suboptimal to be larger than that of an optimal arm, then the learner suffers

only sublinear regret, which captures an optimal arm. We use this intuition to study offline RL under data corruption. There is a separate line of works studying distributionally robust RL problem Shi and Chi (2022); Panaganti et al. (2022) where the state transition is specified by some uncertainty sets. Our setting is significantly different from this line of works.

6.3 Preliminary

MDP formulation: We consider a finite horizon episodic tabular Markov Decision Process (MDP) $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, H, p_0)$ with finite state space $|\mathcal{S}| = S$, finite action space $|\mathcal{A}| = A$, transition matrices $\mathcal{P} = \{P_h\}_{h=1}^H$, reward distributions $\mathcal{R} = \{\mathcal{R}_h\}_{h=1}^H$, and initial state distribution p_0. We assume the rewards are scholastic and the expectations of reward distributions are bounded in $[0, 1]$, i.e. for all $(s, a, h) \in \mathcal{S} \times \mathcal{A} \times [H]$, $r_h(s, a) := \mathbb{E}_{R_h(s,a)\sim\mathcal{R}_h(s,a)}[R_h(s, a)] \in [0, 1]$. Later on, we will study MDPs with different concentration assumptions on the reward distributions.

Policy and value function: A policy $\pi = \{\pi_h\}_{h=1}^H$ from a deterministic policy class Π is a sequence of deterministic functions that map from state to action: $\pi_h : \mathcal{S} \mapsto \mathcal{A}, \forall h$. The state value function of π is defined as

$$V_h^\pi(s) := \mathbb{E}\left[\sum_{t=h}^{H} R_t(s_t, \pi_t(s_t)) \middle| s_t = s\right].$$

We similarly define the state-action value function:

$$Q_h^\pi(s, a) := \mathbb{E}[R_h(s, a)] + \mathbb{E}_{s_{h+1}\sim P_h(\cdot|s,a)}\left[V_{h+1}^\pi(s_{h+1})\right].$$

The value of a policy is the expectation of $V_1^\pi(s)$ over the initial state distribution: $V_{p_0}^\pi := \mathbb{E}_{s_1\sim p_0}[V_1^\pi(s_1)]$. An *optimal policy* is one that simultaneously maximizes $V_h^\pi(s)$ for all h and s. We use $\Pi^* \subseteq \Pi$ to denote the set of all deterministic optimal policies. And we use $V_h^*(\cdot)$, $Q_h^*(\cdot, \cdot)$, $V_{p_0}^*$ to denote the state value function, state-action value function, and value of the optimal policies. We use $d_h^\pi(s) := \mathbb{E}_\pi[\mathbb{I}\{s_h = s\}]$, $d_h^\pi(s, a) := \mathbb{E}_\pi[\mathbb{I}\{(s_h, a_h) = (s, a)\}]$ to denote the state occupancy distribution and

state-action occupancy distribution under policy π.

Performance measure: In this paper, we mainly focus on the offline setting and use the suboptimality gap as the performance measure for a policy: $\mathrm{SubOpt}(\pi) := V_{p_0}^* - V_{p_0}^\pi$. Our goal is to find a policy with a small suboptimality gap.

Policy gap and action gap: Among policies that fail to achieve the optimal value, the best one has the smallest suboptimality gap. We call this gap the *policy gap*: $\Delta_{\min}^{\Pi} := \min_{\pi\in\Pi: V_{p_0}^\pi < V_{p_0}^*} \mathrm{SubOpt}(\pi)$. In contrast, we define a more fine-grained *action gap* by $\Delta_{\min}^{\mathcal{A}} := \min_{(h,s,a):\Delta_h(s,a)>0, s\in\mathcal{S}_h} \Delta_h(s,a)$, where $\Delta_h(s,a) := V_h^*(s) - Q_h^*(s,a)$ and $\mathcal{S}_h := \left\{s \in \mathcal{S} : \exists \pi^* \in \Pi^*, \text{ s.t. } d_h^{\pi^*}(s) > 0\right\}$. For notation convenience we assume there is at least one (s,a,h) tuple s.t. $s \in \mathcal{S}_h$ and $\Delta_h(s,a) > 0$ to exclude trivial MDPs. A similar notion of $\Delta_{\min}^{\mathcal{A}}$ has been introduced in Simchowitz and Jamieson (2019); Wang et al. (2022), Our notation of $\Delta_{\min}^{\mathcal{A}}$ is a refinement over theirs where the minimum is over *only the* (s,h) *pairs covered by at least an optimal policy*. We can show that our action gap is always no less than policy gap, and the difference can be large:

Proposition 6.3.1. For any MDP $\mathcal{M}$, there exists $(\pi^*, s', h') \in \Pi^* \times \mathcal{S} \times [H]$, s.t.

$$d_{h'}^{\pi^*}(s') > 0, \text{ and } \Delta_{\min}^{\Pi} \le d_{h'}^{\pi^*}(s')\Delta_{\min}^{\mathcal{A}} \le \Delta_{\min}^{\mathcal{A}}.$$

Intuitively, by definition of $\Delta_{\min}^{\mathcal{A}}$, there exists a (s', a', h') tuples and an optimal policy π^* s.t. $\Delta_{h'}(s', a') = \Delta_{\min}^{\mathcal{A}}$ and $d_{h'}^{\pi^*}(s') > 0$. We can design a suboptimal policy $\tilde{\pi}$ by choosing the suboptimal action a' at state s' and step h' and follow π^* in all other states or steps. The suboptimality of $\tilde{\pi}$, $d_{h'}^{\pi^*}(s')\Delta_{\min}^{\mathcal{A}}$, depends on the state occupancy measure $d_{h'}^{\pi^*}(s')$. Because $\Delta_{\min}^{\Pi}$ is a lower bound on the suboptimality of all suboptimal policies, we conclude that $\Delta_{\min}^{\Pi} \le d_{h'}^{\pi^*}(s')\Delta_{\min}^{\mathcal{A}}$. $d_{h'}^{\pi^*}(s')$ can be very close to 0 in some MDPs, thus $\Delta_{\min}^{\Pi}$ can be much smaller than $\Delta_{\min}^{\mathcal{A}}$.

6.4 Sufficient Condition for Exact Optimal Policy Recovery in Offline RL

In this section, we provide a sufficient condition for exact optimal policy recovery in offline RL. Our characterization is based on the well-known PEVI algorithm Jin et al. (2021), we slightly generalize it in Algorithm 8 to decouple RL from the estimators on mean rewards and transitions. This enables us to plug in different estimators later based on specific data assumptions, such as when the data is drawn from heavy-tailed distributions or adversarially corrupted. We then achieve different exact optimal policy recovery guarantees accordingly. Concretely, Algorithm 8 calls a RewardEstimator f to obtain a confidence interval $\hat{r}_h(s,a) \pm b_h^1(s,a)$ for the reward $r_h(s,a)$, and a TransitionEstimator g to obtain a confidence interval $\widehat{\mathrm{PV}}_{h,s,a} \pm b_h^2(s,a)$ for the expectation of a vector $\underline{V}_h$ under the transition multinomial $P_{h,s,a}$. These estimators will be instantiated differently in Section 6.5 based on different data assumptions. The notation $\mathcal{D}_{r|hsa}$ stands for the set of reward values observed at stage h in state s under action a in the offline dataset; similarly for the set of next states $\mathcal{D}_{s'|hsa}$.

If the sum of confidence bound $b_h^1(s,a)+b_h^2(s,a)$ is uniformly bounded on (s,a,h) tuples that are covered by the optimal policies, we can get a clean suboptimality guarantee for Algorithm 8:

Theorem 6.4.1 (Bound on suboptimality)**.** *Suppose for all $(s,a,h) \in \mathcal{S}\times\mathcal{A}\times[H]$, with probability at least $1-\frac{\delta}{SAH}$, we have:*

$$|\hat{r}_h(s,a) - r_h(s,a)| \le b_h^1(s,a)$$
$$\left|\widehat{\mathrm{PV}}_{h,s,a} - P_{h,s,a}^\top \underline{V}_{h+1}\right| \le b_h^2(s,a).$$

If $\forall (s,a,h) \in \left\{(s,a,h) : \exists \pi^* \in \Pi^*,\ s.t.\ d_h^{\pi^*}(s,a) > 0\right\}$, we have $b_h^1(s,a) + b_h^2(s,a) \le b$, then with probability at least $1-\delta$, $\hat{\pi}$ returned by Algorithm 8 satisfies

$$\mathrm{SubOpt}(\hat{\pi}) \le 2Hb. \tag{6.1}$$

Algorithm 8 Generalized PEVI

Input: dataset $\mathcal{D} = \bigcup_{h=1}^{H} \left\{ \left(s_{h,i}, a_{h,i}, r_{h,i}, s'_{h,i} \right) \right\}_{i=1}^{N}$. confidence level δ.
Set $\underline{Q}_{H+1}(s,a) = 0$, $\underline{V}_{H+1}(s) = 0$ for all (s,a)
for $h = H, \ldots, 1$ **do**
  for $s \in \mathcal{S}$, $a \in \mathcal{A}$ **do**
    $(\hat{r}_h(s,a), b_h^1(s,a)) \leftarrow f(\mathcal{D}_{r|hsa}, \frac{\delta}{2SAH})$
    $\left(\widehat{\mathrm{PV}}_{h,s,a}, b_h^2(s,a)\right) \leftarrow g(\mathcal{D}_{s'|hsa}, \underline{V}_{h+1}, \frac{\delta}{2SAH})$
    $\underline{Q}_h(s,a) = \max(0, \hat{r}_h(s,a) - b_h^1(s,a) + \widehat{\mathrm{PV}}_{h,s,a} - b_h^2(s,a))$
  end for
  for $s \in \mathcal{S}$ **do**
    $\underline{V}_h(s) = \max_{a \in \mathcal{A}} \underline{Q}_h(s,a)$
    $\hat{\pi}_h(s) = \operatorname{argmax}_{a \in \mathcal{A}} \underline{Q}_h(s,a)$
  end for
end for
Return: $\hat{\pi}$.

When the confidence bounds are small enough, the estimation for value function in Algorithm 8 will be accurate and $\hat{\pi}$ will choose the optimal action in each state with positive occupancy measure. With this intuition, we get a sufficient condition for optimality:

Theorem 6.4.2 (Optimality condition)**.** *Under the conditions in Theorem 6.4.1, if* $2Hb < \Delta_{\min}^{\mathcal{A}}$, *then* $\mathrm{SubOpt}(\hat{\pi}) = 0$ *with probability at least* $1 - \delta$.

Theorem 6.4.2 provides a general condition for optimal policy identification, which results in different guarantees given different estimators and corresponding confidence bounds. One can also derive an optimality condition using policy gap $\Delta_{\min}^{\Pi}$: because the set of deterministic optimal policies Π^* is discrete, when (6.1) is less than $\Delta_{\min}^{\Pi}$, $\mathrm{SubOpt}(\hat{\pi}) = 0$. However, this argument usually results in an overly conservative optimality condition. We defer the detailed discussion to Section 6.6.

In the case of learning with i.i.d. offline dataset, Theorem 4.1 of Wang et al. (2022) provides a dedicated sample complexity guarantee for offline optimal policy identification when the rewards are deterministic and known. Under a similar i.i.d. learning setting but with subGaussian rewards, we show, in Section 6.5, that

when the reward and transition estimators f and g are specified to be empirical mean estimators with Hoeffding-style confidence bound, Theorem 6.4.2 provides a similar sample complexity bound. However, our main focus is to use Algorithm 8 to study the robust offline learning setting in Section 6.5, which is much more challenging.

6.5 Case Studies

The meta-algorithm Algorithm 8 and its theoretical guarantee in Section 6.4 can be applied to various data generative models and reward distributions given estimators with proper confidence bounds. In this section, we present two case studies. We start with a standard learning setting in Section 6.5 as a warm-up where the dataset consists of i.i.d. samples and the reward distributions are subGaussian; we then present our main result in Section 6.5 with a harder learning setting where the dataset can be corrupted and reward distributions are heavy-tailed. In both case studies, we provide sufficient conditions for optimality derived using Theorem 6.4.2.

Warm-up: i.i.d. dataset with subGaussian rewards

We first consider the standard offline learning setting with an i.i.d. dataset and a sub-Gaussian rewards distribution. The exact policy recovery condition is known Wang et al. (2022), but our purpose here is to illustrate how one can instantiate Theorem 6.4.2 with f, g, in anticipation of our main result in the next section. We assume the reward distributions are subGausian:

Assumption 6.5.1 (SubGaussian rewards)**.** For all $(s, a, h) \in \mathcal{S}\times\mathcal{A}\times[H]$, $\mathcal{R}_h(s, a)$ is subGaussian with mean $r_h(s, a) := \mathbb{E}_{X\sim\mathcal{R}_h(s,a)}[X] \in [0, 1]$ and parameter σ^2, $\sigma > 0$, i.e. $\mathbb{E}_{X\sim\mathcal{R}_h(s,a)}[\exp(s(X - r_h(s, a)))] \leq \exp(\sigma^2 s^2/2)$, for all $s \in \mathbb{R}$.

In our offline learning setting, we consider the data generative model similar to Wang et al. (2020a), where the learning agent has access to an offline dataset drawn from some data distribution but cannot have further interaction with the

MDP. The i.i.d. dataset is generated as a set of transition tuples instead of trajectories. Specifically,

Definition 6.5.1 (Offline dataset)**.** An *offline dataset* $\mathcal{D}$ of size N collected with data distributions $\mu = \{\mu_h\}_{h\in[H]}$ is a multiset consisting of N transition tuples sampled at each time step:

$$\mathcal{D} = \bigcup_{h=1}^{H} \left\{ \left(s_{h,i}, a_{h,i}, r_{h,i}, s'_{h,i} \right) \right\}_{i=1}^{N}$$

where $(s_{h,i}, a_{h,i}) \sim \mu_h$, $r_{h,i} \sim \mathcal{R}_h(s_{h,i}, a_{h,i})$ and $s'_{h,i} \sim P_h(\cdot \mid s_h^i, a_h^i)$.

We assume the data distribution μ has *uniform coverage on all optimal policies*:

Assumption 6.5.2 (Uniform optimal policy coverage)**.** There exists $P > 0$, s.t. $\mu_h(s,a) \geq P$, for all $(s,a,h) \in \left\{(s,a,h) : \exists \pi^* \in \Pi^*, \text{ s.t. } d_h^{\pi^*}(s,a) > 0\right\}$.

As shown in Section D of Wang et al. (2022), this assumption is necessary for optimal policy recovery.

Under this standard offline learning setting, it is sufficient to use empirical mean estimator in both the reward estimator and transition estimator:

$$\hat{r}^{\text{emp}}_{h,s,a} = \frac{1}{N_h(s,a)} \sum_{r \in \mathcal{D}_{r|hsa}} r \tag{6.2}$$

$$\widehat{\mathrm{PV}}^{\text{emp}}_{h,s,a} = \frac{1}{N_h(s,a)} \sum_{s' \in \mathcal{D}_{s'|hsa}} \underline{V}_{h+1}(s'), \tag{6.3}$$

where $N_h(s,a) = \left|\mathcal{D}_{r|hsa}\right| = \left|\mathcal{D}_{s'|hsa}\right|$. We use the convention that $0/0 = 0$. The confidence bounds are given by the following lemma:

Proposition 6.5.1 (Confidence bound)**.** If Assumption 6.5.1 holds, then for all $(s,a,h) \in \mathcal{S}\times\mathcal{A}\times[H]$, with probability at least $1 - \frac{\delta}{2SAH}$:

$$\begin{aligned} \left|\hat{r}^{\text{emp}}_{h,s,a} - r_h(s,a)\right| &\leq b^{1,\text{emp}}_{h,s,a}, \\ \left|\widehat{\mathrm{PV}}^{\text{emp}}_{h,s,a} - P^{\top}_{h,s,a}\underline{V}_{h+1}\right| &\leq b^{2,\text{emp}}_{h,s,a}. \end{aligned}$$

where

$$b^{1,\text{emp}}_{h,s,a} = \sigma\sqrt{\frac{2\log\frac{8SAH}{\delta}}{N_h(s,a)}}, b^{2,\text{emp}}_{h,s,a} = H\sqrt{\frac{\log\frac{8SAH}{\delta}}{2N_h(s,a)}}$$

In this case study, the reward and transition estimators are defined to be:

$$f_{\text{emp}}\left(\mathcal{D}_{r|hsa}, \frac{\delta}{2SAH}\right) := \left(\hat{r}^{\text{emp}}_{h,s,a}, b^{1,\text{emp}}_{h,s,a}\right)$$
$$g_{\text{emp}}\left(\mathcal{D}_{s'|hsa}, \underline{V}_{h+1}, \frac{\delta}{2SAH}\right) := \left(\widehat{\text{PV}}^{\text{emp}}_{h,s,a}, b^{2,\text{emp}}_{h,s,a}\right)$$

Given the reward estimator and transition estimator, we can get the following optimality condition by applying Theorem 6.4.2:

Proposition 6.5.2 (Optimality condition)**.** Suppose Assumption 6.5.1, 6.5.2 hold. We specify the reward and transition estimators in Algorithm 8 to be f_{emp} and g_{emp}. Let $\hat{\pi}$ be the policy returned by Algorithm 8 given an offline dataset $\mathcal{D}$ generated according to Definition 6.5.1. If $4H(2\sigma + H)\frac{\log\frac{8SAH}{\delta}}{\sqrt{NP}} < \Delta^{\mathcal{A}}_{\min}$, then $\text{SubOpt}(\hat{\pi}) = 0$ with probability at least $1-\delta$.

Proposition 6.5.2 translates Theorem 6.4.2 to a sample complexity bound by using empirical mean estimation with Hoeffding-style confidence bound. This result is similar to Theorem 4.1 of Wang et al. (2022) but with a slightly worse dependence on H. We are now ready to present our main results in the robust offline learning setting.

Main results: corrupted dataset and heavy-tailed reward distributions

When (i) the reward distributions have weaker concentrations, and (ii) the dataset is corrupted, the learning problem becomes more challenging. Nonetheless, Algorithm 8 can be adapted to this setting by using powerful robust estimators. We first provide a novel analysis that allows an existing robust estimator to handle

unbounded variance and *data corruption*, then instantiate the exact policy recovery condition under this estimator.

Formally, we first relax the SubGaussian reward assumption in Assumption 6.5.1 by only assuming the reward distributions to have bounded $(1+\gamma)$-th centered moment:

Assumption 6.5.3 (Heavy-tailed reward distributions)**.** There exsits $\gamma \in (0, 1]$ and $\sigma > 0$, s.t. for all $(s, a, h) \in \mathcal{S}\times\mathcal{A}\times[H]$, $\mathbb{E}_{X\sim\mathcal{R}_h(s,a)}\left[(X - r_h(s, a))^{1+\gamma}\right] \leq \sigma^{1+\gamma}$, where $r_h(s, a) = \mathbb{E}_{X\sim\mathcal{R}_h(s,a)}[X] \in [0, 1]$.

Bubeck et al. (2013) first studies this reward distribution in multi-armed bandits. The reward distributions may not have finite variance, making the reward estimation itself a hard problem, even given clean data without data corruption. Bubeck et al. (2013) shows that empirical mean estimator results in a significantly wider confidence interval, which is not satisfactory. In this section, we study offline RL with a corrupted dataset, on top of this heavy-tailed reward model. Specifically, we consider an ϵ-corruption model on the offline dataset where **both rewards and transitions** can be corrupted, which is much more challenging than the learning problem in Definition 6.5.1:

Definition 6.5.2 (ϵ-corruption model)**.** Let $\epsilon \geq 0$. An *ϵ-corrupted offline dataset* $\mathcal{D}$ is a multiset generated by the following procedure: a clean offline dataset $\tilde{\mathcal{D}} = \bigcup_{h=1}^{H}\left\{\left(s_{h,i}, a_{h,i}, \tilde{r}_{h,i}, \tilde{s}'_{h,i}\right)\right\}_{i=1}^{N}$ is generated according to Definition 6.5.1; an adversary is allowed to inspect the whole dataset $\tilde{\mathcal{D}}$ and replace up to ϵ fraction of the reward entries and transition entries with something arbitrary for each (s, a, h) tuple. We denote the corrupted dataset as $\mathcal{D} = \bigcup_{h=1}^{H}\left\{\left(s_{h,i}, a_{h,i}, r_{h,i}, s'_{h,i}\right)\right\}_{i=1}^{N}$. In other words, we require $\frac{\sum_{i=1}^{N}\mathbb{I}\left\{\left(s_{h,i},a_{h,i}\right)=(s,a), r_{h,i}\neq\tilde{r}_{h,i}\right\}}{N_h(s,a)} \leq \epsilon$ and $\frac{\sum_{i=1}^{N}\mathbb{I}\left\{\left(s_{h,i},a_{h,i}\right)=(s,a), s'_{h,i}\neq\tilde{s}'_{h,i}\right\}}{N_h(s,a)} \leq \epsilon$ for all (s, a, h).

In the robust learning setting defined in Definition 6.5.2, the corrupted rewards can be unbounded. And importantly, the learning agent has no access to the clean dataset $\tilde{\mathcal{D}}$ and can only learn from the corrupted dataset $\mathcal{D}$.

Similar to Section 6.5, our first step is to design RewardEstimator f and TransitionEstimator g with proper confidence bound for Algorithm 8. We first formally define the robust mean estimation problem, which captures the hardness of the reward estimation problem:

Definition 6.5.3 (Robust mean estimation with heavy-tailed distribution)**.** Let $\gamma \in (0, 1]$, $\sigma \geq 0$, $\epsilon \in (0, 1)$. Let $\mathcal{P}$ be a heavy-tailed distribution in $\mathbb{R}$ with bounded $(1+\gamma)$-th centered moment: $\mathbb{E}_{X\sim\mathcal{P}}\left[|X - \mu|^{1+\gamma}\right] \leq \sigma^{1+\gamma}$, where $\mu := \mathbb{E}_{X\sim\mathcal{P}}[X]$. Given an i.i.d. dataset $\tilde{X}_1, \ldots, \tilde{X}_N$ drawn from $\mathcal{P}$, an adversary can inspect the dataset and replace an ϵ-faction of the data points with arbitrary values. The corrupted dataset $X_1, \ldots, X_N$ is revealed to the learning algorithm, which attempts to estimate μ, the mean of $\mathcal{P}$.

Trimmed Mean estimation is a well-studied estimator in robust statistics Lugosi and Mendelson (2021, 2019a). However, most prior work are limited to distributions with *subGaussian* distribution or at most distribution with *bounded variance*. Surprisingly, we show that the Trimmed Mean estimator in Lugosi and Mendelson (2021) can be directly applied to robust mean estimation in Definition 6.5.3 and resolves both difficulties simultaneously. For completeness, we present the Trimmed Mean estimator: Trimmed-Mean in Algorithm 17 in Appendix D.1.

Theorem 6.5.4 (Trimmed-Mean for heavy-tailed distribution)**.** *Suppose $\gamma \in (0, 1]$, $\epsilon < \frac{1}{32}$, $\delta \in (0, 1)$ and $N > 96 \log \frac{4}{\delta}$. Given N samples generated by the ϵ-corruption model in Definition 6.5.3, Algorithm 17 outputs a $\hat{\mu}$, s.t. with probability at least $1 - \delta$, $|\hat{\mu} - \mu| \leq C_{1,\gamma}\sigma\epsilon^{\frac{\gamma}{1+\gamma}} + C_{2,\gamma}\sigma\left(\frac{1}{N}\log\frac{8}{\delta}\right)^{\frac{\gamma}{1+\gamma}}$, where $C_{1,\gamma} = 128A_\gamma$, $C_{2,\gamma} = 768A_\gamma$ and A_γ is the smallest value s.t. $A_\gamma((1+x)\log(1+x) - x) \geq x^{\frac{\gamma+1}{\gamma}}/\left(1 + x^{\frac{1}{\gamma}}\right)$ for all $x > 0$.*

The error bound in Theorem 6.5.4 involves a bias term $O\left(\sigma\epsilon^{\frac{\gamma}{1+\gamma}}\right)$ and a statistical error term $\tilde{O}\left(\sigma N^{-\frac{\gamma}{1+\gamma}}\right)$. The bias is caused by data corruption why the statistical error term is due to finite sample. Importantly, both bias and statistical error term *meets the information-theoretic lower bound* (up to constants). Our new analysis is based on a variant of *Bernstein inequality under weak moment assumption*. We defer the details and more discussion about Theorem 6.5.4 to the end of this section.

We use the TRIMMED-MEAN estimator in Algorithm 17 and its confidence bound for reward estimation to handle the corrupted reward. The estimated reward is set to be: for all $(s,a,h) \in \mathcal{S}\times\mathcal{A}\times[H]$,

$$\hat{r}^{\mathrm{TM}}_{h,s,a} = \textsc{Trimmed-Mean}\left(\mathcal{D}_{r|hsa}, \epsilon, \frac{\delta}{4SAH}\right), \tag{6.4}$$

recall that $\mathcal{D}_{r|hsa}$ is the set of all rewards received in (s,a) visitations at step h. We use the same empirical mean estimator in (6.3) but with modified confidence bound to account for the effect of data corruption on the state transition. Formally, we have:

Proposition 6.5.3 (Confidence bound)**.** If Assumption 6.5.3 holds, then for all $(s,a,h) \in \mathcal{S}\times\mathcal{A}\times[H]$, with probability at least $1-\frac{\delta}{2SAH}$:

$$\begin{aligned}\left|\hat{r}^{\mathrm{TM}}_{h,s,a} - r_h(s,a)\right| \leq & b^{1,\mathrm{TM}}_{h,s,a} \\ \left|\widehat{\mathrm{PV}}^{\mathrm{emp}}_{h,s,a} - P^{\top}_{h,s,a}\underline{V}_{h+1}\right| \leq & b^{2,\mathrm{robust}}_{h,s,a}.\end{aligned}$$

where $\widehat{\mathrm{PV}}^{\mathrm{emp}}_{h,s,a}$ is defined in (6.3) and

$$b^{1,\mathrm{TM}}_{h,s,a} = \begin{cases} \infty & \text{if } N_h(s,a) \leq 96\log\frac{8SAH}{\delta} \\ C_{1,\gamma}\sigma\epsilon^{\frac{\gamma}{1+\gamma}} + C_{2,\gamma}\sigma\left(\frac{\log\frac{32SAH}{\delta}}{N_h(s,a)}\right)^{\frac{\gamma}{1+\gamma}} & \text{o.w.} \end{cases} \tag{6.5}$$

$$b^{2,\mathrm{robust}}_{h,s,a} = \epsilon H + H\sqrt{\frac{\log\frac{8SAH}{\delta}}{2N_h(s,a)}}, \tag{6.6}$$

where $C_{1,\gamma}$ and $C_{2,\gamma}$ are specified in Theorem 6.5.4.

$b^{1,\mathrm{TM}}_{h,s,a}$ is the confidence bound for the TRIMMED-MEAN estimator when applied to reward estimation. The success of the TRIMMED-MEAN estimation requires a minimum number of samples. So we simply set $b^{1,\mathrm{TM}}_{h,s,a}$ to ∞ when $N_h(s,a)$ is less than the threshold. Setting $b^{1,\mathrm{TM}}_{h,s,a}$ to ∞ looks excessive at the first glance. However, by Theorem 6.4.1, we can see that the suboptimality of $\hat{\pi}$ only depends on the bonus for (s,a,h) tuples covered by some optimal policy. By Assumption 6.5.2, the sample

size requirement of TRIMMED-MEAN is met with high probability for any (s, a, h) tuples covered by some optimal policy when N, the number of samples, is large enough.

In this case study, the reward and transition estimators are defined to be:

$$f_{\text{robust}}\left(\mathcal{D}_{r|hsa}, \frac{\delta}{2SAH}\right) := \left(\hat{r}^{\text{TM}}_{h,s,a}, b^{1,\text{TM}}_{h,s,a}\right)$$
$$g_{\text{robust}}\left(\mathcal{D}_{s'|hsa}, \underline{V}_{h+1}, \frac{\delta}{2SAH}\right) := \left(\widehat{\text{PV}}^{\text{emp}}_{h,s,a}, b^{2,\text{robust}}_{h,s,a}\right)$$

By applying Theorem 6.4.2, we get the following optimality condition:

Theorem 6.5.5 (Optimality condition)**.** *Suppose Assumption 6.5.3, 6.5.2 holds and* $\epsilon < \frac{1}{32}$, $N > \frac{768}{P}\left(\log \frac{8SA}{\delta}\right)^2$. *We specify the reward and transition estimators in Algorithm 8 to be* f_{robust} *and* g_{robust}. *Let* $\hat{\pi}$ *be the policy returned by Algorithm 8 given an offline dataset* $\mathcal{D}$, *where* $\mathcal{D}$ *is generated according to Definition 6.5.2. If* $2H\left(C_{1,\gamma}\sigma\epsilon^{\frac{\gamma}{1+\gamma}} + \epsilon H\right) + 4H\left(\frac{\sqrt{2}C_{2,\gamma}\sigma}{(NP)^{\frac{\gamma}{1+\gamma}}} + \frac{H}{\sqrt{NP}}\right)\log\frac{32SAH}{\delta} < \Delta^{\mathcal{A}}_{\min}$, *then* $\text{SubOpt}(\hat{\pi}) = 0$ *with probability at least* $1 - \delta$.

There are two terms on the LHS of the optimality condition in Theorem 6.5.5: the first term $2H\left(C_{1,\gamma}\sigma\epsilon^{\frac{\gamma}{1+\gamma}} + \epsilon H\right)$ involves the corruption level ϵ, which characterizes the bias caused by data corruption; the second term $4H\left(\frac{\sqrt{2}C_{2,\gamma}\sigma}{(NP)^{\frac{\gamma}{1+\gamma}}} + \frac{H}{\sqrt{NP}}\right)\log\frac{32SAH}{\delta}$ involves N, the size of the dataset, which characterizes the statistical error. If

$$2H\left(C_{1,\gamma}\sigma\epsilon^{\frac{\gamma}{1+\gamma}} + \epsilon H\right) < \Delta^{\mathcal{A}}_{\min} \tag{6.7}$$

then for N large enough, the optimality condition holds with high probability. This implies a key difference between robust RL and robust mean estimation: in robust mean estimation, it is never possible to learn the true mean even regardless of sample size due to the data corruption Lai et al. (2016); however, in robust RL, $\Delta^{\mathcal{A}}_{\min}$ creates a quantization effect, enabling the exact identification of a policy with the

optimal value despite minor corruption. This is reassuring because we can still aim to find a policy with the optimal value as long as (6.7) holds.

More discussion on Theorem 6.5.4 and the minimax optimality

Bubeck et al. (2013) provides a Median-of-Means estimator and a truncated empirical mean estimator for the mean estimation problem under heavy-tailed distribution, both are designed without the consideration of data corruption. The Median-of-Means estimator achieves the same rate as Theorem 6.5.4 for $\epsilon = 0$. Their truncated empirical mean estimator requires the uncentered moment $\mathbb{E}_{X\sim\mathcal{P}}\left[|X|^{1+\gamma}\right]$ to be bounded by some constant u, which increases as μ moves away from 0. However, this assumption leads to their error bound blowing up as u increases. In contrast, our algorithm handles data corruption and the error bound in Theorem 6.5.4 is translation invariant w.r.t. μ, which makes it significantly stronger.

Importantly, Theorem 17 is minimax optimal up-to some constant:

Theorem 6.5.6 (Error lower bound of the learning problem in Theorem 6.5.4)**.** *Given any learning algorithm $\mathscr{A}$, $\sigma > 0, \epsilon > 0$ and sufficiently large $N \in \mathbb{Z}_+$, there exists a distribution $\mathcal{P}$ with bounded $(1+\gamma)$-th centered moment and an adversary satisfying the constraints in Definition 6.5.3, s.t. any learning algorithm, given N data points from $\mathcal{P}$ with ϵ-fraction of corruption, will suffer an error at least $\Omega\left(\sigma\epsilon^{\frac{\gamma}{1+\gamma}} + \sigma N^{-\frac{\gamma}{1+\gamma}}\right)$ with at least constant probability.*

When $\epsilon = 0$, Theorem 6.5.6 implies the following error lower bound for mean estimation problem with i.i.d. data from a distribution with bounded $(1+\gamma)$-th centered moment:

Corollary 6.5.1. *Given any σ and sufficiently large N, there exists a distribution $\mathcal{D}$ with bounded $(1+\gamma)$-th centered moment, s.t. given N i.i.d. samples from the distribution, any learning algorithm will suffer an error at least $\Omega\left(\sigma N^{-\frac{\gamma}{1+\gamma}}\right)$ with at least constant probability.*

Lugosi and Mendelson (2021) guarantees an error $\tilde{O}(\sigma\sqrt{\epsilon} + \sigma/\sqrt{N})$ for the case when $\gamma = 1$, which is captured by Theorem 6.5.4. When $\gamma < 1$, our Theorem 6.5.4

provides a larger bias term of $O\left(\sigma\epsilon^{\frac{\gamma}{1+\gamma}}\right)$ and a slower convergence rate of $O(\sigma N^{-\frac{\gamma}{1+\gamma}})$. As shown in Theorem 6.5.6, these discrepancies are consequences of the inherent difficulty of the learning problem. The weaker moment assumption makes the estimation more challenging, leading to a larger error.

Proof sketch of Theorem 6.5.4

Algorithm 17 chooses $\tilde{\epsilon} = \tilde{O}(\epsilon + 1/N)$ as the trimming portion. It first splits the sample into two batches: D_1 and D_2. The trimming threshold α, β are set to be the $\tilde{\epsilon}$ and $(1-\tilde{\epsilon})$-quantile of D_1. The algorithm use α, β to define a clipping function $\phi_{\alpha,\beta}(\cdot)$, s.t. $\phi_{\alpha,\beta}(x) = \beta$ if $x > \beta$; $\phi_{\alpha,\beta}(x) = x$ if $\alpha \le x \le \beta$; $\phi_{\alpha,\beta}(x) = \alpha$ if $x < \alpha$. The algorithm simply returns the truncated mean of D_2: $\hat{\mu} = \frac{1}{|D_2|}\sum_{x\in D_2}\phi_{\alpha,\beta}(x)$.

In the proof of Theorem 6.5.4, we derive a novel Bernstein's inequality under weak moment assumption as a key lemma and conduct a refined analysis on the quantile of the heavy-tailed distribution. The remaining parts of the proof of Theorem 6.5.4 follow the main steps in Proof of Theorem 1 in Lugosi and Mendelson (2021). We first present the variant of Bernstein's inequality below:

Lemma 6.5.1 (Bernstein's inequality under weak moment assumption)**.** *Suppose X_j, $j = 1, \ldots, n$ is a sequence of independent zero-mean random variable bounded by $|X_j| \le M$ and there exists $\gamma \in (0, 1]$, s.t.*

$$\mathbb{E}\,|X_j|^{1+\gamma} \le \sigma^{1+\gamma}, \text{ for all } j = 1, \ldots, n.$$

then there exists $A_\gamma \ge 1$ (depending only on γ) s.t.:

$$\mathbb{P}\left(\frac{1}{n}\sum_{j=1}^{n} X_j > t\right) \le \exp\left\{-\frac{n}{A_\gamma}\frac{t^{\frac{\gamma+1}{\gamma}}}{\sigma^{\frac{1+\gamma}{\gamma}} + Mt^{\frac{1}{\gamma}}}\right\}.$$

Let $\tilde{D}_1 \cup \tilde{D}_2$ be the uncorrupted dataset. The estimation error of $\hat{\mu}$ can be

decomposed as:

$$\begin{aligned}|\hat{\mu}-\mu| \leq & \left|\frac{1}{|D_2|}\sum_{x\in D_2}\phi_{\alpha,\beta}(x)-\frac{1}{\left|\tilde{D}_2\right|}\sum_{x\in\tilde{D}_2}\phi_{\alpha,\beta}(x)\right| \\ & +\left|\frac{1}{\left|\tilde{D}_2\right|}\sum_{x\in\tilde{D}_2}\phi_{\alpha,\beta}(x)-\mathbb{E}_{X\sim\mathcal{P}}[\phi_{\alpha,\beta}(X)]\right| \\ & +\left|\mathbb{E}_{X\sim\mathcal{P}}[\phi_{\alpha,\beta}(X)]-\mu\right| \\ =: & B_1+B_2+B_3\end{aligned}$$

Because $\tilde{D}_2$ and D_2 differ by at most $2\epsilon\left|D_2\right|$ entries,

$$B_1 \leq 2\epsilon\max_{x,y\in\mathbb{R}}\left|\phi_{\alpha,\beta}(x)-\phi_{\alpha,\beta}(y)\right| = 2\epsilon(\beta-\alpha).$$

Because $\left\{\phi_{\alpha,\beta}(x) : x\in\tilde{D}_2\right\}$ consists i.i.d samples from a distribution with bounded $(1+\gamma)$-th centered moment and support bounded between $[\alpha,\beta]$, by Lemma D.7.1:

$$B_2 \leq \tilde{O}\left(\frac{\sigma}{N^{\frac{\gamma}{1+\gamma}}}+\frac{|\beta-\alpha|}{N}\right)$$

By concentration of Bernoulli random variables, α and β are close to the $\tilde{\epsilon}$ and $(1-\tilde{\epsilon})$-quantile of distribution $\mathcal{P}$. Furthermore, we can show that the truncated random variable $\phi_{\alpha,\beta}(X)$, where $X\sim\mathcal{P}$, has a mean close to the original random variable:

$$B_3 \leq O\left(\sigma\tilde{\epsilon}^{\frac{\gamma}{1+\gamma}}\right).$$

We finish the proof by combining these together.

6.6 Comparison between Different Optimality Conditions

In Section 6.4. we derive an optimality condition based on the suboptimal gap of actions in Theorem 6.4.2. Alternatively, we can get another optimality condition with the following observation: $\hat{\pi}$ is optimal if the suboptimality gap $\mathrm{SubOpt}(\hat{\pi})$ is less than the policy gap $\Delta_{\min}^{\Pi}$. Formally, we can get the following sufficient condition for optimality with Theorem 6.4.1:

Proposition 6.6.1 (Optimality condition)**.** Under the conditions in Theorem 6.4.1, if $2Hb < \Delta_{\min}^{\Pi}$, then $\mathrm{SubOpt}(\hat{\pi}) = 0$ with probability at least $1 - \delta$.

By Proposition 6.3.1, the action gap $\Delta_{\min}^{\mathcal{A}} \leq \Delta_{\min}^{\Pi}$ and the difference can be large. This means the condition in Proposition 6.6.1 is usually more conservative and thus less preferable than that in Theorem 6.4.2. In the following, we use contextual bandit as an illustrative example to show that why utilizing the action gap idea leads to a better sufficient condition.

When $H = 1$, MDP is specialized to contextual bandit. And Algorithm 8 returns a policy $\hat{\pi}$ that chooses the action with the largest lower confidence bound (LCB) in each state. Similar to the discussion above, we can make sure $\hat{\pi}$ is optimal by comparing either the action gap or policy gap. We will show that utilizing the action gap is preferable.

In contextual bandit, the action gap can be written as:

$$\Delta_{\min}^{\mathcal{A}} = \min_{(s,a)\in\mathcal{C}} (r_1(s, \pi^*(s)) - r_1(s, a)),$$

where π^* is an optimal policy and

$$\mathcal{C} := \{(s, a) : s \in \mathbf{supp}\,(p_0), r_1(s, a) \neq r_1(s, \pi^*(s))\}.$$

Because the best *suboptimal* policy should only choose a suboptimal action in one

state, we can write the policy gap as:

$$\Delta^{\Pi}_{\min} = \min_{(s,a)\in\mathcal{C}} p_0(s)(r_1(s,\pi^*(s)) - r_1(s,a)).$$

Because $p_0(s)$ can be very small for some state s, the policy gap $\Delta^{\Pi}_{\min}$ can be much smaller than the action gap $\Delta^{\mathcal{A}}_{\min}$.

Since there is no state transition in contextual bandits, $b^2_1(\cdot,\cdot) = 0$ and the value function estimation in Algorithm 8 can be written as:

$$\underline{Q}_1(s,a) = \max\{0, \hat{r}_1(s,a) - b^1_1(s,a)\} \quad \forall s,a \tag{6.8}$$

By the definition of $\hat{\pi}$ and the fact that $b_1(\cdot,\cdot)$ is a proper confidence bound, with probability at least $1-\delta/4$, the suboptimality of $\hat{\pi}$ at any s can be bounded by:

$$\begin{aligned}
&V^*_1(s) - Q_1(s,\hat{\pi}(s)) = r_1(s,\pi^*(s)) - r_1(s,\hat{\pi}(s)) \\
\leq& r_1(s,\pi^*(s)) - \underline{Q}_1(s,\hat{\pi}(s)) \leq r_1(s,\pi^*(s)) - \underline{Q}_1(s,\pi^*(s)) \\
=& r_1(s,\pi^*(s)) - \max\{0, \hat{r}_1(s,\pi^*(s)) - b^1_1(s,\pi^*(s))\} \\
\leq& 2b^1_1(s,\pi^*(s)),
\end{aligned} \tag{6.9}$$

where π^* is an optimal policy. Thus under the conditions in Theorem 6.4.1, the suboptimality gap of $\hat{\pi}$ can be bounded by:

$$\begin{aligned}
\mathrm{SubOpt}(\hat{\pi}) =& \mathbb{E}_{s\sim p_0}[V^*_1(s) - Q_1(s,\hat{\pi}(s))] \\
\leq& \mathbb{E}_{s\sim p_0}\left[2b^1_1(s,\pi^*(s))\right] \leq 2b.
\end{aligned} \tag{6.10}$$

We can ensure the optimality of $\hat{\pi}$ by using either the action gap or policy gap:

- on one hand, by (6.9), if $2b_1(s,\pi^*(s)) \leq 2b < \Delta^{\mathcal{A}}_{\min}$ for all $s \in \mathcal{S}$, then for all $s \in \mathcal{S}$, $\hat{\pi}$ chooses an optimal action and thus achieves the optimal value;
- on the other hand, by (6.10), if $2b < \Delta^{\Pi}_{\min}$, then $\hat{\pi}$ achieve the optimal value.

However, the condition $2b < \Delta^{\Pi}_{\min}$ is more conservative than $2b < \Delta^{\mathcal{A}}_{\min}$ because

$\Delta^{\Pi}_{\min}$ can be much smaller than $\Delta^{\mathcal{A}}_{\min}$. Similarly, in the more general MDP setting, $\Delta^{\mathcal{A}}_{\min}$ and $\Delta^{\Pi}_{\min}$ differ by at least a factor of state occupancy probability as shown in Proposition 6.3.1, thus Theorem 6.4.2 provides a more desirable optimality condition than Proposition 6.6.1.

6.7 Conclusion

We provided a new optimality condition for corruption-robust offline RL with heavy-tailed rewards. We show that if $\tilde{O}\left(H\sigma\epsilon^{\frac{\gamma}{1+\gamma}} + \epsilon H^2\right) < \Delta^{\mathcal{A}}_{\min}$, then a modified pessimistic value iteration algorithm can obtain a policy with the optimal value even under data corruption.

Future work should answer the question: what is the **sufficient and necessary** condition for learners to get a policy with optimal value? A less fundamental but equally interesting direction is to strengthen the sample complexity in this paper.

In this chapter, we extend the instance-dependent analysis to multi-agent setting. Two-player zero-sum games are fundamentally different from tabular MDPs: tabular MDPs always have deterministic optimal policy while some two-player zero-sum games may have only mixed strategy Nash equilibria, which makes it impossible to learn an exact Nash in some cases. As a starting point, we study the perturbation stability of the two-player zero-sum games and the recovery of Nash equilibrium (NE) when learning on a perturbed game matrix. We provide the sufficient and necessary conditions for the learner to get an exact NE by learning on the perturbed game. When it's impossible to learn the exact NE, we provide conditions for NE support recovery as a compromise. In robust offline learning setting, the estimation for the game matrix is a perturbed game of the expected pay-off matrix. Using our results on the perturbation-stability, we further establish gap-dependent analysis and certifiable guarantees.

7.1 Introduction

Nash equilibrium is a fundamental concept in game theory that explores the behavior of rational players Nash Jr (1996). It finds numerous applications in economics Kreps (1990), network security Roy et al. (2010), supply chain management Leng and Parlar (2005), and more. However, a drawback of the current framework is that game models are often approximations of the real game, introducing estimation errors. When planning based on the estimated environment, it becomes unclear how far the chosen strategy is from the actual Nash equilibrium. To address this issue, perturbation stability emerges as a valuable property for Nash equilibrium learning.

Perturbation-stability characterizes the property that, if the entries of the game matrix are only changed slightly, the NE of the perturbed game is not far away from the original game. Perturbation-stable games, as described in Balcan and Braver-

man (2017), can model practical situations such as public good games, matching pennies, and identical interest games. Perturbations exist because game matrices are abstractions of reality. Without perturbation stability, learning an NE far from the real NE is possible, which hampers understanding of the game structure Balcan and Braverman (2017); Lipton and Mehta (2006). Perturbation-stable games offer interesting properties, enabling guarantees beyond worst-case analysis in terms of computational and statistical aspects. They facilitate more efficient algorithm design for equilibrium computation Balcan and Braverman (2017).

Perturbation stability also provides insights into the learning of optimal policy in Multi-armed Bandits (MABs) and more generally, tabular Markov Decision Processes (MDPs). In MABs and tabular MDPs, there exist optimal deterministic policies Lattimore and Szepesvári (2020); Sutton and Barto (2018). So it's sufficient to consider the set of deterministic policies, which is a discrete set. As a result, MABs and tabular MDPs are perturbation-stable and planning on a slightly different environment could still lead to an optimal policy. For example, in the offline learning setting, the suboptimality gap of the policy $\hat{\pi}$ returned by the PEVI algorithm can be bounded by $\tilde{O}(N^{-\frac{1}{2}})$, where N is the size of the dataset Jin et al. (2021). When the upper bound $\tilde{O}(N^{-\frac{1}{2}})$ is smaller than the suboptimality of the best suboptimal policy, it's guaranteed that $\hat{\pi}$ is optimal due to discreteness. Similar gap-dependent analysis has been used in MABs, offline reinforcement learning (RL) and online RL Lattimore and Szepesvári (2020); Wang et al. (2022); Simchowitz and Jamieson (2019). However, it's not clear how gap-dependent analysis can be applied to two-player zero-sum games because: on one hand, it's not clear what's the proper gap notion here; on the other hand, there are games with only mixed Nash equilibria Nash Jr (1996), which might breaks the perturbation stability as it's no longer sufficient to consider only the set of pure strategies when learning the NE.

Most recent work in multi-agent reinforcement learning (MARL) Xie et al. (2020); Cui and Du (2022) chooses the duality gap as a performance metric, guaranteeing value-related results. However, the resulting strategy may deviate significantly from the NE of the real game. Studies by Cohen (1986); Troutt (1990)

examine game sensitivity using the derivative of NE, but they offer limited insight into perturbation in the game space. Furthermore, they focus only on fully mixed NE games, which is restrictive.

In this paper, we study the perturbation stability of two-player zero-sum games and NE recovery in the offline learning setting. We demonstrate that under suitable conditions, slight perturbations yield some extent of NE recovery. Consequently, even with finite and potentially corrupted data, NE recovery might be feasible. Our contributions are as follows:

1. We propose three levels of NE robustness for two-player zero-sum games: **exact-NE-robust**, **subset-NE-robust**, and **NE-support-robust**. These notions of NE robustness correspond to different levels of **perturbation-stability**;

2. We introduce the **switch-out gap** as a key property for perturbation-stability. Built upon this, we derive the **sufficient and necessary conditions** for both exact-NE-robust and subset-NE-robust. We also present a **sufficient condition** for NE-support-robust;

3. We apply our results to **corruption-robust offline learning** setting. By using Trimmed-Mean estimation, we reduce the robust offline learning problem to perturbation problem. We first present a set of **gap-dependent** results, which translate to sample complexity for NE (support) recovery. We then present **certifiable results**, which provide some computable criteria serving as certification for NE (support) recovery.

7.2 Related Works

Perturbation Stability of Games Lipton et al. Lipton and Mehta (2006) analyze stability in game theory and economics when subjected to perturbations in the input. Troutt et al. Troutt (1986) and Kimura et al. Kimura et al. (2000) explored perturbation effects in game theory, with a specific focus on perturbations with particular structures. Cohen (1986); Troutt (1990) investigated the sensitivity and

local perturbations in fully mixed games. They examined the derivative of matrix entries to gain a deeper understanding of the perturbation effects. It is important to note that their analysis primarily concentrated on games with fully mixed Nash equilibria, where the equilibria have full support on all rows and columns. Balcan and Braverman (2017); Awasthi et al. (2010) proposed more efficient algorithms for Nash equilibrium computation in perturbation-stable games.

RL and MARL Reinforcement Learning (RL) is a field that investigates the learning and planning processes in unknown environments Sutton and Barto (2018). Gap-independent results focus on the value space. Those results study the regret in the online setting Jin et al. (2020b) and suboptimality gap in the offline setting Jin et al. (2021). Gap-dependent analysis refine the gap-independent analysis with a notion of gap. It achieves logarithmic regret in the online setting Simchowitz and Jamieson (2019); Lattimore and Szepesvári (2020) and provide sample complexity bound for optimal policy identification in the offline setting Wang et al. (2022). However, the gap-dependent results and optimality policy recovery are mainly derived for single-agent RL. In the context of multi-agent reinforcement learning (MARL), the emphasis in recent studies has been on the duality gap as a performance measure, primarily focusing on the value space Xie et al. (2020); Cui and Du (2022). In this paper, we extend the gap-dependent analysis and optimal policy identification to two-player zero-sum matrix game, which is a special case of MARL.

7.3 Preliminary

We consider the two-player zero-sum matrix game. Let $A \in \mathbb{R}^{m \times n}$ be the *payoff matrix*, $[m]$ be the *action space* of the row player, and $[n]$ be the action space of the column player. The *payoff* of the game when the row player chooses action i and the column player chooses action j is given by $\mathbf{e}_i^\top A \mathbf{e}_j$, where $\mathbf{e}_i$ and $\mathbf{e}_j$ are one-hot vectors of the appropriate dimensions.

The *strategy space* is a probability distribution over *pure strategies* (actions). We denote the strategy space of the row player as $\Delta([m])$, which represents the probability simplex over all rows of A, and the strategy space of the column player as $\Delta([n])$, representing the probability simplex over all columns. The payoff of a strategy pair $\pi := (\mathbf{p}, \mathbf{q}) \in \Delta([m]) \times \Delta([n])$ is given by $\mathbf{p}^\top A\mathbf{q}$. The row player aims to maximize the payoff, while the column player aims to minimize it. A strategy pair $\pi^* := (\mathbf{p}^*, \mathbf{q}^*)$ is a *Nash equilibrium* (NE) of game A if for all $i \in [m]$, $\mathbf{e}_i^\top A\mathbf{q}^* \leq \mathbf{p}^{*\top} A\mathbf{q}^*$ and for all $j \in [n]$, $\mathbf{p}^{*\top} A\mathbf{e}_j \geq \mathbf{p}^{*\top} A\mathbf{q}^*$. At an NE, no player can achieve a better payoff by unilaterally choosing another strategy. The *value* of A is denoted as $v^* := \mathbf{p}^{*\top} A\mathbf{q}^*$. It is known that the NE of a two-player zero-sum game can be found using linear programming Nisan et al. (2007). We use $\mathrm{NE}(A)$ to denote the set of NE of game A. We use $\mathcal{P}$ and $\mathcal{Q}$ to denote the sets of optimal strategies of row and column players, i.e.:

$$\begin{aligned}\mathcal{P} :=& \{\mathbf{p} \in \Delta([m]) : \exists \mathbf{q} \in \Delta([n]),\ \text{s.t. } (\mathbf{p}, \mathbf{q}) \in \mathrm{NE}(A)\}\\ \mathcal{Q} :=& \{\mathbf{q} \in \Delta([n]) : \exists \mathbf{p} \in \Delta([m]),\ \text{s.t. } (\mathbf{p}, \mathbf{q}) \in \mathrm{NE}(A)\}.\end{aligned}$$

We use $\mathcal{I}_A$ and $\mathcal{J}_A$ to denote the NE support of the row and column players:

$$\mathcal{I}_A := \bigcup_{\mathbf{p} \in \mathcal{P}} \mathbf{supp}\,(\mathbf{p}), \quad \mathcal{J}_A := \bigcup_{\mathbf{q} \in \mathcal{Q}} \mathbf{supp}\,(\mathbf{q}).$$

Given $\mathcal{I}_A$ and $\mathcal{J}_A$, we define the switch-out gap as the uniform lower bound of sub-optimality incurred by switching from NE to a pure strategy outside $\mathcal{I}_A$ or $\mathcal{J}_A$:

Definition 7.3.1 (Switch-out gap)**.** The *switch-out gap* is defined to be:

$$\begin{aligned}\Delta_{\mathcal{I}_A} :=& \min_{\mathbf{q} \in \mathcal{Q}} \min_{i \notin \mathcal{I}_A} \Big(v^* - \mathbf{e}_i^\top A\mathbf{q}\Big),\\ \Delta_{\mathcal{J}_A} :=& \min_{\mathbf{p} \in \mathcal{P}} \min_{j \notin \mathcal{J}_A} \Big(\mathbf{p}^\top A\mathbf{e}_j - v^*\Big),\end{aligned}$$

where v^* is the value of A. When $\mathcal{I}_A = [m]$ (or $\mathcal{J}_A = [n]$), $\Delta_{\mathcal{I}_A}$ (or $\Delta_{\mathcal{J}_A}$) is defined to be ∞.

An *approximate Nash equilibrium*, or α-approximate NE, is defined similarly, where the players have only a small incentive to deviate. Specifically, $\pi = (\mathbf{p}, \mathbf{q})$ is an α-approximate NE of A if for all $i \in [m]$, $\mathbf{e}_i^\top A\mathbf{q} \leq \mathbf{p}^\top A\mathbf{q} + \alpha$, and for all $j \in [n]$, $\mathbf{p}^\top A\mathbf{e}_j \geq \mathbf{p}^\top A\mathbf{q} - \alpha$.

In this paper, we investigate learning the Nash equilibrium with a perturbed game matrix denoted as $A + \Gamma$ where $\Gamma \in \mathbb{R}^{m\times n}$ is a perturbation matrix. Standard results Cui and Du (2022) typically focus on **duality gap** of a strategy pair as the performance metric in the **value space**[1]. The duality gap $\mathrm{Gap}(\cdot;\cdot)$ of a strategy pair $\pi = (\mathbf{p}, \mathbf{q})$ on game A is defined to be: $\mathrm{Gap}(\pi; A) := \mathrm{br}(\mathbf{q})^\top A\mathbf{q} - \mathbf{p}^\top A\mathrm{br}(\mathbf{p})$, where $\mathrm{br}(\cdot)$ denotes the *best response* of a strategy in game A. In other words, $\mathrm{br}(\mathbf{p}) \in \mathrm{argmin}_{\mathbf{q}'} \mathbf{p}^\top A\mathbf{q}'$, $\mathrm{br}(\mathbf{q}) \in \mathrm{argmax}_{\mathbf{p}'} \mathbf{p}'^\top A\mathbf{q}$.

However, the main focus of this paper is the recovery of Nash equilibrium in the **strategy space**. Facing the perturbation, we are interested in finding strategy pairs not only with small duality gap but also close to the NE of the original game. In this regard, we present the following criteria for Nash equilibrium recovery.

Recovery Criteria

The best result one can hope for is to exactly recover the set of NEs by planning on the perturbed game. With such property, the player can safely deploy the strategy in the real environment and get the full NE structure of the original game. We introduce the following *exact-NE-robust* criterion:

Definition 7.3.2 (Exact-NE-robust)**.** Game A is said to be **exact-NE-robust** within radius γ if $\forall \Gamma : \|\Gamma\|_{\mathcal{I}_A \cup \mathcal{J}_A} := \max_{(i,j)\in(\mathcal{I}_A\times[n])\cup([m]\times\mathcal{J}_A)} \left|\mathbf{e}_i^\top \Gamma \mathbf{e}_j\right| \leq \gamma$, we have $\mathrm{NE}(A + \Gamma) = \mathrm{NE}(A)$.

As shown later, it's sufficient to consider the perturbation on the $(\mathcal{I}_A\times[n]) \cup ([m]\times\mathcal{J}_A)$ for NE recovery. This is similar to the unilateral concentration assumption in Cui and Du (2022).

[1]For completeness, we include the discussion about the duality gap guarantee in the appendix.

When it is not practical to find all NE of the original game, one may hope to recover at least a subset of the NE as a safety guarantee:

Definition 7.3.3 (Subset-NE-robust)**.** Game A is said to be **subset-NE-robust** within radius γ if $\forall \Gamma : \|\Gamma\|_{\mathcal{I}_A \cup \mathcal{J}_A} \leq \gamma$, $\mathrm{NE}(A+\Gamma) \subseteq \mathrm{NE}(A)$.

In many cases, it may not be possible to recover any NE by learning on the perturbed game. As a compromise, we seek to find all of the actions chosen by the NE strategy, i.e. NE support recovery:

Definition 7.3.4 (NE-support-robust)**.** Game A is said to be **NE-support-robust** within radius γ if $\forall \Gamma : \|\Gamma\|_{\mathcal{I}_A \cup \mathcal{J}_A} \leq \gamma$, and for all Nash equilibrium π' of $A+\Gamma$, there exists π, a Nash equilibrium of A, s.t. $\mathbf{supp}\,(\pi') = \mathbf{supp}\,(\pi)$.

7.4 Main Results: Conditions for Nash Recovery

In this section, we present conditions for different levels of NE recoveries introduced in Section 7.3. For exact-NE-robust and subset-NE-robust, we provide the sufficient and necessary conditions, i.e. the minimum conditions allowing NE recovery. We provide a sufficient condition for NE-support-robust, which also guarantees small distance to the NE of the original game.

To facilitate the discussion, we introduce the following pure base NE assumption, which is crucial for subset-NE-robust:

Assumption 7.4.1 (Pure Base NE)**.** $A_{\mathcal{I}_A, \mathcal{J}_A}$ is a constant matrix.

In other words, Assumption 7.4.1 implies that the each NE of A is some convex combination of pure NEs. Under Assumption 7.4.1, the switch-out gap can be characterized by the switch-out gap of pure NEs:

Lemma 7.4.1. *If A satisfies Assumption 7.4.1, then*

$$\Delta_{\mathcal{I}_A} = \min_{j \in \mathcal{J}_A} \min_{i \notin \mathcal{I}_A} \Big(v^* - \mathbf{e}_i^\top A \mathbf{e}_j \Big),$$
$$\Delta_{\mathcal{J}_A} = \min_{i \in \mathcal{I}_A} \min_{j \notin \mathcal{J}_A} \Big(\mathbf{e}_i^\top A \mathbf{e}_j - v^* \Big).$$

The pure base NE assumption and a positive switch-out gap are indeed sufficient and necessary for subset-NE-robust:

Theorem 7.4.2. *Suppose $\gamma > 0$, A is subset-NE-robust within radius γ if and only if*

1. *A satisfies Assumption 7.4.1;*
2. $\gamma < \frac{1}{2}\min\{\Delta_{\mathcal{I}_A}, \Delta_{\mathcal{J}_A}\}$.

Implication of Subset-NE robust Subset-NE robust is useful in practice. For example, one may estimate the game by learning on a finite dataset. The estimation may differ from the original game up to some estimation error. However, Theorem 7.4.2 demonstrates that as long as the original game satisfies the pure base NE assumption and the estimation error is smaller than the switch-out gap, then the learner will learn an NE despite the estimation error. We defer the detailed discussion to Section 7.5.

The gap condition By Lemma 7.4.1, the gap condition $\gamma < \frac{1}{2}\min\{\Delta_{\mathcal{I}_A}, \Delta_{\mathcal{J}_A}\}$ guarantees that after the perturbation, both players have no incentive to choose actions outside the NE support. In this case, the switch-out gaps $\Delta_{\mathcal{I}_A}$ and $\Delta_{\mathcal{J}_A}$ share a similar functionality with $\Delta_{\min}$, the action gap in Multi-armed Bandits (MABs) Lattimore and Szepesvári (2020): when planning on a slightly different environment, one can still get the optimal strategy due to the fault tolerance.

We now present two examples showing that when the gap condition in Theorem 7.4.2 does not hold, game A is no longer subset-NE-robust:

Example 7.4.1 ($\frac{1}{2}\min\{\Delta_{\mathcal{I}_A}, \Delta_{\mathcal{J}_A}\} = 0$)**.** In game $\begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}$, $\pi_1 = (\mathbf{e}_1, \mathbf{e}_1)$, $\pi_2 = (\mathbf{e}_2, \mathbf{e}_1)$ and any convex combinations of π_1 and π_2 are NEs. We have $\mathcal{I}_A = \{1, 2\}$ and $\mathcal{J}_A = \{1\}$. In this game, the switch-out gaps are $\Delta_{\mathcal{I}_A} = \infty$ and $\Delta_{\mathcal{J}_A} = 0$. As a result, an arbitrarily small perturbation will create a new NE and thus violate the subset-NE-robust criterion. Consider the perturbed game matrix: $\begin{pmatrix} 0 & 0 \\ -\gamma & 1 \end{pmatrix}$, where

$\gamma > 0$ can be arbitrarily close to 0. One can verify that $\pi' = \left(\mathbf{e}_1, \left(\frac{1}{1+\gamma}, \frac{\gamma}{1+\gamma}\right)\right)$ is an NE of the perturbed game but not the original game.

Example 7.4.2 ($\frac{1}{2}\min\{\Delta_{\mathcal{I}_A}, \Delta_{\mathcal{J}_A}\} = \|\Gamma\|_{\mathcal{I}_A \cup \mathcal{J}_A}$)**.** In game $\begin{pmatrix} -1 & 1 \end{pmatrix}$, $\pi = (\mathbf{e}_1, \mathbf{e}_1)$ is the unique NE and we have $\mathcal{I}_A = \mathcal{J}_A = \{1\}$, $\Delta_{\mathcal{I}_A} = \infty$ and $\Delta_{\mathcal{J}_A} = 2$. After perturbation $\begin{pmatrix} 1 & -1 \end{pmatrix}$, the game matrix becomes $\begin{pmatrix} 0 & 0 \end{pmatrix}$. Any strategy pairs are NEs of the perturbed game, which violates the subset-NE robust criterion.

The Pure Base NE assumption Perturbation may result in slight changes on NEs. The pure base NE assumption provides tolerance for such changes. Specifically, it makes sure that as long as the NE support of the perturbed game is a subset of that of the original game, the NE of the perturbed game will be NE of the original game.

Example 7.4.3 (Game with only mixed NE)**.** Game $\begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}$ has a unique mixed NE $\pi = \left(\left(\frac{1}{2}, \frac{1}{2}\right), \left(\frac{1}{2}, \frac{1}{2}\right)\right)$. It has infinite switch-gap $\Delta_{\mathcal{I}_A} = \Delta_{\mathcal{J}_A} = \infty$ but does not satisfy Assumption 7.4.1. As a result, an arbitrarily small perturbation will create a new NE. Consider the perturbed game matrix: $\begin{pmatrix} 1+\gamma & -1 \\ -1-\gamma & 1 \end{pmatrix}$, where $\gamma > 0$ can be arbitrarily close to 0. The perturbed game has a unique NE $\left(\left(\frac{1}{2}, \frac{1}{2}\right), \left(\frac{1}{2+\gamma}, \frac{1+\gamma}{2+\gamma}\right)\right)$, which is different from π.

These examples informally demonstrate that the conditions in Theorem 7.4.2 are necessary. We defer the detailed proof of necessity to the appendix and present the proof of sufficiency below:

Proof of "$\Leftarrow$" in Theorem 7.4.2. We only need to show that under the condition in Theorem 7.4.2, any NE of $A + \Gamma$ has support only on $\mathcal{I}_A$ and $\mathcal{J}_A$. Then, by Assumption 7.4.1, any NE of $A + \Gamma$ is also NE of A.

We first show that the NE of the subgame defined by $(A + \Gamma)_{\mathcal{I}_A, \mathcal{J}_A}$ is also NE of $A + \Gamma$. Let $\left(\mathbf{p}^*_{\mathcal{I}_A}, \mathbf{q}^*_{\mathcal{J}_A}\right)$ be NE of the subgame $(A + \Gamma)_{\mathcal{I}_A, \mathcal{J}_A}$. By Assumption 7.4.1,

$\left(\mathbf{p}^*_{\mathcal{I}_A}, \mathbf{q}^*_{\mathcal{J}_A}\right)$ is NE of A. For all $i \notin \mathcal{I}_A$:

$$\begin{aligned}
\mathbf{e}_i^\top (A+\Gamma)\mathbf{q}^*_{\mathcal{J}_A} \leq & \mathbf{e}_i^\top A\mathbf{q}^*_{\mathcal{J}_A} + \gamma \mathbf{e}_i^\top \mathbf{1}\mathbf{1}^\top \mathbf{q}^*_{\mathcal{J}_A} \\
= & \mathbf{e}_i^\top A\mathbf{q}^*_{\mathcal{J}_A} + \gamma \\
\leq & \mathbf{p}^{*\top}_{\mathcal{I}_A} A\mathbf{q}^*_{\mathcal{J}_A} - \Delta_{\mathcal{I}_A} + \gamma \\
& (\text{ By definition of } \Delta_{\mathcal{I}_A}) \\
= & \mathbf{p}^{*\top}_{\mathcal{I}_A} (A+\Gamma)\mathbf{q}^*_{\mathcal{J}_A} - \Delta_{\mathcal{I}_A} + \gamma - \mathbf{p}^{*\top}_{\mathcal{I}_A} \Gamma \mathbf{q}^*_{\mathcal{J}_A} \\
\leq & \mathbf{p}^{*\top}_{\mathcal{I}_A} (A+\Gamma)\mathbf{q}^*_{\mathcal{J}_A} - \Delta_{\mathcal{I}_A} + 2\gamma \\
< & \mathbf{p}^{*\top}_{\mathcal{I}_A} (A+\Gamma)\mathbf{q}^*_{\mathcal{J}_A} \qquad (7.1) \\
& (\text{ Because } \Delta_{\mathcal{I}_A} > 2\gamma)
\end{aligned}$$

Because $\left(\mathbf{p}^*_{\mathcal{I}_A}, \mathbf{q}^*_{\mathcal{J}_A}\right)$ is the NE of $(A+\Gamma)_{\mathcal{I}_A,\mathcal{J}_A}$, for all $i \in \mathcal{I}_A$,

$$\mathbf{e}_i^\top (A+\Gamma)\mathbf{q}^*_{\mathcal{J}_A} \leq \mathbf{p}^{*\top}_{\mathcal{I}_A} (A+\Gamma)\mathbf{q}^*_{\mathcal{J}_A} \qquad (7.2)$$

By (7.1), (7.2) and similar argument on $\mathbf{p}^*_{\mathcal{I}_A}$, $\left(\mathbf{p}^*_{\mathcal{I}_A}, \mathbf{q}^*_{\mathcal{J}_A}\right)$ is NE of $A+\Gamma$.

We now prove that NE of $A+\Gamma$ has support only on $\mathcal{I}_A$ and $\mathcal{J}_A$ by contradiction. Suppose $(\mathbf{p}, \mathbf{q})$ is an NE of $A+\Gamma$, s.t. $\mathbf{supp}\,(\mathbf{p}) \setminus \mathcal{I}_A \neq \varnothing$, then $(\mathbf{p}, \mathbf{q}^*_{\mathcal{J}_A})$ is also NE of $A+\Gamma$. By (7.1), $\mathbf{p}^\top (A+\Gamma)\mathbf{q}^*_{\mathcal{J}_A} < \mathbf{p}^{*\top}_{\mathcal{I}_A} (A+\Gamma)\mathbf{q}^*_{\mathcal{J}_A}$, which contradicts with the fact that NEs of game have the same payoff. Thus $\mathbf{supp}\,(\mathbf{p}) \subseteq \mathcal{I}_A$. Similarly, we can show that $\mathbf{supp}\,(\mathbf{q}) \subseteq \mathcal{J}_A$. ■

Exact-NE-robust requires the NE sets of the perturbed game and original game to be exactly equal and is thus a special case of subset-NE-robust. As a result, it also requires a strictly stronger sufficient and necessary condition:

Theorem 7.4.3. *Suppose $\gamma > 0$, A is exact-NE-robust within radius γ if and only if A satisfies:*

1. *A satisfies: $|\mathcal{I}_A| = |\mathcal{J}_A| = 1$;*
2. *$\gamma < \frac{1}{2}\min\{\Delta_{\mathcal{I}_A}, \Delta_{\mathcal{J}_A}\}$.*

Compared to Theorem 7.4.2, Theorem 7.4.3 further requires game A and the perturbed game to have a unique pure NE. On one hand, $|\mathrm{NE}(A)| = |\mathrm{NE}(A+\Gamma)| = 1$ and $\mathrm{NE}(A+\Gamma) \subseteq \mathrm{NE}(A)$ implies $\mathrm{NE}(A+\Gamma) = \mathrm{NE}(A)$; on the other hand, the new condition is indeed necessary because if the game satisfies the pure base NE assumption but has multiple NEs, the perturbation can easily shrink the NE set and violate the criterion for exact-NE-robust:

Example 7.4.4. Game $\begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$ satisfies the pure base NE assumption and any strategy pairs are NE. After a small perturbation, the perturbed game $\begin{pmatrix} 0 & \gamma \\ -\gamma & 0 \end{pmatrix}$ has a unique NE $(\mathbf{e}_1, \mathbf{e}_1)$, where $\gamma > 0$ can be arbitrarily close to 0. This violates the criterion of exact-NE-robust.

By Theorem 7.4.2 and 7.4.3, the recovery of (at least one) NEs requires the original game to satisfy the pure base NE assumption. For more general games, we have to relax the recovery criterion. We thus consider the NE-supp-robust in Definition 7.3.4 instead. We require the NE of the perturbed game to have the same support as that of the original game but allow the mixing probability to be slightly different.

We provide a sufficient condition for support-robust for the games with unique but potentially mixed NE:

Theorem 7.4.4. *If A has a unique NE $\pi^* = (\mathbf{p}^*, \mathbf{q}^*)$, then there exists $\gamma_A, C_A > 0$ s.t.:*

1. *A is NE-support-robust within radius γ for all $\gamma < \gamma_A$;*
2. *$\forall \|\Gamma\|_{\mathcal{I}_A \cup \mathcal{J}_A} < \gamma_A$, $A + \Gamma$ has a unique NE $\pi = (\mathbf{p}, \mathbf{q})$ and*

$$\|\pi^* - \pi\|_1 := \max\{\|\mathbf{p} - \mathbf{p}^*\|_1, \|\mathbf{q} - \mathbf{q}^*\|_1\} \leq C_A \|\Gamma\|_{\mathcal{I}_A \cup \mathcal{J}_A}.$$

Theorem 7.4.4 demonstrates that when game A has a unique NE, there is a safe zone around A, such that any game nearby has a unique NE that shares the same support with $\mathrm{NE}(A)$. Additionally, the distance between NEs of A and the

perturbed game $A + \Gamma$ is Lipschitz in the perturbation magnitude, which means NE of $A + \Gamma$ converges to NE of A as $\|\Gamma\|_{\mathcal{I}_A \cup \mathcal{J}_A} \to 0$.

Balcan and Braverman (2017) studies the computation of approximate NE for games with (ϵ, Δ)-perturbation stability. Theorem 7.4.4 shows that any two-player zero-sum game A with unique NE is $(\gamma, C_A\gamma)$-perturbation stable for $\gamma < \gamma_A$.

7.5 Applications to Corruption-Robust Offline Learning

Section 7.4 presents conditions for NE recovery when planning on a perturbed game matrix. We allow perturbations from any source as long as the perturbations and the game matrices satisfy the corresponding conditions.

In this section, we consider the corruption robust offline learning setting. Given a potentially corrupted offline dataset generated by a stochastic game with mean B, we use trimmed-mean to get an estimation $\widehat{B}$ with confidence bound Σ and calculate NE of $\widehat{B} \pm \Sigma$. Using trimmed-mean estimation, we reduce the learning problem to a planning problem with perturbation coming from both finite data and data corruption. We apply the results in Section 7.4 from two different aspects and get two different sets of results:

- by interpreting the true game matrix B as the original game A in Section 7.4, we get a set of *gap-dependent* results with a sample complexity bound;
- We also get a *certification* for NE (support) recovery when planning on $\widehat{B} \pm \Sigma$ by alternatively interpreting the estimated game $\widehat{B} \pm \Sigma$ as the original game A and B as the perturbed game $A + \Gamma$.

Corruption-Robust Offline Learning Setting and Classic Results

We start by introducing the corruption-robust offline learning setting. Consider a stochastic matrix game whose payoff is specified by a family of distributions $D_{[m]\times[n]}$. Suppose the expected payoff matrix is $B \in \mathbb{R}^{m\times n}$ and the variance of

payoff is bounded by σ^2 for some $\sigma > 0$. Specifically, for each $(i,j) \in [m]\times[n]$, $\mathbb{E}_{x\sim D_{i,j}}[x] = \mathbf{e}_i^\top B \mathbf{e}_j$ and $\mathbb{V}_{x\sim D_{i,j}}[x] \le \sigma^2$.

Suppose there is a data collector that uses an exploration policy $\mu \in \Delta([m]\times[n])$ to collect data. It takes a pair of pure strategy $(\mathbf{e}_i, \mathbf{e}_j)$ drawn from μ and observe $\tilde{x}_{i,j}$, an instantiation of the stochastic payoff $D_{i,j}$. This process is repeated for N times and results in an offline dataset: $\tilde{X} := \left\{(i_k, j_k, \tilde{x}^k_{i_k,j_k})\right\}_{k=1}^N$. On top of this, we consider the **ϵ-corruption model**, where the adversary can inspect the dataset $\tilde{X}$ and replace the payoff entry $x^k_{i_k,j_k}$ with arbitrary value. The only constraint is, for any $(i,j) \in [m]\times[n]$, there are at most ϵ-fraction of corrupted rewards. We use $X := \left\{(i_k, j_k, x^k_{i_k,j_k})\right\}_{k=1}^N$ to denote the corrupted dataset.

Our **goal** is to learn the NE of game B given the corrupted offline dataset X.

We assume that the behavior policy μ has uniform coverage on the extended NE support $(\mathcal{I}_B\times[n]) \cup ([m]\times\mathcal{J}_B)$:

Assumption 7.5.1. The behavior policy μ satisfies

$$C := \min_{(i,j)\in(\mathcal{I}_B\times[n])\cup([m]\times\mathcal{J}_B)} \mu_{i,j} > 0.$$

We present our algorithm in Algorithm 9, which uses Trimmed-Mean (the Univariate mean estimator in Lugosi and Mendelson (2021)) as a subroutine in the pessimistic algorithm (see PNVI in Cui and Du (2022)). It estimates $\widehat{B}$, the mean payoff of the game using the offline dataset, and designs the pessimistic bonuses Σ based on the error guarantee of the trimmed mean estimator. Due to the outlier removal step, Trimmed-Mean requires a minimum amount of samples to get a meaningful error upper bound. So we simply set the entries of Γ to $+\infty$ as a generic upper bound if there are no sufficient data. By concentration and coverage assumption Assumption 7.5.1, there would be sufficient data on $(\mathcal{I}_B\times[n])\cup([m]\times\mathcal{J}_B)$ to obtain a meaningful estimation given N large enough. The algorithm then constructs two pessimistically estimated games for both row and column players using estimation $\widehat{B}$ and bonus Σ. The algorithm then computes $(\underline{\mathbf{p}}, \overline{\mathbf{q}})$ as a conservative approximation for the NE of A.

Algorithm 9 Robust-PN

Input: Corrupted offline dataset X, corruption level ϵ, upper bound of variance σ^2, confidence level δ
for all $(i,j) \in [m]\times[n]$, let $N(i,j)$ be the count of (i,j) pair in X.
for $(i,j) \in [m]\times[n]$ **do**
  $\widehat{B}_{i,j} \leftarrow \textsc{Trimmed-Mean}\Big(\bigcup_{(i',j'):(i',j',x')\in X}\{x'\}\Big)$
  if $N(i,j) > 96\log\frac{8mn}{\delta}$ **then**
    $\Sigma_{i,j} \leftarrow 48\sigma\sqrt{\epsilon} + 86\sigma\sqrt{\frac{\log\frac{8mn}{\delta}}{N(i,j)}}$
  else
    $\Sigma_{i,j} \leftarrow +\infty$
  end if
end for
Compute the NE of $\widehat{B} - \Sigma$ as $(\underline{\mathbf{p}}, \underline{\mathbf{q}})$
Compute the NE of $\widehat{B} + \Sigma$ as $(\overline{\mathbf{p}}, \overline{\mathbf{q}})$
Return: $\hat{\pi} = (\underline{\mathbf{p}}, \overline{\mathbf{q}})$.

We first present an upper bound on the duality gap as a universal guarantee in the value space. This can be viewed as the worst-case gap-independent result, which holds with high probability, regardless of the game matrix:

Proposition 7.5.1. Suppose Assumption 7.5.1 holds. If

$$\epsilon < 1/32 \text{and} N > 768\Big(\log\frac{8mn}{\delta}\Big),$$

then with probability at least $1-\delta$, Algorithm 9 outputs a strategy pair $\hat{\pi}$, s.t.

$$\mathrm{Gap}(\hat{\pi}; B) = O\left(\sigma\sqrt{\epsilon} + \sigma\frac{\log\frac{mn}{\delta}}{\sqrt{CN}}\right) \tag{7.3}$$

Similar results also appear in Cui and Du (2022). At a high level, (7.3) includes a bias term $\sigma\sqrt{\epsilon}$ as the effect of data corruption and a statistical error term decreasing at the rate of $\frac{1}{\sqrt{N}}$. However, this result only guarantees the strategy achieves a good value but does not provide further insight regarding the game structure or NE recovery.

The TRIMMED-MEAN indeed reduces the problem of NE learning with data corruption to the problem of planning on the pessimistically estimated game matrices. In the following, we apply the results in Section 7.4 to show NE (support) recovery. The perturbation between B and $\widehat{B} - \Sigma$ or $\widehat{B} + \Sigma$ composes of the estimation error and the bonus matrix Σ. Interpreting B as the original game or perturbed leads to the following gap-dependent results and certifiable guarantees.

Prior Guarantee: Gap-dependent Results

The expected game matrix B can be viewed as the "original" game matrix and the estimated matrix $\widehat{B} - \Sigma$ or $\widehat{B} + \Sigma$ can be viewed as the perturbed matrices. By concentration and the error guarantee of TRIMMED-MEAN, with probability at least $1 - \delta$, for all $(i, j) \in (\mathcal{I}_B \times [n]) \cup ([m] \times \mathcal{J}_B)$,

$$\begin{aligned}\left|B_{i,j} - \widehat{B}_{i,j}\right| \leq & 48\sigma\sqrt{\epsilon} + 86\sigma\sqrt{\frac{\log\frac{8mn}{\delta}}{N(i,j)}} \\ \leq & 48\sigma\sqrt{\epsilon} + 258\sigma\frac{\log\frac{8mn}{\delta}}{\sqrt{CN}}.\end{aligned}$$

This means, with probability at least $1 - \delta$, the perturbation is bounded by:

$$\begin{aligned}\left\|\widehat{B} + \Sigma - B\right\|_{\mathcal{I}_B \cup \mathcal{J}_B} \leq & 2\left\|\Sigma\right\|_{\mathcal{I}_B \cup \mathcal{J}_B} \\ \leq & 96\sigma\sqrt{\epsilon} + 516\sigma\frac{\log\frac{8mn}{\delta}}{\sqrt{CN}} =: \gamma_1.\end{aligned}$$

Similarly,

$$\left\|\widehat{B} - \Sigma - B\right\|_{\mathcal{I}_B \cup \mathcal{J}_B} \leq \gamma_1.$$

According to Theorem 7.4.2, if B satisfies the pure base NE assumption and $\gamma_1 < \frac{1}{2}\min\{\Delta_{\mathcal{I}_B}, \Delta_{\mathcal{J}_B}\}$, then $\hat{\pi}$ outputted by Algorithm 9 is NE of B. We can similarly apply conditions for exact-NE-robust and NE-support-robust. More formally, we have

Corollary 7.5.1. *Suppose Assumption 7.5.1 holds,* $\epsilon < 1/32$ *and* $N > 768\left(\log \frac{8mn}{\delta}\right)$. *If*

- B *satisfies Assumption 7.4.1 and* $\gamma_1 < \frac{1}{2}\min\{\Delta_{\mathcal{I}_B}, \Delta_{\mathcal{J}_B}\}$;
- *OR* $|\mathcal{I}_B| = |\mathcal{J}_B| = 1$ *and* $\gamma_1 < \frac{1}{2}\min\{\Delta_{\mathcal{I}_B}, \Delta_{\mathcal{J}_B}\}$;
- *OR* B *has a unique NE* π^* *and* $\gamma_1 < \gamma_B$ *(defined in Theorem 7.4.4),*

then $\hat{\pi}$ *outputted by Algorithm 9 satisfies:*

- $\Pr[\hat{\pi} \in \mathrm{NE}(B)] \geq 1 - \delta$;
- *OR* $\Pr[\mathrm{NE}(B) = \{\hat{\pi}\}] \geq 1 - \delta$;
- *OR* $\Pr[\mathbf{supp}\,(\pi^*) = \mathbf{supp}\,(\hat{\pi}), \|\pi^* - \hat{\pi}\|_1 \leq C_B\gamma_1] \geq 1 - \delta$, *where* C_B *is defined in Theorem 7.4.4.*

Corollary 7.5.1 shows that, if we have some prior knowledge on the expected payoff matrix B, then we can utilize this information to plan beforehand. For example, if we know that A satisfies the pure base NE assumption and $96\sigma\sqrt{\epsilon} < \frac{1}{2}\min\{\Delta_{\mathcal{I}_B}, \Delta_{\mathcal{I}_B}\}$, then Corollary 7.5.1 provides a sample complexity bound: if

$$N > \left(\frac{516\sigma \log \frac{8mn}{\delta}}{\sqrt{C}\left(\frac{1}{2}\min\{\Delta_{\mathcal{I}_B}, \Delta_{\mathcal{I}_B}\} - 96\sigma\sqrt{\epsilon}\right)}\right)^2,$$

then with probability at least $1 - \delta$, Algorithm 9 outputs a NE of B. As a result, we also improve the bound in Proposition 7.5.1 to $\mathrm{Gap}(\hat{\pi}; B) = 0$. This means, even if the dataset is corrupted, as long as B satisfies the pure base NE assumption and $96\sigma\sqrt{\epsilon} < \frac{1}{2}\min\{\Delta_{\mathcal{I}_B}, \Delta_{\mathcal{I}_B}\}$, we can still learn a NE of B given sufficient data. We can similarly establish sample complexity for exact NE set recovery and NE support recovery. Wang et al. (2022) provides similar results for offline RL with an i.i.d. dataset.

Posterior Guarantee: Certifiable Results

Alternatively, we can also treat $\widehat{B} - \Sigma$ and $\widehat{B} + \Sigma$ as the "original" game matrix and treat B as the perturbed game matrix. Different from Corollary 7.5.1, we get a set of certifiable guarantees:

Corollary 7.5.2. *If $\epsilon < 1/32$, then $\hat{\pi}$ outputted by Algorithm 9 satisfies:*

$$\Pr[\textit{if } E^1_{\text{cond}} \textit{ holds, then } \mathrm{NE}(B) = \{\hat{\pi}\}] \geq 1 - \delta,$$
$$\Pr[\textit{if } E^2_{\text{cond}} \textit{ holds, then } E^2_{\text{result}}] \geq 1 - \delta,$$

where

$$\begin{aligned}
E^1_{\text{cond}} &:= E^1_1 \cap E^1_2 \cap E^1_3, \\
E^1_1 &:= \left\{ \left|\mathcal{I}_{\widehat{B}+\Sigma}\right| = \left|\mathcal{J}_{\widehat{B}+\Sigma}\right| = \left|\mathcal{I}_{\widehat{B}-\Sigma}\right| = \left|\mathcal{J}_{\widehat{B}-\Sigma}\right| = 1 \right\}, \\
E^1_2 &:= \left\{ \|\Sigma\|_{\mathcal{I}_{\widehat{B}+\Sigma} \cup \mathcal{J}_{\widehat{B}+\Sigma}} < \frac{1}{4} \min\left\{ \Delta_{\mathcal{I}_{\widehat{B}+\Sigma}}, \Delta_{\mathcal{J}_{\widehat{B}+\Sigma}} \right\} \right\}, \\
E^1_3 &:= \left\{ \|\Sigma\|_{\mathcal{I}_{\widehat{B}-\Sigma} \cup \mathcal{J}_{\widehat{B}-\Sigma}} < \frac{1}{4} \min\left\{ \Delta_{\mathcal{I}_{\widehat{B}-\Sigma}}, \Delta_{\mathcal{J}_{\widehat{B}-\Sigma}} \right\} \right\}, \\
E^2_{\text{cond}} &:= E^2_1 \cap E^2_2 \cap E^2_3 \cap E^2_4, \\
E^2_1 &:= \left\{ \widehat{B} + \Sigma \textit{ has a unique NE} \right\}, \\
E^2_2 &:= \left\{ \widehat{B} - \Sigma \textit{ has a unique NE} \right\}, \\
E^2_3 &:= \left\{ \ \|\Sigma\|_{\mathcal{I}_{\widehat{B}+\Sigma} \cup \mathcal{J}_{\widehat{B}+\Sigma}} < \tfrac{1}{2}\gamma_{\widehat{B}+\Sigma} \ \right\}, \\
E^2_4 &:= \left\{ \ \|\Sigma\|_{\mathcal{I}_{\widehat{B}-\Sigma} \cup \mathcal{J}_{\widehat{B}-\Sigma}} < \tfrac{1}{2}\gamma_{\widehat{B}-\Sigma} \ \right\}, \\
E^2_{\text{result}} &:= E^2_5 \cap E^2_6 \cap E^2_7 \cap E^2_8, \\
E^2_5 &:= \{A \textit{ has a unique NE } \pi^* = (\mathbf{p}^*, \mathbf{q}^*)\}, \\
E^2_6 &:= \{\mathbf{supp}\,(\pi^*) = \mathbf{supp}\,(\hat{\pi})\}, \\
E^2_7 &:= \left\{ \left\| \underline{\mathbf{p}} - \mathbf{p}^* \right\|_1 \leq 2C_{\widehat{B}-\Sigma} \|\Sigma\|_{\mathcal{I}_{\widehat{B}-\Sigma} \cup \mathcal{J}_{\widehat{B}-\Sigma}} \right\}, \\
E^2_8 &:= \left\{ \|\overline{\mathbf{q}} - \mathbf{q}^*\|_1 \leq 2C_{\widehat{B}+\Sigma} \|\Sigma\|_{\mathcal{I}_{\widehat{B}+\Sigma} \cup \mathcal{J}_{\widehat{B}+\Sigma}} \right\}.
\end{aligned}$$

The expressions in Corollary 7.5.1 and 7.5.2 are slightly different because the conditions E^1_{cond} and E^2_{cond} are stochastic events. Other than that, the conditions and results in Corollary 7.5.2 look similar to that in Corollary 7.5.1, but here is a key difference: the condition E^1_{cond} and E^2_{cond} in Corollary 7.5.2 can be verified numerically because both $\widehat{B}$ and Σ are computable for the learning agent. E^1_{cond} and E^2_{cond} thus serve as certifications for exact NE recovery and NE support recovery. Such results are meant to be used at the end of the learning process: after obtaining $\widehat{B}$, Σ and $\hat{\pi}$, one can verify whether $\hat{\pi}$ is NE of B by checking if E^1_{cond} or E^2_{cond} holds.

7.6 Conclusion

We study the perturbation-stability of NEs in two-player zero-sum games. We propose three levels of criteria for NE-robust. Our main results provide sufficient and necessary conditions or sufficient conditions for these criteria. We apply these conditions to corruption-robust offline learning settings, resulting in gap-dependent results and certifiable guarantees. Future research directions include extending the results to games with multiple mixed NEs and general-sum games.

In the robust decision-making problem, an adversary corrupts the data to manipulate the learning process. Fortunately, we've seen that robust statistics is an effective approach in decision-making problems against data corruption. However, when all agents seek to improve their own individual benefits without adversarial intent, the situation becomes much different. In this chapter, we establish a collaborative learning mechanism, serving as an alternative to robust statistics. Instead of detecting for data corruption, our mechanism incentivizes truthful data-sharing. As a result, empirical mean estimation is already sufficient for the problem.

8.1 Introduction

With the rise in popularity of machine learning, data is becoming an increasingly valuable resource for businesses, scientific organizations, and government institutions. However, data collection is often costly. For instance, to collect data, businesses may need to carry out market research, scientists may need to conduct experiments, and government institutions may need to perform surveys on public services. However, once data has been generated, it can be freely replicated and used by many organizations (Jones and Tonetti, 2020). Hence, instead of simply collecting and learning from their own data, by sharing data with each other, organizations can mutually reduce their own data collection costs and improve the utility they derive from data (Kairouz et al., 2021). In fact, there are already several platforms to facilitate data sharing among businesses (goo; Zheng et al., 2019), scientific organizations (pub; 53 and 68, 2013), and public institutions (Flores et al., 2021; Sheller et al., 2019).

However, simply pooling everyone's data and sharing with each other can lead to free-riding (Karimireddy et al., 2022; Sim et al., 2020). For instance, if an agent (e.g an organization) sees that other agents are already contributing a large amount of data, then, the cost she incurs to collect her own dataset may not offset

the marginal improvement in *her own* learned model due to diminishing returns of increasing dataset sizes (we describe this rigorously in Section 8.2). Hence, while she benefits from others' data, she has no incentive to collect and contribute data to the pool. A seemingly simple fix to this free-riding problem is to only return the datasets of the others if an agent submits a large enough dataset herself. However, this can be easily manipulated by a strategic agent who submits a large fabricated (fake) dataset without incurring any cost, receives the others' data, and then discards her fabricated dataset when learning. While the agent has benefited by this bad behavior, other agents who may use this fabricated dataset are worse off. Moreover, a naive test by the mechanism to check if the agent has fabricated data can be sidestepped by agents who collect only a small dataset and fabricate a larger dataset using this small dataset (e.g by fitting a model to the small dataset and then sampling from this fitted model).

In this work, we study these challenges in data sharing in one of the most foundational statistical problems, normal mean estimation, where the goal is to estimate the mean μ of a normal distribution $\mathcal{N}(\mu, \sigma^2)$ with known variance σ^2. We wish to design *mechanisms* for data sharing that satisfy the three fundamental desiderata of mechanism design; *Nash incentive compatibility* (*NIC*)*:* agents have incentive to collect a sufficiently large amount of data and share it truthfully provided that all other agents are doing so; *individual rationality* (*IR*)*:* agents are better off participating in the mechanism than working on their own; and *efficiency:* the mechanism leads to outcomes with small estimation error and data collection costs for all agents.

Contributions: (*i*) In Section 8.2, we formalize collaborative normal mean estimation in the presence of strategic agents. (*ii*) In Section 8.3, we design an NIC and IR mechanism for this problem to prevent free-riding and data fabrication and show that its social penalty, i.e sum of all agents' estimation errors and data collection costs, is at most twice that of the global minimum. (*iii*) In Appendix F.5, we study the same mechanism in high dimensional settings and relax the Gaussian assumption to distributions with bounded variance. We show that the mechanism retains its properties, with only a slight weakening of the NIC and efficiency guarantees. (*iv*) In Section 8.4, we consider two special cases where we impose natural

restrictions on the agents' strategy space. We show that it is possible to design mechanisms which essentially achieve the global minimum social penalty in both settings. Next, we will summarize our primary mechanism and the associated theorem in Section 8.3.

Summary of main results

Formalism: We assume that all agents have a fixed cost for collecting one sample, and define an agent's penalty (negative utility) as the sum of her estimation error and the cost she incurred to collect data. To make the problem well-defined, for the estimation error, we find it necessary to consider the *maximum risk*, i.e maximum expected error over all $\mu \in \mathbb{R}$. A mechanism asks agents to collect data, and then shares the data among the agents in an appropriate manner to achieve the three desiderata. An agent's strategy space consists of three components: how much data she wishes to collect, what she chooses to submit after collecting the data, and how she estimates the mean μ using the dataset she collected, the dataset she submitted, and the information she received from the mechanism.

Mechanism and theoretical result: In our mechanism, which we call C3D (Cross-Check and Corrupt based on Difference), each agent i collects a dataset X_i and submits a possibly fabricated or altered version Y_i to the mechanism. The mechanism then determines agent i's allocation in the following manner. It pools the data from the other agents and splits them into two subsets Z_i, Z_i'. Then, Z_i is returned as is, while Z_i' is corrupted by adding noise that is proportional to the difference between Y_i and Z_i. If an agent collects less or fabricates, she risks looking different to the others, and will receive a dataset Z_i' of poorer quality. We show that this mechanism has a Nash equilibrium where all agents collect a sufficiently large amount of data, submit it truthfully, and use a carefully weighted average of the three datasets X_i, Z_i, and Z_i' as their estimate for μ. The weighting uses some additional side information that the mechanism provides to each agent. Below, we state an informal version of the main theoretical result of this paper, which summarizes the properties of our mechanism.

Theorem 8.3.1 (informal): *The above mechanism is Nash incentive compatible, individually rational, and achieves a social penalty that is at most twice the globally minimum social penalty.*

Corruption is the first of two ingredients to achieving NIC. The second is the design of the weighted average estimator which is (minimax) optimal after corruption. To illustrate why this is important, say that the mechanism had assumed that the agents will use any other sub-optimal estimator (e.g a simple average). Then it will need to lower the amount of corruption to ensure IR and efficiency. However, a strategic agent will realize that she can achieve a lower maximum risk with a better estimator (instead of collecting more data herself and/or receiving less corrupt data from the mechanism). She can leverage this insight to collect less data and lower her overall penalty.

Proof techniques: The most challenging part of our analysis is to show NIC, First, to show minimax optimality of our estimator, we construct a sequence of normal priors for μ and show that the minimum Bayes' risk converges to the maximum risk of the weighted average estimator. However, when compared to typical minimax proofs, we face more significant challenges. The first of these is that the combined dataset $X_i \cup Z_i \cup Z_i'$ is neither independent nor identically distributed as the corruption is data-dependent. The second is that the agent's submission Y_i also determines the degree of corruption, so we cannot look at the estimator in isolation when computing the minimum Bayes' risk; we should also consider the space of functions an agent may use to determine Y_i from X_i. The third is that the expressions for the minimum Bayes' risk do not have closed form solutions and require non-trivial algebraic manipulations. To complete the NIC proof, we show that due to the carefully chosen amount of corruption, the agent should collect a sufficient amount of data to avoid excessive corruption, but not too much so as to increase her data collection costs.

Related Work

Mechanism design is one of the core areas of research in game theory (Vickrey, 1961; Groves, 1979; Clarke, 1971). Our work here is more related to mechanism design without payments, which has seen applications in fair division Procaccia (2013), matching markets (Roth, 1986), and kidney exchange (Roth et al., 2004) to name a few. There is a long history of work in the intersection of machine learning and mechanism design, although the overwhelming majority apply learning techniques when there is incomplete information about the mechanism or agent preferences, (e.g (Amin et al., 2013; Mansour et al., 2015; Athey and Segal, 2013; Nazerzadeh et al., 2008; Kakade et al., 2010)). On the flip side, some work have designed data marketplaces, where customers may purchase data from contributors (Agarwal et al., 2020c, 2019b; Jia et al., 2019; Wang et al., 2020b). These differ from our focus where we wish to incentivize agents to collaborate without payments.

Due to the popularity of shared data platforms (pub; Flores et al., 2021; Sheller et al., 2019; goo) and federated learning (Kairouz et al., 2021), there has been a recent interest in designing mechanisms for data sharing. Sim et al. (2020) and Xu et al. (2021b) study fairness in collaborative data sharing, where the goal is to reward agents according to the amount of data they contribute. However, their mechanisms do not apply when strategic agents may try to manipulate a mechanism. Blum et al. (2021) and Karimireddy et al. (2022) study collaboration in federated learning. However, the strategy space of an agent is restricted to how much data they collect and their mechanism rewards each agent according to the quantity of the data she submitted. The above four works recognize that free-riding can be detrimental to data sharing, but assume that agents will not fabricate data. As discussed above, if this assumption is not true, agents can easily manipulate such mechanisms. Fraboni et al. (2021) and Lin et al. (2019) study federated learning settings where free-riders may send in fabricated gradients without incurring the computational cost of computing the gradients. However, their focus is on designing gradient descent algorithms that are robust to such attacks and not on incentivizing agents to perform the gradient computations. Some work have designed mechanisms

for federated learning so as to elicit private information (such as data collection costs), but their focus is not on preventing free-riding or fabrication (Ding et al., 2020b; Liu et al., 2022). Miller et al. Miller et al. (2005) uses scoring systems to develop mechanisms that prevent signal fabrication. However, the agents in their settings can only choose to report either their true signal or something else but can not freely choose how much data to collect. Cai et al. Cai et al. (2015) study mechanism design where a learner incentivizes agents to collect data via payments. Their mechanism, which also cross-checks the data submitted by the agents, has connections to our setting in Section 8.4 where we consider a restricted strategy space for the agents.

Our approach of using corruption to engender good behaviour draws inspiration from the robust estimation literature, which design estimators that are robust to data from malicious agents (Diakonikolas et al., 2016; Lugosi and Mendelson, 2021; Chen et al., 2023). However, to the best of our knowledge, the specific form of corruption and the subsequent design of the minimax optimal estimator are new in this work, and require novel analysis techniques.

8.2 Problem Setup

We will now formally define our problem. We have m agents, who are each able to collect i.i.d samples from a normal distribution $\mathcal{N}(\mu, \sigma^2)$, where σ^2 is known. They wish to estimate the mean μ of this distribution. To collect one sample, the agent has to incur a cost c. We will assume that σ^2, c, and m are public information. However, $\mu \in \mathbb{R}$ is unknown, and no agent has auxiliary information, such as a prior, about μ. An agent wishes to minimize her estimation error, while simultaneously keeping the cost of data collection low. While an agent may collect data on her own to manage this trade-off, by sharing data with other agents, she can reduce costs while simultaneously improving her estimate. We wish to design mechanisms to facilitate such sharing of data.

Mechanism: A mechanism receives a dataset from each agent, and in turn returns

an *allocation* A_i to each agent. An agent will use her allocation to estimate μ. This allocation could be, for instance, a larger dataset obtained with other agents' datasets. The mechanism designer is free to choose a space of allocations $\mathcal{A}$ to achieve the desired goals. Formally, we define a mechanism as a tuple $M = (\mathcal{A}, b)$ where $\mathcal{A}$ denotes the space of allocations, and b is a procedure to map the datasets collected from the m agents to m allocations. Denoting the universal set by $\mathcal{U}$, we write the space of mechanisms $\mathcal{M}$ as

$$\mathcal{M} = \{M = (\mathcal{A},\, b) : \quad \mathcal{A} \subset \mathcal{U}, \quad b : (\textstyle\bigcup_{n \geq 0} \mathbb{R}^n)^m \to \mathcal{A}^m\}. \tag{8.1}$$

As is customary, we will assume that the mechanism designer will publish the space of allocations $\mathcal{A}$ and the mapping b (the procedure used to obtain the allocations) ahead of time, so that agents can determine their strategies. However, specific values computed/realized during the execution of the mechanism are not revealed, unless the mechanism chooses to do so via the allocation A_i.

Agents' strategy space: Once the mechanism is published, the agent will choose a strategy. In our setting, this will be the tuple (n_i, f_i, h_i), which determines how much data she wishes to collect, what she chooses to submit, and how she wishes to estimate the mean μ. First, the agent samples n_i points to collect her initial dataset $X_i = \{x_{i,j}\}_{j=1}^{n_i}$, where $x_{i,j} \sim \mathcal{N}(\mu, \sigma^2)$, incurring cn_i cost. She then submits $Y_i = \{y_{i,j}\}_j = f_i(X_i)$ to the mechanism. Here f_i is a function which maps the collected dataset to a possibly fabricated or falsified dataset of a potentially different size. In particular, this fabrication can depend on the data she has collected. For instance, the agent could collect only a small dataset, fit a Gaussian, and then sample from it.

Finally, the mechanism returns the agent's allocation A_i, and the agent computes an estimate $h_i(X_i, Y_i, A_i)$ for μ using her initial dataset X_i, the dataset she submitted Y_i, and the allocation she received A_i. We include Y_i as part of the estimate since an agent's submission may affect the allocation she receives. Consequently, agents could try to elicit additional information about μ via a carefully chosen Y_i. We can write the strategy space of an agent as $\mathcal{S} = \mathbb{N} \times \mathcal{F} \times \mathcal{H}$, where $\mathcal{F}$ is the space

of functions mapping the dataset collected to the dataset submitted, and $\mathcal{H}$ is the space of all estimators using all the information she has. We have:

$$\mathcal{F} = \{f : \bigcup_{n\geq 0} \mathbb{R}^n \to \bigcup_{n\geq 0} \mathbb{R}^n\}, \qquad \mathcal{H} = \{h : \bigcup_{n\geq 0} \mathbb{R}^n \times \bigcup_{n\geq 0} \mathbb{R}^n \times \mathcal{A} \to \mathbb{R}\}. \tag{8.2}$$

One element of interest in $\mathcal{F}$ is the identity $\mathbf{I}$ which maps a dataset to itself. A mechanism designer would like an agent to use $f_i = \mathbf{I}$, i.e to submit the data that she collected as is, so that other agents can benefit from her data.

Going forward, when $s = \{s_i\}_i \in \mathcal{S}^m$ denotes the strategies of all agents, we will use $s_{-i} = \{s_j\}_{j\neq i}$ to denote the strategies of all agents except i. Without loss of generality, we will assume that agent strategies are deterministic. If they are stochastic, our results will carry through for every realization of any external source of randomness that the agent uses to determine (n_i, f_i, h_i).

Agent penalty: The agent's *penalty* p_i (i.e negative utility) is the sum of her squared estimation error and the cost cn_i incurred to collect her dataset X_i of n_i points. The agent's penalty depends on the mechanism M and the strategies $s = \{s_j\}_j$ of all the agents. Making this explicit, p_i is defined as:

$$p_i(M, s) = \sup_{\mu\in\mathbb{R}} \mathbb{E}\left[(h_i(X_i, Y_i, A_i) - \mu)^2 \,\middle|\, \mu\right] + cn_i \tag{8.3}$$

The term inside the expectation is the squared difference between the agent's estimate and the true mean (conditioned on the true mean μ). The expectation is with respect to the randomness of all agents' data and possibly any randomness in the mechanism. We consider the *maximum risk*, i.e supremum over $\mu \in \mathbb{R}$, since the true mean μ is unknown to the agent a priori, and their strategy should yield good estimates, and hence small penalty, over all possible values μ. To illustrate this further, note that when the value of true mean μ is μ', the optimal strategy for an agent will always be to not collect any data and choose the estimator $h_i(\cdot, \cdot, \cdot) = \mu'$ leading to 0 penalty. However, this strategy can be meaningfully realized by an agent only if she knew that $\mu = \mu'$ a priori which renders the problem meaningless[1].

[1]This is akin to the reason why it is customary to study the maximum risk in frequentist

Considering the maximum risk accounts for the fact that μ is unknown and makes the problem well-defined.

Recommended strategies: In addition to publishing the mechanism, the mechanism designer will recommend strategies $s^\star = \{s_i^\star\}_i \in \mathcal{S}^m$ for the agents so as to incentivize collaboration and induce optimal social outcomes.

Desiderata: We can now define the three desiderata for a mechanism:

1. *Nash Incentive compatibility (NIC):* A mechanism $M = (\mathcal{A}, b)$ is said to be NIC at the recommended strategy profile $s^\star$ if, for each agent i, and for every other alternative strategy $s_i \in \mathcal{S}$ for that agent, we have $p_i(M, s^\star) \leq p_i(M, (s_i, s^\star_{-i}))$. That is, $s^\star$ is a Nash equilibrium so no agent has incentive to deviate if all other agents are following $s^\star$.

2. *Individual rationality (IR):* We say that a mechanism M is IR at $s^\star$ if no agent suffers from a higher penalty by participating in the mechanism than the lowest possible penalty she could achieve on her own when all other agents are following $s^\star$. If an agent does not participate, she does not submit nor receive any data from the mechanism; she will simply choose how much data to collect and design the best possible estimator. Formally, we say that a mechanism M is IR if the following is true for each agent i:

$$p_i(M, s^\star) \leq \inf_{n_i \in \mathbb{N},\, h_i \in \mathcal{H}} \left\{ \sup_{\mu \in \mathbb{R}} \mathbb{E}\left[(h_i(X_i, \varnothing, \varnothing) - \mu)^2 \,|\, \mu \right] \;+\; c n_i \right\}. \tag{8.4}$$

3. *Efficiency:* The *social penalty* $P(M, s)$ of a mechanism M when agents follow strategies s, is the sum of agent penalties (defined below). We define $\mathrm{PR}(M, s^\star)$ to be the ratio between the social penalty of a mechanism at the recommended strategies $s^\star$, and the lowest possible social penalty among all possible mechanisms

statistics (Lehmann and Casella, 2006; Wald, 1939). An alternative approach is to take a Bayesian view, considering a prior on μ and using the Bayes' risk $\mathbb{E}_\mu[\mathbb{E}[(h_i(X_i, Y_i, A_i) - \mu)^2 | \mu]]$ instead of the maximum risk in p_i. While we have adopted a frequentist formalism here, our main proof ideas can be ported over to the Bayesian setting as well (See Appendix F.6 for more details)

and strategies (*without* NIC or IR constraints). We have:

$$P(M,s) = \sum_{i\in[m]} p_i(M,s), \qquad \mathrm{PR}(M,s^\star) = \frac{P(M,s^\star)}{\inf_{M'\in\mathcal{M},\, s\in\mathcal{S}^m} P(M',s)} \tag{8.5}$$

Note that $\mathrm{PR} \geq 1$. We say that a mechanism is efficient if $\mathrm{PR}(M,s^\star) = 1$ and that it is approximately efficient if $\mathrm{PR}(M,s^\star)$ is bounded by some constant that does not depend on m. If $s^\star$ is a Nash equilibrium, then $\mathrm{PR}(M,s^\star)$ can be viewed as an upper bound on the price of stability (Anshelevich et al., 2008).

For what follows, we will discuss optimal strategies for agents working on her own and present a simple mechanism which minimizes the social penalty, but has a poor Nash equilibrium.

Optimal strategies for an agent working on her own: Recall that, given n samples $\{x_i\}_{i=1}^n$ from $\mathcal{N}(\mu,\sigma^2)$, the sample mean is a minimax optimal estimator (Lehmann and Casella, 2006); i.e among all possible estimators h, the sample mean minimizes the maximum risk $\sup_{\mu\in\mathbb{R}} \mathbb{E}[(\mu - h(\{x_i\}_{i=1}^n,\varnothing,\varnothing))^2 \,|\, \mu]$ (note that the agent only has the dataset she collected). Moreover, its mean squared error is σ^2/n for all $\mu\in\mathbb{R}$. Hence, an agent acting on her own will choose the sample mean and collect $n_i = \sigma/\sqrt{c}$ samples so as to minimize their penalty; as long as the amount of data is less than $\sigma/\sqrt{c}$, an agent has incentive to collect more data since the cost of collecting one more point is offset by the marginal decrease in estimation error. This can be seen via the following simple calculation:

$$\inf_{\substack{n_i\in\mathbb{R}\\ h_i\in\mathcal{H}}} \Big(\sup_{\mu} \mathbb{E}\left[(h_i(X_i,\varnothing,\varnothing) - \mu)^2 \,\middle|\, \mu \right] \;+\; c n_i \Big) = \min_{n_i\in\mathbb{R}} \Big(\frac{\sigma^2}{n_i} + c n_i \Big) = 2\sigma\sqrt{c} \;\triangleq\; p_{\min}^{\mathrm{IR}}\,. \tag{8.6}$$

Let $p_{\min}^{\mathrm{IR}} = 2\sigma\sqrt{c}$ denote the lowest achievable penalty by an agent working on her own. If all m agents work independently, then the total social penalty is $m p_{\min}^{\mathrm{IR}} = 2\sigma m\sqrt{c}$. Next, we will look at a simple mechanism and an associated set of strategies which achieve the global minimum penalty. This will show that it is possible for all

agents to achieve a significantly lower penalty via collaboration.

A globally optimal mechanism *without* strategic considerations: The following simple mechanism M_{pool}, pools all the data from the other agents and gives it back to an agent. Precisely, it chooses the space of allocation $\mathcal{A} = \bigcup_{n\geq 0} \mathbb{R}^n$ to be datasets of arbitrary length, and sets agent i's allocation to be $A_i = \bigcup_{j\neq i} Y_i$. The recommended strategies $s^{\text{pool}} = \{(n_i^{\text{pool}}, f_i^{\text{pool}}, h_i^{\text{pool}})\}_i$ asks each agent to collect $n_i^{\text{pool}} = \sigma/\sqrt{cm}$ points[2], submit it as is $f_i^{\text{pool}} = \mathbf{I}$, and use the sample mean of all points as her estimate $h_i^{\text{pool}}(X_i, X_i, A_i) = \frac{1}{|X_i \cup A_i|} \sum_{z\in X_i\cup A_i} z$. It is straightforward to show that this minimizes the social penalty if all agents follow s^{pool}. After each agent has collected their datasets $\{X_i\}_i$, the social penalty is minimized if all agents have access to each other's datasets and they all use a minimax optimal estimator: this justifies using M_{pool} with $f_i^{\text{pool}} = \mathbf{I}$ and setting h_i^{pool} to be the sample mean. The following simple calculation justifies the choice of $\sum_i n_i^{\text{pool}}$:

$$\inf_{s\in\mathcal{S}^m} \sum_{i=1}^{m} \left(\sup_{\mu} \mathbb{E}\left[(h_i(X_i, f_i, A_i) - \mu)^2 \,\middle|\, \mu \right] + cn_i \right) = \min_{\{n_i\}_i} \left(\frac{m\sigma^2}{\sum_i n_i} + c\sum_i n_i \right) = 2\sigma\sqrt{mc}.$$

However, s^{pool} is not a Nash equilibrium of this mechanism, as an agent will find it beneficial to free-ride. If all other agents are submitting $\sigma/\sqrt{cm}$ points, by collecting no points, an agent's penalty is $\sigma\sqrt{mc}/(m-1)$, as she does not incur any data collection cost. This is strictly smaller than $2\sigma\sqrt{c/m}$ when $m \geq 3$. In fact, it is not hard to show that M_{pool} is at a Nash equilibrium only when the total amount of data is $\sigma/\sqrt{c}$; for additional points, the marginal reduction in the estimation error for an individual agent does not offset her data collection costs. The social penalty at these equilibria is $\sigma\sqrt{c}(m+1)$ which is significantly larger than the global minimum when there are many agents.

A seemingly simple way to fix this mechanism is to only return the datasets of the other agents if an agent submits at least $\sigma/\sqrt{cm}$ points. However, as we will see in Section 8.4, such a mechanism can also be manipulated by an agent who submits a fabricated dataset of $\sigma/\sqrt{cm}$ points without actually collecting any data

[2]To avoid rounding effects, henceforth we will treat $\sigma/\sqrt{cm}$, and $\sigma/\sqrt{c}$ as integers.

Algorithm 10 M_{C3D}

Mechanism designer publishes:
The allocation space $\mathcal{A} = \bigcup_{n\geq 0} \mathbb{R}^n \times \bigcup_{n\geq 0} \mathbb{R}^n \times \mathbb{R}_+$, and the procedure in lines 6–15.
Each agent i**:**
Choose strategy $s_i = (n_i, f_i, h_i)$. # See (8.8) for recommended strategies.
Sample n_i points $X_i = \{x_{i,j}\}_{j=1}^{n_i}$ and submit $Y_i = f_i(X_i)$ to the mechanism.
Mechanism:
For each agent $i \in [m]$: # can be done simultaneously for all agents
$Y_{-i} \leftarrow \bigcup_{j\neq i} Y_j$.
If $m \leq 4$: # Simply pool and return all of the other agents' data to agent i.
$A_i \leftarrow (Y_{-i}, \varnothing, 0)$. Return A_i to agent i.
Else:
$Z_i \leftarrow$ sample $\min\{|Y_{-i}|, \sigma/\sqrt{cm}\}$ points in Y_{-i} without replacement.
$\eta_i^2 \leftarrow \alpha^2 \left(\frac{1}{|Y_i|}\sum_{y\in Y_i} y - \frac{1}{|Z_i|}\sum_{z\in Z_i} z\right)^2$ # See (8.7) for α.
$Z_i' \leftarrow \{z + \epsilon_{z,i},$ for all $z \in Y_{-i}\backslash Z_i$ where $\epsilon_{z,i} \sim \mathcal{N}(0, \eta_i^2)\}$
$A_i \leftarrow (Z_i, Z_i', \eta_i^2)$. Return A_i to agent i.
Each agent i**:**
Compute estimate $h_i(X_i, Y_i, A_i)$. # See (8.8) for recommended estimator.

and incurring any cost and then discarding the fabricated dataset when estimating. Any naive test to check for the quality of the data can also be sidestepped by agents who sample only a few points, and use that to fabricate a larger dataset (e.g by sampling a large number of points from a Gaussian fitted to the small sample). Next, we will present our mechanism for this problem which satisfies all three desiderata.

8.3 Method and Results

We have outlined our mechanism M_{C3D}, and its interaction with the agents in Algorithm 10 in the natural order of events. We will first describe it procedurally, and then motivate our design choices. Our mechanism uses the following allocation space, $\mathcal{A} = \bigcup_{n\geq 0} \mathbb{R}^n \times \bigcup_{n\geq 0} \mathbb{R}^n \times \mathbb{R}_+$. An allocation $A_i = (Z_i, Z_i', \eta_i^2) \in \mathcal{A}$ consists of an uncorrupted dataset Z_i, a corrupted dataset Z_i', and the variance η_i^2 of the

noise added to Z_i' for corruption. Once the mechanism and the allocation space are published, agent i chooses her strategy $s = (n_i, f_i, h_i)$. She collects a dataset $X_i = \{x_{i,j}\}_{j=1}^{n_i}$, where $x_{i,j} \sim \mathcal{N}(\mu, \sigma^2)$, and submits $Y_i = f_i(X_i)$ to the mechanism.

Our mechanism determines agent i's allocation as follows. Let Y_{-i} be the union of all datasets submitted by the other agents. If there are at most four agents, we simply return all of the other agents' data without corruption by setting $A_i \leftarrow (Y_{-i}, \varnothing, 0)$. If there are more agents, the mechanism first chooses a random subset of size $\min\{|Y_{-i}|, \sigma/\sqrt{cm}\}$ from Y_{-i}; denote this Z_i. In line 13, the mechanism individually adds Gaussian noise to the remaining points $Y_{-i} \backslash Z_i$ to obtain Z_i' (line 14). The variance η_i^2 of the noise depends on the difference between the sample means of the subset Z_i and the agent's submission Y_i. It is modulated by a value α, which is a function of c, m, and σ^2. Precisely, α is the smallest number larger than $\sqrt{\sigma}(cm)^{-1/4}$ which satisfies $G(\alpha) = 0$, where:

$$G(\alpha) := \left(\frac{m-4}{m-2}\frac{4\alpha^2}{\sigma/\sqrt{cm}} - 1\right)\frac{4\alpha}{\sqrt{\sigma}(m/c)^{1/4}} - \left(4(m+1)\frac{\alpha^2}{\sigma\sqrt{m/c}} - 1\right)\sqrt{2\pi}\exp\left(\frac{\sigma\sqrt{m/c}}{8\alpha^2}\right)\operatorname{Erfc}\left(\frac{\sqrt{\sigma}(m/c)^{1/4}}{2\sqrt{2}\alpha}\right) \tag{8.7}$$

Finally, the mechanism returns the allocation $A_i = (Z_i, Z_i', \eta_i^2)$ to agent i and the agent estimates μ.

Recommended strategies: The recommended strategy $s_i^\star = (n_i^\star, f_i^\star, h_i^\star)$ for agent i is given in (8.8). The agent should collect $n_i^\star = \sigma/(m\sqrt{c})$ samples if there are at most four agents, and $n_i^\star = \sigma/\sqrt{cm}$ samples otherwise. She should submit it without fabrication or alteration $f_i = \mathbf{I}$, and then use a weighted average of the datasets (X_i, Z_i, Z_i') to estimate μ. The weighting is proportional to the inverse variance of the data. For X_i and Z_i this is simply σ^2, but for Z_i', the variance is $\sigma^2 + \eta_i^2$ since the mechanism adds Gaussian noise with variance η_i^2. We have:

$$n_i^\star = \begin{cases} \frac{\sigma}{m\sqrt{c}} & \text{if } m \le 4 \\ \frac{\sigma}{\sqrt{cm}} & \text{if } m > 4 \end{cases}, \qquad f_i^\star = \mathbf{I},$$

$$h_i^\star(X_i, Y_i, (Z_i, Z_i', \eta_i^2)) = \frac{\frac{1}{\sigma^2}\sum_{u \in X_i \cup Z_i} u + \frac{1}{\sigma^2+\eta_i^2}\sum_{u \in Z_i'} u}{\frac{1}{\sigma^2}|X_i \cup Z_i'| + \frac{1}{\sigma^2+\eta_i^2}|Z_i'|} \tag{8.8}$$

Design choices: Next, we will describe our design choices and highlight some key challenges. When $m \leq 4$, it is straightforward to show that the mechanism satisfies all our desired properties (see beginning of Section 8.3), so we will focus on the case $m > 4$. First, recall that the mechanism needs to incentivize agents to collect a sufficient amount of samples. However, simply counting the number of samples can be easily manipulated by an agent who simply submits a fabricated dataset of a large number of points. Instead, Algorithm 10 attempts to infer the quality of the data submitted by the agents using how well an agent's submission Y_i approximates μ. Ideally, we would set the variance η_i^2 of this corruption to be proportional to the difference $(\frac{1}{|Y_i|}\sum_{y\in Y_i} y - \mu)^2$, so that the more data she submits, the less the variance of Z_i', which in turn yields a more accurate estimate for μ. However, since μ is unknown, we use a subset Z_i obtained from other agents' data as a proxy for μ, and set η_i^2 proportional to $(\frac{1}{|Y_i|}\sum_{y\in Y_i} y - \frac{1}{|Z_i|}\sum_{z\in Z_i} z)^2$. If all agents are following $s^\star$, then $|Y_i| = |Z_i| = \sigma/\sqrt{cm} = n_i^\star$; it is sufficient to use only $n_i^\star$ points for validating Y_i since both $\frac{1}{|Y_i|}\sum_{y\in Y_i} y$ and $\frac{1}{|Z_i|}\sum_{z\in Z_i} z$ will have the same order of error in approximating μ.

The second main challenge is the design of the recommended estimator $h_i^\star$. In Section 8.3 we show how splitting Y_{-i} into a clean and corrupted parts Z_i, Z_i' allows us to design a minimax optimal estimator. A minimax optimal estimator is crucial to achieving NIC. To explain this, say that the mechanism assumes that agents will use a sub-optimal estimator, e.g sample mean of $X_i \cup Z_i \cup Z_i'$. Then, to account for the larger estimation error, it will need to choose a lower level of corruption η_i^2 to minimize the social penalty. However, a smart agent will realize that she can achieve a lower maximum risk by using a better estimator, such as the weighted average, instead of collecting more data in order to reduce the amount of corruption used by the mechanism. She can leverage this insight to collect less data and reduce her overall penalty.

This concludes the description of our mechanism. The following theorem, which is the main theoretical result of this paper, states that M_{C3D} achieves the three desiderata outlined in Section 8.2.

Theorem 8.3.1. *Let $m > 1$, α be as defined in (8.7), and $s_i^\star$ be as defined in (8.8). Then, the following statements are true about the mechanism M_{C3D} in Algorithm 10. (i) The strategy profile $s^\star$ is a Nash equilibrium. (ii) The mechanism is individually rational at $s^\star$. (iii) The mechanism is approximately efficient, with $\text{PR}(M_{\text{C3D}}, s^\star) \leq 2$.*

The mechanism is NIC as, provided that others are following $s_i^\star$, there is no reason for any one agent to deviate. Moreover, we achieve low social penalty at $s_i^\star$. Other than $s^\star$, there is also a set of similar Nash equilibria with the same social penalty: the agents can each add a same constant to the data points they collect and subtract the same value from the final estimate. Before we proceed, the expression for α in (8.7) warrants explanation. If we treat α is a variable, we find that different choices of α can lead to other Nash equilibria with corresponding bounds on PR. This specific choice of α leads to a Nash equilibrium where agents collect $\sigma/\sqrt{cm}$ points, and a small bound on PR. Throughout this manuscript, we will treat α as the specific value obtained by solving (8.7), and *not* as a variable.

High dimensional non-Gaussian distributions: In Appendix F.5, we study M_{C3D} when applied to d–dimensional distributions. In Theorem F.5.1, we show that under bounded variance assumptions, $s^\star$ is an ε_m-approximate Nash equilibrium and that $\text{PR}(M_{\text{C3D}}, s^\star) \leq 2 + \varepsilon_m$ where $\varepsilon_m \in \mathcal{O}(1/m)$.

Proof sketch of Theorem 8.3.1

When $m \leq 4$: First, consider the (easy) case $m \leq 4$. At $s_i^\star$, the total amount of data collected is $\sigma/\sqrt{c}$ (see $n_i^\star$ in (8.8)), and as there is no corrupted dataset, $h_i^\star$ simply reduces to the sample mean of $X_i \cup Y_{-i}$. The mechanism is IR since an agent's penalty will be $\sigma\sqrt{c}(1 + 1/m)$ which is smaller than $p_{\min}^{\text{IR}}$ (8.6). It is approximately efficient since the social penalty is $\sigma\sqrt{c}(m + 1)$ which is at most twice the global minimum $2\sigma\sqrt{mc}$ when $m \leq 4$. Finally, NIC is guaranteed by the same argument used in (8.6); as long as the total amount of data is less than $\sigma/\sqrt{c}$, the cost of collecting one more point is offset by the marginal decrease in the estimation error; hence, the agent is incentivized to collect more data. Moreover, as A_i does not

depend on f_i under these conditions, there is no incentive to fabricate or falsify data.

When $m > 4$: We will divide this proof into four parts. We first show that $G(\alpha) = 0$ in line (6) has a solution α larger than $\sqrt{n_i^\star} = \sqrt{\sigma}(cm)^{-1/4}$. This will also be useful when analyzing the efficiency.

1. Equation (8.7) has a solution. We derive an asymptotic expansion of $\mathrm{Erfc}(\cdot)$ using integration by parts to analyze the solution to (8.7). When $m \geq 5$, we show that $G(\sqrt{n_i^\star}) \times G(\sqrt{n_i^\star}(1 + 8/\sqrt{m})) < 0$. By continuity of G, there exists $\alpha_m \in (\sqrt{n_i^\star}, \sqrt{n_i^\star}(1 + 8/\sqrt{m}))$ s.t. $G(\alpha_m) = 0$. For m large enough such that the residual in the asymptotic expansion is negligible, we show $\alpha_m \in (\sqrt{n_i^\star}, \sqrt{n_i^\star}(1 + \log m/m))$ via an identical technique.

2. The strategies $s^\star$ in (8.8) is a Nash equilibrium: We show this via the following two steps. First (**2.1**), We show that fixing any n_i, the maximum risk and thus the penalty p_i is minimized when agent i submits the raw data and uses the weighted average as specified in (8.8), i.e for all n_i,

$$p_i(M_{\text{C3D}}, ((n_i, f_i^\star, h_i^\star), s_{-i}^\star)) \leq p_i(M_{\text{C3D}}, ((n_i, f_i, h_i), s_{-i}^\star)), \quad \forall (n_i, f_i, h_i) \in \mathbb{N} \times \mathcal{F} \times \mathcal{H}. \tag{8.9}$$

Second (**2.2**), we show that p_i is minimized when agent i collects $n_i^\star$ samples under $(f_i^\star, h_i^\star)$, i.e.

$$p_i(M_{\text{C3D}}, ((n_i^\star, f_i^\star, h_i^\star), s_{-i}^\star)) \leq p_i(M_{\text{C3D}}, ((n_i, f_i^\star, h_i^\star), s_{-i}^\star)), \qquad \forall\, n_i \in \mathbb{N}. \tag{8.10}$$

***2.1: Proof of* (8.9).** As the data collection cost does not change for fixed n_i, it is sufficient to show that $(f_i^\star, h_i^\star)$ minimizes the maximum risk. Our proof is inspired by the following well-known recipe for proving minimax optimality of an estimator Lehmann and Casella (2006): design a sequence of priors $\{\Lambda_\ell\}_\ell$, compute the minimum Bayes' risk $\{R_\ell\}_\ell$ for any estimator, and then show that R_ℓ converges to the maximum risk of the proposed estimator as $\ell \to \infty$.

To apply this recipe, we use a sequence of normal priors $\Lambda_\ell = \mathcal{N}(0, \ell^2)$ for μ. Howeiver, before we proceed, we need to handle two issues. The first of these concerns the posterior for μ when conditioned on (X_i, Z_i, Z_i'). Since the corruption terms $\epsilon_{z,i}$ added to Z_i' depend on X_i and Z_i, this dataset is not independent. Moreover, as the variance η_i^2 is the difference between two normal random variables, Z_i' is not normal. Despite these, we are able to show that the posterior $\mu|(X_i, Z_i, Z_i')$ is normal. The second challenge is that the submission f_i also affects the estimation error as it determines the amount of noise η_i^2. We handle this by viewing $\mathcal{F} \times \mathcal{H}$ as a rich class of estimators and derive the optimal Bayes' estimator $(f_{i,\ell}^{\mathrm{B}}, h_{i,\ell}^{\mathrm{B}}) \in \mathcal{F} \times \mathcal{H}$ under the prior Λ_ℓ. We then show that the minimum Bayes' risk converges to the maximum risk when using $(f_i^\star, h_i^\star)$.

Next, under the prior $\Lambda_\ell = \mathcal{N}(0, \ell^2)$, we can minimize the Bayes' risk with respect to $h_i \in \mathcal{H}$ by setting $h_{i,\ell}^{\mathrm{B}}$ to be the posterior mean. Then, the minimum Bayes' risk R_ℓ can be written as,

$$R_\ell = \inf_{f_i \in \mathcal{F}} \mathbb{E}\left[\left(|Z_i'|\left(\sigma^2 + \alpha^2\left(\frac{1}{|Y_i|}\sum_{y \in Y_i} y - \frac{1}{|Z_i|}\sum_{z \in Z_i} z\right)^2\right)^{-1} + \frac{|X_i| + |Z_i|}{\sigma^2} + \frac{1}{\ell^2}\right)^{-1}\right]$$

Note that $Y_i = f_i(X_i)$ depends on f_i. Via the Hardy-Littlewood inequality (Burchard, 2009), we can show that the above quantity is minimized when $f_{i,\ell}^{\mathrm{B}}$ is chosen to be a shrunk version of the agent's initial dataset X_i, i.e $f_{i,\ell}^{\mathrm{B}}(X_i) = \{(1 + \sigma^2/(|X|\ell^2))^{-1}\, x,\ \forall\, x \in X_i\}$. This gives us an expression for the minimum Bayes' risk R_ℓ under prior Λ_ℓ. To conclude the proof, we note that the minimum Bayes' risk under any prior is a lower bound on the maximum risk, and show that R_ℓ approaches the maximum risk of $(f_i^\star, h_i^\star)$ from below. Hence, $(f_i^\star, h_i^\star)$ is minimax optimal for any n_i. (Above, it is worth noting that $f_{i,\ell}^{\mathrm{B}} \to f_i^\star = \mathbf{I}$ as $\ell \to \infty$. In the Appendix, we also find that $h_{i,\ell}^{\mathrm{B}} \to h_i^\star$.)

2.2: Proof of (8.10)**.** We can now write $p_i(M_{\text{C3D}}, ((n_i, f_i^\star, h_i^\star), s_{-i}^\star)) = R_\infty + cn_i$, where R_∞ is the maximum risk of $(f_i^\star, h_i^\star)$ (and equivalently, the limit of the mini-

mum Bayes' risk):

$$R_\infty := \mathbb{E}_{x\sim\mathcal{N}(0,1)}\left[\left((m-2)n_i^\star\left(\sigma^2+\alpha^2\left(\sigma^2/n_i+\sigma^2/n_i^\star\right)x^2\right)^{-1}+(n_i+n_i^\star)\sigma^{-2}\right)^{-1}\right]$$

The term inside the expectation is convex in n_i for each fixed x. As expectation preserves convexity, we can conclude that p_i is a convex function of n_i. The choice of α in (8.7) ensures that the derivative is 0 at $n^\star$ which implies that $n^\star$ is a minimum of this function.

3. M_{C3D} **is individually rational at** $s^\star$**:** This is a direct consequence of step 2 as we can show that an agent 'working on her own' is a valid strategy in M_{C3D}.

4. M_{C3D} **is approximately efficient at** $s^\star$**:** By observing that the global minimum penalty is $2\sigma\sqrt{cm}$, we use a series of nontrivial algebraic manipulations to show $\text{PR}(M_{\text{C3D}}, s^\star) = \frac{1}{2}\left(\frac{10\alpha^2/n_i^\star-1}{4(m+1)\alpha^2/(mn_i^\star)-1}+1\right)$. As $\alpha > \sqrt{n_i^\star}$, some simple algebra leads to $\text{PR}(M_{\text{C3D}}, s^\star) < 2$.

8.4 Special Cases: Restricting the Agents' Strategy Space

In this section, we study two special cases motivated by some natural use cases, where we restrict the agents' strategy space. In addition to providing better guarantees on the efficiency, this will also help us better illustrate the challenges in our original setting.

Agents cannot fabricate or falsify data

First, we study a setting where agents are not allowed to fabricate data or falsify data. Specifically, in (8.2), $\mathcal{F}$ is restricted to functions which map a dataset to any subset. This is applicable when there are regulations preventing such behavior (e.g government institutions, hospitals)

Mechanism: The discussion at the end of Section 8.2 motivates the following modification to the pooling mechanism. We set the allocation space to be $\mathcal{A} = \bigcup_{n\geq 0} \mathbb{R}^n$, i.e the space of all datasets. If an agent i submits at least $\sigma/\sqrt{cm}$ points, then give her all the other agents' datasets, i.e $A_i = \cup_{j\neq i} Y_j$; otherwise, set $A_i = \varnothing$. The recommended strategy $s_i^\star = (n_i^\star, f_i^\star, h_i^\star)$ of each agent is to collect $\sigma/\sqrt{cm}$ points, submit it as is $f_i^\star = \mathbf{I}$, and then use the sample mean of $Z_i \cup A_i$ to estimate μ. The theorem below, whose proof is straightforward, states the main properties of this mechanism.

Theorem 8.4.1. *The following statements about the mechanism and strategy profile $s^\star$ in the paragraph above are true when $\mathcal{F}$ is restricted to functions which map a dataset to any subset: (i) $s^\star$ is a Nash equilibrium. (ii) The mechanism is individually rational at $s^\star$. (iii) At $s^\star$, the mechanism is efficient.*

It is not hard to see that this mechanism can be easily manipulated by the agent if there are no restrictions on $\mathcal{F}$. As the mechanism only checks for the amount of data submitted, the agent can submit a fabricated dataset of $\sigma/\sqrt{cm}$ points, and then discard this dataset when computing the estimate, which results in detrimental free-riding.

Agents accept an estimated value from the mechanism

Our next setting is motivated by use cases where the mechanism may directly deploy the estimated value for μ in some downstream application for the agent, i.e the agents are forced to use this value. This is motivated by federated learning, where agents collect and send data to a server (mechanism), which deploys a model (estimate) directly on the agent's device (Blum et al., 2021; Karimireddy et al., 2022). This requires modifying the agent's strategy space to $\mathcal{S} = \mathbb{N} \times \mathcal{F}$. Now, an agent can only choose (n_i, f_i), how much data she wishes to collect, and how to fabricate or falsify the dataset. A mechanism is defined as a procedure $b : (\bigcup_{n\geq 0} \mathbb{R}^n)^m \to \mathbb{R}^m$, which maps m datasets to m estimated mean values.

Algorithm 19 (see Appendix F.4) outlines a family of mechanisms parametrized by $\epsilon > 0$ for this setting. As we will see shortly, with parameter ϵ, the mechanism

can achieve a PR of $(1+\epsilon)$. This mechanism computes agent i's estimate for μ as follows. First, let Y_{-i} be the union of all datasets submitted by the other agents. Similar to Algorithm 10, the algorithm individually adds Gaussian to each Y_{-i} to obtain Z_i (line 10). Unlike before, this noise is added to the entire dataset and the variance η_i^2 of this noise depends on the difference between the sample means of the agent's submission Y_i and all of the other agents' submissions Y_{-i}. It also depends on two ϵ-dependent parameters defined in line 6. Finally, the mechanism deploys the sample mean of $Y_i \cup Z_i$ as the estimate for μ. The recommended strategies $s_i^\star = (n_i^\star, f_i^\star)$ for the agents is to simply collect $n_i^\star = \sigma/\sqrt{cm}$ points and submit it as is $f_i^\star = \mathbf{I}$. The following theorem states the main properties of the mechanism.

Theorem 8.4.2. *Let $\epsilon > 0$. The following statements about Algorithm 19 and the strategy profile $s^\star$ given in the paragraph above are true: (i) $s^\star$ is a Nash equilibrium. (ii) The mechanism is individually rational at $s^\star$. (iii) At $s^\star$, the mechanism is approximately efficient with $\mathrm{PR}(M, s^\star) \leq 1+\epsilon$.*

The above theorem states that it is possible to obtain a social penalty that is arbitrarily close to the global minimum under the given restriction of the strategy space. However, this mechanism is not NIC if agents are allowed to design their own estimator. For instance, if the mechanism returns $A_i = Z_i$ (line 10), then using a weighted average of the data in X_i and Z_i yields a lower estimation error than simple average used by the mechanism (see Appendix F.4). An agent can leverage this insight to collect and submit less data and obtain a lower overall penalty at the expense of other agents. Cai et al. (2015) study a setting where agents are incentivized to collect data and submit it truthfully via payments. Interestingly, their corruption method can be viewed as a special case of Algorithm 19 with $k_\epsilon = 1$ and only achieves a $1.5\times$ factor of the global minimum social penalty. Moreover, when applied to the more general strategy space, it shares the same shortcomings as the mechanism in Theorem 8.4.2.

8.5 Conclusion

We studied collaborative normal mean estimation in the presence of strategic agents. Naive mechanisms which only look at the quantity of the dataset submitted, can be manipulated by agents who under-collect and/or fabricate data, leaving all agents worse off. To address this issue, when sharing the others' data with an agent, our mechanism M_{C3D} corrupts this dataset proportional to how much the data reported by the agent differs from the other agents. We design minimax optimal estimators for this corrupted dataset to achieve a socially desirable Nash equilibrium.

Future directions: We believe that designing mechanisms for other collaborative learning settings may require relaxing the *exact* NIC guarantees to make the analysis tractable. For many learning problems, it is difficult to design exactly optimal estimators, and it is common to settle for rate-optimal (i.e up to constants) estimators (Lehmann and Casella, 2006). For instance, even simply relaxing to high dimensional distributions with bounded variance, M_{C3D} can only provide an approximate NIC guarantee.

9 FUTURE WORK

This work opens up new research directions along multiple lines.

Robust RL: in Chapter 3 and 4, we provide both upper and lower bounds on the suboptimality gap in the robust online and offline RL settings. However, the upper and low bounds match in neither of these two settings. We believe closing this gap in robust online and offline RL would be an important future research direction.

Byzantine Robust RL: Chapter 5 studies robust mean estimation from heterogeneous batches as a sub-problem while the application of such robust mean estimator goes beyond RL. For example, data batches with dramatically different sizes are common in applications like crowd-sourcing, which requires the generalization of the robust mean estimation to supervised learning settings. Before that, it's necessary to study the mean estimation problem in high-dimensional settings and understand the hardness of the problem. However, the information-theoretic limit of this problem is unknown yet. Because the data batches have different sizes. It's necessary to take the difference into consideration when deriving the information-theoretic lower bound as well as more efficient algorithms. Given an efficient robust mean estimator that works in high dimensions, we study robust supervised learning problems using the framework in Zhu et al. (2023). We leave these to future work.

Perturbation stability of two-player zero-sum games: as shown by Balcan and Braverman (2017), the approximate Nash equilibria of perturbation-stable games can be computed more efficiently, so it would be interesting to apply the perturbation-stability result to improve computation. However, Chapter 7 only studies games with pure base NEs or unique NE. An intermediate next step is to generalize the perturbation stability results to arbitrary two-player zero-sum games or more generally, general-sum games.

Mechanism Design: Chapter 8 studies mechanism design in the most fundamental mean estimation problem, which opens up multiple research directions. Firstly, it remains unclear whether the proposed mechanism is optimal, in the sense of achieving the best possible social penalty. Secondly, in order to apply our mechanism in some real-world scenarios, it's important to study more general learning settings, including: 1. when agents have different unit costs; 2. when agents have access to different distributions; 3. more complicated learning tasks like supervised learning. To study these problems, the first step is to study a proper strategy combination without consideration for NIC or IR. Then we may apply the technique in Chapter 8 to enforce the target strategy.

Ads Data Hub. https://developers.google.com/ads-data-hub/guides/intro. Accessed: 2022-05-10.

PubChem. https://pubchem.ncbi.nlm.nih.gov/. Accessed: 2022-05-10.

53, Data Coordinating Center Burton Robert 67 Jensen Mark A 53 Kahn Ari 53 Pihl Todd 53 Pot David 53 Wan Yunhu, and Tissue Source Site Levine Douglas A 68. 2013. The cancer genome atlas pan-cancer analysis project. *Nature genetics* 45(10): 1113–1120.

Abadi, Martín, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, et al. 2016. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*.

Abbasi-Yadkori, Yasin, Dávid Pál, and Csaba Szepesvári. 2011. Improved algorithms for linear stochastic bandits. In *Nips*, vol. 11, 2312–2320.

Agarwal, Alekh, Mikael Henaff, Sham Kakade, and Wen Sun. 2020a. Pc-pg: Policy cover directed exploration for provable policy gradient learning. *arXiv preprint arXiv:2007.08459*.

Agarwal, Alekh, Sham Kakade, Akshay Krishnamurthy, and Wen Sun. 2020b. Flambe: Structural complexity and representation learning of low rank mdps. *Advances in Neural Information Processing Systems* 33.

Agarwal, Alekh, Sham M Kakade, Jason D Lee, and Gaurav Mahajan. 2019a. On the theory of policy gradient methods: Optimality, approximation, and distribution shift. *arXiv preprint arXiv:1908.00261*.

Agarwal, Anish, Munther Dahleh, Thibaut Horel, and Maryann Rui. 2020c. Towards data auctions with externalities. *arXiv preprint arXiv:2003.08345*.

Agarwal, Anish, Munther Dahleh, and Tuhin Sarkar. 2019b. A marketplace for data: An algorithmic solution. In *Proceedings of the 2019 acm conference on economics and computation*, 701–726.

Agarwal, Mridul, Bhargav Ganguly, and Vaneet Aggarwal. 2021. Communication efficient parallel reinforcement learning. In *Uncertainty in artificial intelligence*, 247–256. PMLR.

Agarwal, Rishabh, Dale Schuurmans, and Mohammad Norouzi. 2020d. An optimistic perspective on offline reinforcement learning. In *International conference on machine learning*, 104–114. PMLR.

Akkaya, Ilge, Marcin Andrychowicz, Maciek Chociej, Mateusz Litwin, Bob McGrew, Arthur Petron, Alex Paino, Matthias Plappert, Glenn Powell, Raphael Ribas, et al. 2019. Solving rubik's cube with a robot hand. *arXiv preprint arXiv:1910.07113*.

Amin, Kareem, Afshin Rostamizadeh, and Umar Syed. 2013. Learning Prices for Repeated Auctions with Strategic Buyers. In *Advances in neural information processing systems*, 1169–1177.

Anscombe, Frank J. 1960. Rejection of outliers. *Technometrics* 2(2):123–146.

Anshelevich, Elliot, Anirban Dasgupta, Jon Kleinberg, Éva Tardos, Tom Wexler, and Tim Roughgarden. 2008. The price of stability for network design with fair cost allocation. *SIAM Journal on Computing* 38(4):1602–1623.

Arnold, Barry C. 2014. Pareto distribution. *Wiley StatsRef: Statistics Reference Online* 1–10.

Athey, Susan, and Ilya Segal. 2013. An Efficient Dynamic Mechanism. *Econometrica* 81(6):2463–2485.

Auer, Peter, Thomas Jaksch, and Ronald Ortner. 2009. Near-optimal regret bounds for reinforcement learning. In *Advances in neural information processing systems*, 89–96.

Awasthi, Pranjal, Maria-Florina Balcan, Avrim Blum, Or Sheffet, and Santosh Vempala. 2010. On nash-equilibria of approximation-stable games. In *Algorithmic game theory: Third international symposium, sagt 2010, athens, greece, october 18-20, 2010. proceedings 3*, 78–89. Springer.

Ayoub, Alex, Zeyu Jia, Csaba Szepesvari, Mengdi Wang, and Lin F Yang. 2020. Model-based reinforcement learning with value-targeted regression. *arXiv preprint arXiv:2006.01107*.

Azar, Mohammad Gheshlaghi, Ian Osband, and Rémi Munos. 2017. Minimax regret bounds for reinforcement learning. In *International conference on machine learning*, 263–272. PMLR.

Bai, Yu, Tengyang Xie, Nan Jiang, and Yu-Xiang Wang. 2019. Provably efficient q-learning with low switching cost. *Advances in Neural Information Processing Systems* 32.

Bakshi, Ainesh, and Adarsh Prasad. 2020. Robust linear regression: Optimal rates in polynomial time. *arXiv preprint arXiv:2007.01394*.

Balcan, Maria-Florina, and Mark Braverman. 2017. Nash equilibria in perturbation-stable games. *Theory of Computing*.

Bazzan, Ana LC. 2009. Opportunities for multiagent systems and multiagent reinforcement learning in traffic control. *Autonomous Agents and Multi-Agent Systems* 18(3):342–375.

Behzadan, Vahid, and Arslan Munir. 2017. Vulnerability of deep reinforcement learning to policy induction attacks. In *Machine learning and data mining in pattern recognition: 13th international conference, mldm 2017, new york, ny, usa, july 15-20, 2017, proceedings 13*, 262–275. Springer.

———. 2019. Adversarial reinforcement learning framework for benchmarking collision avoidance mechanisms in autonomous vehicles. *IEEE Intelligent Transportation Systems Magazine* 13(2):236–241.

Berner, Christopher, Greg Brockman, Brooke Chan, Vicki Cheung, Przemysław Dębiak, Christy Dennison, David Farhi, Quirin Fischer, Shariq Hashme, Chris Hesse, et al. 2019. Dota 2 with large scale deep reinforcement learning. *arXiv preprint arXiv:1912.06680*.

Blum, Avrim, Nika Haghtalab, Richard Lanas Phillips, and Han Shao. 2021. One for one, or all for all: Equilibria and optimality of collaboration in federated learning. In *International conference on machine learning*, 1005–1014. PMLR.

Borak, Szymon, Wolfgang Härdle, and Rafal Weron. 2005. Stable distributions. *Statistical tools for finance and insurance* 1:21–44.

Brown, Tom, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems* 33:1877–1901.

Bubeck, Sébastien, and Nicolo Cesa-Bianchi. 2012. Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *arXiv preprint arXiv:1204.5721*.

Bubeck, Sébastien, Nicolo Cesa-Bianchi, and Gábor Lugosi. 2013. Bandits with heavy tail. *IEEE Transactions on Information Theory* 59(11):7711–7717.

Buckman, Jacob, Carles Gelada, and Marc G Bellemare. 2020. The importance of pessimism in fixed-dataset policy optimization. *arXiv preprint arXiv:2009.06799*.

Burchard, Almut. 2009. A short course on rearrangement inequalities. *Lecture notes, IMDEA Winter School, Madrid*.

Cai, Qi, Zhuoran Yang, Chi Jin, and Zhaoran Wang. 2019. Provably efficient exploration in policy optimization. *arXiv preprint arXiv:1912.05830*.

———. 2020. Provably efficient exploration in policy optimization. In *International conference on machine learning*, 1283–1294. PMLR.

Cai, Yang, Constantinos Daskalakis, and Christos Papadimitriou. 2015. Optimum statistical estimation with strategic data sources. In *Conference on learning theory*, 280–296. PMLR.

Chan, Siu-On, Ilias Diakonikolas, Paul Valiant, and Gregory Valiant. 2014. Optimal algorithms for testing closeness of discrete distributions. In *Proceedings of the twenty-fifth annual acm-siam symposium on discrete algorithms*, 1193–1203. SIAM.

Charikar, Moses, Jacob Steinhardt, and Gregory Valiant. 2017. Learning from untrusted data. In *Proceedings of the 49th annual acm sigact symposium on theory of computing*, 47–60.

Chen, Sitan, Jerry Li, and Ankur Moitra. 2020. Efficiently learning structured distributions from untrusted batches. In *Proceedings of the 52nd annual acm sigact symposium on theory of computing*, 960–973.

Chen, Tianyi, Kaiqing Zhang, Georgios B Giannakis, and Tamer Basar. 2021a. Communication-efficient policy gradient methods for distributed reinforcement learning. *IEEE Transactions on Control of Network Systems*.

Chen, Yiding, Xuezhou Zhang, Kaiqing Zhang, Mengdi Wang, and Xiaojin Zhu. 2022. Byzantine-robust online and offline distributed reinforcement learning. *arXiv preprint arXiv:2206.00165*.

———. 2023. Byzantine-robust online and offline distributed reinforcement learning. In *International conference on artificial intelligence and statistics*, 3230–3269. PMLR.

Chen, Yifang, Simon S Du, and Kevin Jamieson. 2021b. Improved corruption robust algorithms for episodic reinforcement learning. *arXiv preprint arXiv:2102.06875*.

Chen, Yudong, Lili Su, and Jiaming Xu. 2017. Distributed statistical machine learning in adversarial settings: Byzantine gradient descent. *Proceedings of the ACM on Measurement and Analysis of Computing Systems* 1(2):1–25.

Cheung, Wang Chi, David Simchi-Levi, and Ruihao Zhu. 2019. Non-stationary reinforcement learning: The blessing of (more) optimism. *Available at SSRN 3397818*.

Clarke, Edward H. 1971. Multipart Pricing of Public Goods. *Public Choice*.

Cohen, Joel E. 1986. Perturbation theory of completely mixed matrix games. *Linear algebra and its applications* 79:153–162.

Cui, Qiwen, and Simon S Du. 2022. When are offline two-player zero-sum markov games solvable? *Advances in Neural Information Processing Systems* 35:25779–25791.

Dann, Christoph, and Emma Brunskill. 2015. Sample complexity of episodic fixed-horizon reinforcement learning. In *Advances in neural information processing systems*, 2818–2826.

Dann, Christoph, Tor Lattimore, and Emma Brunskill. 2017. Unifying pac and regret: Uniform pac bounds for episodic reinforcement learning. *Advances in Neural Information Processing Systems* 30.

Dann, Christoph, Teodor Vanislavov Marinov, Mehryar Mohri, and Julian Zimmert. 2021. Beyond value-function gaps: Improved instance-dependent regret bounds for episodic reinforcement learning. *Advances in Neural Information Processing Systems* 34:1–12.

Derman, Esther, Daniel Mankowitz, Timothy Mann, and Shie Mannor. 2020. A bayesian approach to robust reinforcement learning. In *Uncertainty in artificial intelligence*, 648–658. PMLR.

Diakonikolas, I, G Kamath, DM Kane, J Li, A Moitra, and A Stewart. 2016. Robust estimators in high dimensions without the computational intractability. In *2016 ieee 57th annual symposium on foundations of computer science (focs)*, 655–664.

Diakonikolas, Ilias, Gautam Kamath, Daniel Kane, Jerry Li, Ankur Moitra, and Alistair Stewart. 2019a. Robust estimators in high-dimensions without the computational intractability. *SIAM Journal on Computing* 48(2):742–864.

Diakonikolas, Ilias, Gautam Kamath, Daniel Kane, Jerry Li, Jacob Steinhardt, and Alistair Stewart. 2019b. Sever: A robust meta-algorithm for stochastic optimization. In *International conference on machine learning*, 1596–1606. PMLR.

Diakonikolas, Ilias, Gautam Kamath, Daniel M Kane, Jerry Li, Ankur Moitra, and Alistair Stewart. 2017. Being robust (in high dimensions) can be practical. In *International conference on machine learning*, 999–1008. PMLR.

Diakonikolas, Ilias, and Daniel M Kane. 2019. Recent advances in algorithmic high-dimensional robust statistics. *arXiv preprint arXiv:1911.05911*.

———. 2023. *Algorithmic high-dimensional robust statistics*. Cambridge university press.

Diakonikolas, Ilias, Daniel M Kane, and Ankit Pensia. 2020. Outlier robust mean estimation with subgaussian rates via stability. *Advances in Neural Information Processing Systems* 33:1830–1840.

Diakonikolas, Ilias, Weihao Kong, and Alistair Stewart. 2019c. Efficient algorithms and lower bounds for robust linear regression. In *Proceedings of the thirtieth annual acm-siam symposium on discrete algorithms*, 2745–2754. SIAM.

Ding, Guohui, Joewie J Koh, Kelly Merckaert, Bram Vanderborght, Marco M Nicotra, Christoffer Heckman, Alessandro Roncone, and Lijun Chen. 2020a. Distributed reinforcement learning for cooperative multi-robot object manipulation. *arXiv preprint arXiv:2003.09540*.

Ding, Ningning, Zhixuan Fang, and Jianwei Huang. 2020b. Incentive mechanism design for federated learning with multi-dimensional private information. In *2020 18th international symposium on modeling and optimization in mobile, ad hoc, and wireless networks (wiopt)*, 1–8. IEEE.

Domingues, Omar Darwiche, Pierre Ménard, Matteo Pirotta, Emilie Kaufmann, and Michal Valko. 2020. A kernel-based approach to non-stationary reinforcement learning in metric spaces. *arXiv preprint arXiv:2007.05078*.

Du, Simon S, Yuping Luo, Ruosong Wang, and Hanrui Zhang. 2019. Provably efficient q-learning with function approximation via distribution shift error checking oracle. In *Advances in neural information processing systems*, 8060–8070.

Dubey, Abhimanyu, and Alex Pentland. 2020. Private and byzantine-proof cooperative decision-making. In *Aamas*, 357–365.

———. 2021. Provably efficient cooperative multi-agent reinforcement learning with function approximation. *arXiv preprint arXiv:2103.04972*.

Dubey, Abhimanyu, et al. 2020. Cooperative multi-agent bandits with heavy tails. In *International conference on machine learning*, 2730–2739. PMLR.

Espeholt, Lasse, Hubert Soyer, Remi Munos, Karen Simonyan, Vlad Mnih, Tom Ward, Yotam Doron, Vlad Firoiu, Tim Harley, Iain Dunning, et al. 2018. Impala: Scalable distributed deep-rl with importance weighted actor-learner architectures. In *International conference on machine learning*, 1407–1416. PMLR.

Even-Dar, Eyal, Sham M Kakade, and Yishay Mansour. 2009. Online markov decision processes. *Mathematics of Operations Research* 34(3):726–736.

Eykholt, Kevin, Ivan Evtimov, Earlence Fernandes, Bo Li, Amir Rahmati, Chaowei Xiao, Atul Prakash, Tadayoshi Kohno, and Dawn Song. 2018. Robust physical-world attacks on deep learning visual classification. In *Proceedings of the ieee conference on computer vision and pattern recognition*, 1625–1634.

Fan, Xiaofeng, Yining Ma, Zhongxiang Dai, Wei Jing, Cheston Tan, and Bryan Kian Hsiang Low. 2021. Fault-tolerant federated reinforcement learning with theoretical guarantee. *Advances in Neural Information Processing Systems* 34.

Flores, Mona, Ittai Dayan, Holger Roth, Aoxiao Zhong, Ahmed Harouni, Amilcare Gentili, Anas Abidin, Andrew Liu, Anthony Costa, Bradford Wood, et al. 2021. Federated learning used for predicting outcomes in sars-cov-2 patients. *Research Square*.

Fraboni, Yann, Richard Vidal, and Marco Lorenzi. 2021. Free-rider attacks on model aggregation in federated learning. In *International conference on artificial intelligence and statistics*, 1846–1854. PMLR.

Freedman, David A. 1975. On tail probabilities for martingales. *the Annals of Probability* 100–118.

Fujimoto, Scott, David Meger, and Doina Precup. 2019. Off-policy deep reinforcement learning without exploration. In *International conference on machine learning*, 2052–2062. PMLR.

Gao, Minbo, Tianle Xie, Simon S Du, and Lin F Yang. 2021. A provably efficient algorithm for linear markov decision process with low switching cost. *arXiv preprint arXiv:2101.00494*.

Goldman, Alan J, and Albert W Tucker. 2016. 4. theory of linear programming. In *Linear inequalities and related systems.(am-38), volume 38*, 53–98. Princeton University Press.

Goyal, Priya, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. 2017. Accurate, large minibatch sgd: Training imagenet in 1 hour. *arXiv preprint arXiv:1706.02677*.

Groves, Theodore. 1979. Efficient Collective Choice when Compensation is Possible. *The Review of Economic Studies*.

Gupta, Anupam, Tomer Koren, and Kunal Talwar. 2019. Better algorithms for stochastic bandits with adversarial corruptions. *arXiv preprint arXiv:1902.08647*.

HOPKINS, SAMUEL B. 2020. Mean estimation with sub-gaussian rates in polynomial time. *The Annals of Statistics* 48(2):1193–1213.

Horgan, Dan, John Quan, David Budden, Gabriel Barth-Maron, Matteo Hessel, Hado Van Hasselt, and David Silver. 2018. Distributed prioritized experience replay. *arXiv preprint arXiv:1803.00933*.

Hu, Yichun, Nathan Kallus, and Masatoshi Uehara. 2021. Fast rates for the regret of offline reinforcement learning. *arXiv preprint arXiv:2102.00479*.

Huang, Sandy, Nicolas Papernot, Ian Goodfellow, Yan Duan, and Pieter Abbeel. 2017. Adversarial attacks on neural network policies. *arXiv preprint arXiv:1702.02284*.

Huber, Peter J. 1992. Robust estimation of a location parameter. In *Breakthroughs in statistics*, 492–518. Springer.

Huber, Peter J, et al. 1967. The behavior of maximum likelihood estimates under nonstandard conditions. In *Proceedings of the fifth berkeley symposium on mathematical statistics and probability*, vol. 1, 221–233. University of California Press.

Jadbabaie, Ali, Haochuan Li, Jian Qian, and Yi Tian. 2022. Byzantine-robust federated linear bandits. *arXiv preprint arXiv:2204.01155*.

Jain, Ayush, and Alon Orlitsky. 2021. Robust density estimation from batches: The best things in life are (nearly) free. In *International conference on machine learning*, 4698–4708. PMLR.

Jia, Ruoxi, David Dao, Boxin Wang, Frances Ann Hubis, Nick Hynes, Nezihe Merve Gürel, Bo Li, Ce Zhang, Dawn Song, and Costas J Spanos. 2019. Towards efficient data valuation based on the shapley value. In *The 22nd international conference on artificial intelligence and statistics*, 1167–1176. PMLR.

Jiang, Nan. 2020. Notes on tabular methods.

Jiang, Nan, Akshay Krishnamurthy, Alekh Agarwal, John Langford, and Robert E Schapire. 2017. Contextual decision processes with low bellman rank are pac-learnable. In *International conference on machine learning*, 1704–1713. PMLR.

Jin, Chi, Zeyuan Allen-Zhu, Sebastien Bubeck, and Michael I Jordan. 2018. Is q-learning provably efficient? *Advances in neural information processing systems* 31.

Jin, Chi, Tiancheng Jin, Haipeng Luo, Suvrit Sra, and Tiancheng Yu. 2020a. Learning adversarial markov decision processes with bandit feedback and unknown transition. In *International conference on machine learning*, 4860–4869. PMLR.

Jin, Chi, Praneeth Netrapalli, Rong Ge, Sham M Kakade, and Michael I Jordan. 2019. A short note on concentration inequalities for random vectors with subgaussian norm. *arXiv preprint arXiv:1902.03736*.

Jin, Chi, Zhuoran Yang, Zhaoran Wang, and Michael I Jordan. 2020b. Provably efficient reinforcement learning with linear function approximation. In *Conference on learning theory*, 2137–2143. PMLR.

Jin, Tiancheng, and Haipeng Luo. 2020. Simultaneously learning stochastic and adversarial episodic mdps with known transition. *arXiv preprint arXiv:2006.05606*.

Jin, Ying, Zhuoran Yang, and Zhaoran Wang. 2020c. Is pessimism provably efficient for offline rl? *arXiv preprint arXiv:2012.15085*.

———. 2021. Is pessimism provably efficient for offline rl? In *International conference on machine learning*, 5084–5096. PMLR.

Jones, Charles I, and Christopher Tonetti. 2020. Nonrivalry and the economics of data. *American Economic Review* 110(9):2819–58.

Jonsson, Anders, Emilie Kaufmann, Pierre Ménard, Omar Darwiche Domingues, Edouard Leurent, and Michal Valko. 2020. Planning in markov decision processes with gap-dependent sample complexity. *Advances in Neural Information Processing Systems* 33:1253–1263.

Kairouz, Peter, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. 2021. Advances and open problems in federated learning. *Foundations and Trends® in Machine Learning* 14(1–2):1–210.

Kakade, Sham, Akshay Krishnamurthy, Kendall Lowrey, Motoya Ohnishi, and Wen Sun. 2020. Information theoretic regret bounds for online nonlinear control. *Advances in Neural Information Processing Systems* 33.

Kakade, Sham, and John Langford. 2002. Approximately optimal approximate reinforcement learning. In *Icml*, vol. 2, 267–274.

Kakade, Sham M. 2001. A natural policy gradient. *Advances in neural information processing systems* 14:1531–1538.

Kakade, Sham M, Ilan Lobel, and Hamid Nazerzadeh. 2010. An Optimal Dynamic Mechanism for Multi-armed Bandit Processes. *arXiv preprint arXiv:1001.4598*.

Kapoor, Sayash, Kumar Kshitij Patel, and Purushottam Kar. 2019. Corruption-tolerant bandit learning. *Machine Learning* 108(4):687–715.

Karimireddy, Sai Praneeth, Wenshuo Guo, and Michael I Jordan. 2022. Mechanisms that incentivize data sharing in federated learning. *arXiv preprint arXiv:2207.04557*.

Kidambi, Rahul, Aravind Rajeswaran, Praneeth Netrapalli, and Thorsten Joachims. 2020. Morel: Model-based offline reinforcement learning. *arXiv preprint arXiv:2005.05951*.

Kimura, Yutaka, Yoichi Sawasaki, and Kensuke Tanaka. 2000. A perturbation on two-person zero-sum games. In *Advances in dynamic games and applications*, 279–288. Birkhäuser Boston.

Klivans, Adam, Pravesh K Kothari, and Raghu Meka. 2018. Efficient algorithms for outlier-robust regression. In *Conference on learning theory*, 1420–1430. PMLR.

Kreps, David M. 1990. *Game theory and economic modelling*. Oxford University Press.

Kretchmar, R Matthew. 2002. Parallel reinforcement learning. In *The 6th world conference on systemics, cybernetics, and informatics*. Citeseer.

Kumar, Aviral, Justin Fu, George Tucker, and Sergey Levine. 2019. Stabilizing off-policy q-learning via bootstrapping error reduction. *arXiv preprint arXiv:1906.00949*.

Kumar, Aviral, Aurick Zhou, George Tucker, and Sergey Levine. 2020. Conservative q-learning for offline reinforcement learning. *arXiv preprint arXiv:2006.04779*.

Lai, Kevin A, Anup B Rao, and Santosh Vempala. 2016. Agnostic estimation of mean and covariance. In *2016 ieee 57th annual symposium on foundations of computer science (focs)*, 665–674. IEEE.

LAMPORT, LESLIE, ROBERT SHOSTAK, and MARSHALL PEASE. 1982. The byzantine generals problem. *ACM Transactions on Programming Languages and Systems* 4(3):382–401.

Lange, Sascha, Thomas Gabel, and Martin Riedmiller. 2012. Batch reinforcement learning. In *Reinforcement learning*, 45–73. Springer.

Laroche, Romain, Paul Trichelair, and Remi Tachet Des Combes. 2019. Safe policy improvement with baseline bootstrapping. In *International conference on machine learning*, 3652–3661. PMLR.

Lattimore, Tor, and Csaba Szepesvári. 2020. *Bandit algorithms*. Cambridge University Press.

Lee, Chung-Wei, Haipeng Luo, Chen-Yu Wei, and Mengxiao Zhang. 2020. Bias no more: high-probability data-dependent regret bounds for adversarial bandits and mdps. *Advances in Neural Information Processing Systems* 33.

Lehmann, Erich L, and George Casella. 2006. *Theory of point estimation*. Springer Science & Business Media.

Leng, Mingming, and Mahmut Parlar. 2005. Game theoretic applications in supply chain management: a review. *INFOR: Information Systems and Operational Research* 43(3):187–220.

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

Liebeherr, Jörg, Almut Burchard, and Florin Ciucu. 2012. Delay bounds in communication networks with heavy-tailed and self-similar traffic. *IEEE Transactions on Information Theory* 58(2):1010–1024.

Lin, Jierui, Min Du, and Jian Liu. 2019. Free-riders in federated learning: Attacks and defenses. *arXiv preprint arXiv:1911.12560*.

Lipton, Richard J, and Aranyak Mehta. 2006. On stability properties of economic solution concepts.

Liu, Ximeng, Robert H Deng, Kim-Kwang Raymond Choo, and Yang Yang. 2019. Privacy-preserving reinforcement learning design for patient-centric dynamic treatment regimes. *IEEE Transactions on Emerging Topics in Computing* 9(1):456–470.

Liu, Yao, Adith Swaminathan, Alekh Agarwal, and Emma Brunskill. 2020. Provably good batch reinforcement learning without great exploration. *arXiv preprint arXiv:2007.08202*.

Liu, Yuan, Mengmeng Tian, Yuxin Chen, Zehui Xiong, Cyril Leung, and Chunyan Miao. 2022. A contract theory based incentive mechanism for federated learning. In *Federated and transfer learning*, 117–137. Springer.

Lugosi, Gábor, and Shahar Mendelson. 2019a. Mean estimation and regression under heavy-tailed distributions: A survey. *Foundations of Computational Mathematics* 19(5):1145–1190.

———. 2019b. Sub-gaussian estimators of the mean of a random vector. *The Annals of Statistics* 47(2):783–794.

Lugosi, Gabor, and Shahar Mendelson. 2021. Robust multivariate mean estimation: the optimality of trimmed mean. *The Annals of Statistics* 49(1):393–410.

Lykouris, Thodoris, Vahab Mirrokni, and Renato Paes Leme. 2018. Stochastic bandits robust to adversarial corruptions. In *Proceedings of the 50th annual acm sigact symposium on theory of computing*, 114–122.

Lykouris, Thodoris, Max Simchowitz, Aleksandrs Slivkins, and Wen Sun. 2019. Corruption robust exploration in episodic reinforcement learning. *arXiv preprint arXiv:1911.08689*.

Lykouris, Thodoris, Max Simchowitz, Alex Slivkins, and Wen Sun. 2021. Corruption-robust exploration in episodic reinforcement learning. In *Conference on learning theory*, 3242–3245. PMLR.

Ma, Yuzhe, J Sharp, Ruizhe Wang, Earlence Fernandes, and Xiaojin Zhu. 2021. Adversarial attacks on kalman filter-based forward collision warning systems. In *The thirty-fifth aaai conference on artificial intelligence*.

Ma, Yuzhe, Xuezhou Zhang, Wen Sun, and Jerry Zhu. 2019. Policy poisoning in batch reinforcement learning and control. *Advances in Neural Information Processing Systems* 32.

Mansour, Yishay, Aleksandrs Slivkins, and Vasilis Syrgkanis. 2015. Bayesian Incentive-compatible Bandit Exploration. In *Proceedings of the sixteenth acm conference on economics and computation*, 565–582.

Medina, Andres Munoz, and Scott Yang. 2016. No-regret algorithms for heavy-tailed linear bandits. In *International conference on machine learning*, 1642–1650. PMLR.

Miller, Nolan, Paul Resnick, and Richard Zeckhauser. 2005. Eliciting informative feedback: The peer-prediction method. *Management Science* 51(9):1359–1373.

Nash Jr, John. 1996. Non-cooperative games. In *Essays on game theory*, 22–33. Edward Elgar Publishing.

Nazerzadeh, Hamid, Amin Saberi, and Rakesh Vohra. 2008. Dynamic Cost-per-action Mechanisms and Applications to Online Advertising. In *Proceedings of the 17th international conference on world wide web*, 179–188.

Neff, Gina. 2016. Talking to bots: Symbiotic agency and the case of tay. *International Journal of Communication*.

Neff, Gina, and Peter Nagy. 2016. Automation, algorithms, and politics| talking to bots: Symbiotic agency and the case of tay. *International Journal of Communication* 10:17.

Neu, Gergely, András György, and Csaba Szepesvári. 2010. The online loop-free stochastic shortest-path problem. In *Colt*, vol. 2010, 231–243. Citeseer.

Neu, Gergely, Andras Gyorgy, and Csaba Szepesvári. 2012. The adversarial stochastic shortest path problem with unknown transition probabilities. In *Artificial intelligence and statistics*, 805–813.

Neu, Gergely, and Julia Olkhovskaya. 2020. Online learning in mdps with linear function approximation and bandit feedback. *arXiv preprint arXiv:2007.01612*.

Nisan, Noam, Tim Roughgarden, Eva Tardos, and Vijay V Vazirani. 2007. *Algorithmic game theory*. Cambridge university press.

Niss, Laura, and Ambuj Tewari. 2020. What you see may not be what you get: Ucb bandit algorithms robust to ε-contamination. In *Conference on uncertainty in artificial intelligence*, 450–459. PMLR.

Ornik, Melkior, and Ufuk Topcu. 2019. Learning and planning for time-varying mdps using maximum likelihood estimation. *arXiv preprint arXiv:1911.12976*.

Ortner, Ronald, Pratik Gajane, and Peter Auer. 2019. Variational regret bounds for reinforcement learning. In *Uai*, 16.

Osband, Ian, and Benjamin Van Roy. 2014. Model-based reinforcement learning and the eluder dimension. *Advances in Neural Information Processing Systems* 27: 1466–1474.

———. 2016. On lower bounds for regret in reinforcement learning. *arXiv preprint arXiv:1608.02732*.

Panaganti, Kishan, Zaiyan Xu, Dileep Kalathil, and Mohammad Ghavamzadeh. 2022. Robust reinforcement learning using offline data. *arXiv preprint arXiv:2208.05129*.

Paninski, Liam. 2008. A coincidence-based test for uniformity given very sparsely sampled discrete data. *IEEE Transactions on Information Theory* 54(10):4750–4755.

Pensia, Ankit, Varun Jog, and Po-Ling Loh. 2020. Robust regression with covariate filtering: Heavy tails and adversarial contamination. *arXiv preprint arXiv:2009.12976*.

Petersen, Ian R, Valery A Ugrinovskii, and Andrey V Savkin. 2012. *Robust control design using h-∞ methods*. Springer Science & Business Media.

Pinto, Lerrel, James Davidson, Rahul Sukthankar, and Abhinav Gupta. 2017. Robust adversarial reinforcement learning. In *International conference on machine learning*, 2817–2826. PMLR.

Prasad, Adarsh, Arun Sai Suggala, Sivaraman Balakrishnan, and Pradeep Ravikumar. 2018. Robust estimation via robust gradient estimation. *arXiv preprint arXiv:1802.06485*.

———. 2020. Robust estimation via robust gradient estimation. *Journal of the Royal Statistical Society Series B: Statistical Methodology* 82(3):601–627.

Procaccia, Ariel D. 2013. Cake Cutting: Not just Child's Play. *Communications of the ACM* 56(7):78–87.

Qiao, Mingda, and Gregory Valiant. 2017. Learning discrete distributions from untrusted batches. *arXiv preprint arXiv:1711.08113*.

Rashidinejad, Paria, Banghua Zhu, Cong Ma, Jiantao Jiao, and Stuart Russell. 2021. Bridging offline reinforcement learning and imitation learning: A tale of pessimism. *Advances in Neural Information Processing Systems* 34.

Rosenberg, Aviv, and Yishay Mansour. 2019. Online convex optimization in adversarial markov decision processes. *arXiv preprint arXiv:1905.07773*.

Roth, Alvin E. 1986. On the Allocation of Residents to Rural Hospitals: A General Property of Two-sided Matching Markets. *Econometrica: Journal of the Econometric Society* 425–427.

Roth, Alvin E, Tayfun Sönmez, and M Utku Ünver. 2004. Kidney Exchange. *The Quarterly journal of economics* 119(2):457–488.

Roy, Sankardas, Charles Ellis, Sajjan Shiva, Dipankar Dasgupta, Vivek Shandilya, and Qishi Wu. 2010. A survey of game theory as applied to network security. In *2010 43rd hawaii international conference on system sciences*, 1–10. IEEE.

Sakuma, Jun, Shigenobu Kobayashi, and Rebecca N Wright. 2008. Privacy-preserving reinforcement learning. In *Proceedings of the 25th international conference on machine learning*, 864–871.

Schulman, John, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. 2015a. Trust region policy optimization. In *International conference on machine learning*, 1889–1897.

Schulman, John, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. 2015b. High-dimensional continuous control using generalized advantage estimation. *arXiv preprint arXiv:1506.02438*.

Schulman, John, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.

Shalev-Shwartz, Shai, et al. 2011. Online learning and online convex optimization. *Foundations and trends in Machine Learning* 4(2):107–194.

Shao, Han, Xiaotian Yu, Irwin King, and Michael R Lyu. 2018. Almost optimal algorithms for linear stochastic bandits with heavy-tailed payoffs. *Advances in Neural Information Processing Systems* 31.

Sheller, Micah J, G Anthony Reina, Brandon Edwards, Jason Martin, and Spyridon Bakas. 2019. Multi-institutional deep learning modeling without sharing patient data: A feasibility study on brain tumor segmentation. In *Brainlesion: Glioma, multiple sclerosis, stroke and traumatic brain injuries: 4th international workshop, brainles 2018, held in conjunction with miccai 2018, granada, spain, september 16, 2018, revised selected papers, part i 4*, 92–104. Springer.

Shi, Laixi, and Yuejie Chi. 2022. Distributionally robust model-based offline reinforcement learning with near-optimal sample complexity. *arXiv preprint arXiv:2208.05767*.

Siegel, Noah Y, Jost Tobias Springenberg, Felix Berkenkamp, Abbas Abdolmaleki, Michael Neunert, Thomas Lampe, Roland Hafner, Nicolas Heess, and Martin Riedmiller. 2020. Keep doing what worked: Behavioral modelling priors for offline reinforcement learning. *arXiv preprint arXiv:2002.08396*.

Sim, Rachael Hwee Ling, Yehong Zhang, Mun Choon Chan, and Bryan Kian Hsiang Low. 2020. Collaborative machine learning with incentive-aware model rewards. In *International conference on machine learning*, 8927–8936. PMLR.

Simchowitz, Max, and Kevin G Jamieson. 2019. Non-asymptotic gap-dependent regret bounds for tabular mdps. *Advances in Neural Information Processing Systems* 32.

Sun, Jianwen, Tianwei Zhang, Xiaofei Xie, Lei Ma, Yan Zheng, Kangjie Chen, and Yang Liu. 2020. Stealthy and efficient adversarial attacks against deep reinforcement learning. In *Proceedings of the aaai conference on artificial intelligence*, vol. 34, 5883–5891.

Sun, Wen, Nan Jiang, Akshay Krishnamurthy, Alekh Agarwal, and John Langford. 2019. Model-based rl in contextual decision processes: Pac bounds and exponential improvements over model-free approaches. In *Conference on learning theory*, 2898–2933. PMLR.

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

Sutton, Richard S, David A McAllester, Satinder P Singh, and Yishay Mansour. 1999. Policy gradient methods for reinforcement learning with function approximation. In *Advances in neural information processing systems*, vol. 99, 1057–1063.

Todorov, Emanuel, Tom Erez, and Yuval Tassa. 2012. Mujoco: A physics engine for model-based control. In *2012 ieee/rsj international conference on intelligent robots and systems*, 5026–5033. IEEE.

Tropp, Joel A. 2015. An introduction to matrix concentration inequalities. *arXiv preprint arXiv:1501.01571*.

Troutt, Marvin D. 1986. A stability concept for matrix game optimal strategies and its application to linear programming sensitivity analysis. *Mathematical programming* 36:353–361.

———. 1990. An eigenvalue formula for the radius of stability of a stable game matrix. *SIAM journal on matrix analysis and applications* 11(3):369–372.

Tukey, John W. 1960. A survey of sampling from contaminated distributions. *Contributions to probability and statistics* 448–485.

Verbraeken, Joost, Matthijs Wolting, Jonathan Katzy, Jeroen Kloppenburg, Tim Verbelen, and Jan S Rellermeyer. 2020. A survey on distributed machine learning. *ACM Computing Surveys (CSUR)* 53(2):1–33.

Vickrey, William. 1961. Counterspeculation, Auctions, and Competitive Sealed Tenders. *The Journal of Finance*.

Wagenmaker, Andrew J, Max Simchowitz, and Kevin Jamieson. 2022. Beyond no regret: Instance-dependent pac reinforcement learning. In *Conference on learning theory*, 358–418. PMLR.

Wald, Abraham. 1939. Contributions to the theory of statistical estimation and testing hypotheses. *The Annals of Mathematical Statistics* 10(4):299–326.

Wang, Ruosong, Dean P Foster, and Sham M Kakade. 2020a. What are the statistical limits of offline rl with linear function approximation? *arXiv preprint arXiv:2010.11895*.

Wang, Tianhao, Johannes Rausch, Ce Zhang, Ruoxi Jia, and Dawn Song. 2020b. A principled approach to data valuation for federated learning. *Federated Learning: Privacy and Incentive* 153–167.

Wang, Xinqi, Qiwen Cui, and Simon S Du. 2022. On gap-dependent bounds for offline reinforcement learning. *Advances in Neural Information Processing Systems* 35:14865–14877.

Wei, Chen-Yu, Christoph Dann, and Julian Zimmert. 2022. A model selection approach for corruption robust reinforcement learning. In *International conference on algorithmic learning theory*, 1043–1096. PMLR.

Williams, Ronald J. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning* 8(3-4):229–256.

Wu, Yifan, George Tucker, and Ofir Nachum. 2019. Behavior regularized offline reinforcement learning. *arXiv preprint arXiv:1911.11361*.

Xie, Qiaomin, Yudong Chen, Zhaoran Wang, and Zhuoran Yang. 2020. Learning zero-sum simultaneous-move markov games using function approximation and correlated equilibrium. In *Conference on learning theory*, 3674–3682. PMLR.

Xie, Tengyang, Nan Jiang, Huan Wang, Caiming Xiong, and Yu Bai. 2021. Policy finetuning: Bridging sample-efficient offline and online reinforcement learning. *Advances in neural information processing systems* 34:27395–27407.

Xu, Haike, Tengyu Ma, and Simon Du. 2021a. Fine-grained gap-dependent bounds for tabular mdps via adaptive multi-step bootstrap. In *Conference on learning theory*, 4438–4472. PMLR.

Xu, Xinyi, Lingjuan Lyu, Xingjun Ma, Chenglin Miao, Chuan Sheng Foo, and Bryan Kian Hsiang Low. 2021b. Gradient driven rewards to guarantee fairness in collaborative machine learning. *Advances in Neural Information Processing Systems* 34:16104–16117.

Yadkori, Yasin Abbasi, Peter L Bartlett, Varun Kanade, Yevgeny Seldin, and Csaba Szepesvári. 2013. Online learning in markov decision processes with adversarially chosen transition probability distributions. In *Advances in neural information processing systems*, 2508–2516.

Yang, Lin, and Mengdi Wang. 2019a. Sample-optimal parametric q-learning using linearly additive features. In *International conference on machine learning*, 6995–7004. PMLR.

———. 2020. Reinforcement learning in feature space: Matrix bandit, kernels, and regret bound. In *International conference on machine learning*, 10746–10756. PMLR.

Yang, Lin F, and Mengdi Wang. 2019b. Reinforcement learning in feature space: Matrix bandit, kernels, and regret bound. *arXiv preprint arXiv:1905.10389*.

Yang, Mengjiao, and Ofir Nachum. 2021. Representation matters: Offline pretraining for sequential decision making. *arXiv preprint arXiv:2102.05815*.

Yin, Dong, Yudong Chen, Ramchandran Kannan, and Peter Bartlett. 2018. Byzantine-robust distributed learning: Towards optimal statistical rates. In *International conference on machine learning*, 5650–5659. PMLR.

Yin, Ming, Yu Bai, and Yu-Xiang Wang. 2020. Near optimal provable uniform convergence in off-policy evaluation for reinforcement learning. *arXiv preprint arXiv:2007.03760*.

———. 2021. Near-optimal offline reinforcement learning via double variance reduction. *arXiv preprint arXiv:2102.01748*.

Yu, Tao, HZ Wang, Bin Zhou, Ka Wing Chan, and J Tang. 2014. Multi-agent correlated equilibrium q (λ) learning for coordinated smart generation control of interconnected power grids. *IEEE transactions on power systems* 30(4):1669–1679.

Yu, Tianhe, Aviral Kumar, Rafael Rafailov, Aravind Rajeswaran, Sergey Levine, and Chelsea Finn. 2021. Combo: Conservative offline model-based policy optimization. *arXiv preprint arXiv:2102.08363*.

Yu, Tianhe, Garrett Thomas, Lantao Yu, Stefano Ermon, James Zou, Sergey Levine, Chelsea Finn, and Tengyu Ma. 2020. Mopo: Model-based offline policy optimization. *arXiv preprint arXiv:2005.13239*.

Yu, Xiaotian, Han Shao, Michael R Lyu, and Irwin King. 2018. Pure exploration of multi-armed bandits with heavy-tailed payoffs. In *Uai*, 937–946.

Zanette, Andrea, David Brandfonbrener, Emma Brunskill, Matteo Pirotta, and Alessandro Lazaric. 2020. Frequentist regret bounds for randomized least-squares value iteration. In *International conference on artificial intelligence and statistics*, 1954–1964.

Zanette, Andrea, Ching-An Cheng, and Alekh Agarwal. 2021. Cautiously optimistic policy optimization and exploration with linear function approximation. *arXiv preprint arXiv:2103.12923*.

Zhang, Xuezhou, Yiding Chen, Jerry Zhu, and Wen Sun. 2021a. Corruption-robust offline reinforcement learning. *arXiv preprint arXiv:2106.06630*.

Zhang, Xuezhou, Yiding Chen, Xiaojin Zhu, and Wen Sun. 2021b. Robust policy gradient against strong data corruption. In *International conference on machine learning*, 12391–12401. PMLR.

———. 2022. Corruption-robust offline reinforcement learning. In *International conference on artificial intelligence and statistics*, 5757–5773. PMLR.

Zhang, Xuezhou, Yuzhe Ma, Adish Singla, and Xiaojin Zhu. 2020a. Adaptive reward-poisoning attacks against reinforcement learning. In *International conference on machine learning*, 11225–11234. PMLR.

Zhang, Zihan, Yuan Zhou, and Xiangyang Ji. 2020b. Almost optimal model-free reinforcement learningvia reference-advantage decomposition. *Advances in Neural Information Processing Systems* 33:15198–15207.

Zheng, Wenting, Raluca Ada Popa, Joseph E Gonzalez, and Ion Stoica. 2019. Helen: Maliciously secure coopetitive learning for linear models. In *2019 ieee symposium on security and privacy (sp)*, 724–738. IEEE.

Zhou, Dongruo, Jiafan He, and Quanquan Gu. 2020. Provably efficient reinforcement learning for discounted mdps with feature mapping. *arXiv preprint arXiv:2006.13165*.

Zhou, Kemin, and John Comstock Doyle. 1998. *Essentials of robust control*, vol. 104. Prentice hall Upper Saddle River, NJ.

Zhu, Banghua, Lun Wang, Qi Pang, Shuai Wang, Jiantao Jiao, Dawn Song, and Michael I Jordan. 2023. Byzantine-robust federated learning with optimal statistical rates. In *International conference on artificial intelligence and statistics*, 3151–3178. PMLR.

Zimin, Alexander, and Gergely Neu. 2013. Online learning in episodic markovian decision processes by relative entropy policy search. In *Advances in neural information processing systems*, 1583–1591.

A APPENDIX FOR CHAPTER 3

A.1 Additional Related Work

RL in standard MDPs. Learning MDPs with stochastic rewards and transitions is relatively well-studied for the tabular case (that is, a finite number of states and actions). For example, in the episodic setting, the UCRL2 algorithm Auer et al. (2009) achieves $O(\sqrt{H^4S^2AT})$ regret, where H is the episode length, S is the state space size, A is the action space size, and T is the total number of steps. Later the UCBVI algorithm Azar et al. (2017); Dann et al. (2017) achieves the optimal $O(\sqrt{H^2SAT})$ regret matching the lower-bound Osband and Van Roy (2016); Dann and Brunskill (2015). Recent work extends the analysis to various linear setting Jin et al. (2020b); Yang and Wang (2019b,a); Zanette et al. (2020); Ayoub et al. (2020); Zhou et al. (2020); Cai et al. (2019); Du et al. (2019); Kakade et al. (2020) with known linear feature. For unknown feature, Agarwal et al. (2020b) proposes a sample efficient algorithm that explicitly learns feature representation under the assumption that the transition matrix is low rank. Beyond the linear settings, there are works assuming the function class has low Eluder dimension which so far is known to be small only for linear functions and generalized linear models Osband and Van Roy (2014). For more general function approximation, Jiang et al. (2017); Sun et al. (2019) showed that polynomial sample complexity is achievable as long as the MDP and the given function class together induce low Bellman rank and Witness rank, which include almost all prior models such as tabular MDP, linear MDPs Yang and Wang (2019b); Jin et al. (2020b), Kernelized nonlinear regulators Kakade et al. (2020), low rank MDP Agarwal et al. (2020b), and Bellman completion under linear functions Zanette et al. (2020).

A.2 Proof for lower bound result

Theorem A.2.1 (Theorem 3.3.1)**.** *For any algorithm, there exists an MDP such that the algorithm fails to find an* $\left(\frac{\epsilon}{2(1-\gamma)}\right)$*-optimal policy under the* ϵ*-contamination model with a probability of at least* $1/4$*.*

Proof of Theorem A.2.1. Consider two MDPs M_1, M_2, both with 3 states and 2 actions, defined as

$$P_1(s_2|s_1,a_1) = \frac{1-\epsilon}{2}, P_1(s_3|s_1,a_1) = \frac{1+\epsilon}{2}, P_1(s_3|s_1,a_2) = P_1(s_3|s_1,a_2) = \frac{1}{2} \quad \text{(A.1)}$$

$$P_2(s_2|s_1,a_1) = \frac{1+\epsilon}{2}, P_2(s_3|s_1,a_1) = \frac{1-\epsilon}{2}, P_2(s_3|s_1,a_2) = P_2(s_3|s_1,a_2) = \frac{1}{2} \quad \text{(A.2)}$$

and for both MDPs s_2, s_3 are absorbing states with constant reward 1 and 0, respectively. So for M_1, the optimal policy is $\pi_1^*(s_1) = a_2$, and for M_2, the optimal policy is $\pi_2^*(s_1) = a_1$. In both cases, choosing the alternative action in s_1 will incur a suboptimality gap of $\frac{\epsilon}{2(1-\gamma)}$.

Let $N(\cdot)$ be the probability function of Bernoulli distribution on $\{s_2, s_3\}$: $N(x) = \begin{cases} 1 & \text{if } x = s_2 \\ 0 & \text{if } x = s_3 \end{cases}$. First of all, notice that an 2ϵ*-oblivious adversary* can make the two MDPs M_1, M_2 indistinguishable by changing $P_1(\cdot \mid s_1, a_1)$ to be $(1 - \frac{2\epsilon}{1+\epsilon})P_1(\cdot \mid s_1, a_1) + \frac{2\epsilon}{1+\epsilon}N(\cdot)$, which is exactly $P_2(\cdot \mid s_1, a_1)$. Note that $\frac{2\epsilon}{1+\epsilon} \leq 2\epsilon$ and thus can be achieved by a 2ϵ-oblivious adversary.

When the two MDPs are indistinguishable, any rollout has the same probability under both MDP, and thus conditioned on any roll-out, the learner can at best obtain an $\frac{\epsilon}{2(1-\gamma)}$-optimal policy with probability $1/2$ on both MDP.

What remains to be shown is that with high probability, the ϵ-contamination adversary can simulate the oblivious adversary.

Let X_i, Y_i be Bernoulli random variables s.t.

$$X_i = \begin{cases} s_2 & U \leq \frac{1-\epsilon}{2} \\ s_3 & \text{o.w.} \end{cases}, \quad Y_i = \begin{cases} s_2 & U \leq \frac{1+\epsilon}{2} \\ s_3 & \text{o.w.} \end{cases}$$

, where U is picked uniformly random in $[0, 1]$. Then (X_i, Y_i) is a coupling with law: $P((X_i, Y_i) = (s_2, s_2)) = \frac{1-\epsilon}{2}$, $P((X_i, Y_i) = (s_2, s_3)) = 0$, $P((X_i, Y_i) = (s_3, s_2)) = \epsilon$, $P((X_i, Y_i) = (s_3, s_3)) = \frac{1-\epsilon}{2}$, X_i and Y_i can be thought as the outcome of $P_1(\cdot \mid s_1, a_1)$, $P_2(\cdot \mid s_1, a_1)$ respectively. The ϵ-contamination adversary can simulate the oblivious adversary by changing X_i to Y_i when $X_1 \neq Y_i$, which has probability ϵ. This is possible when there are at most ϵ fraction of index i s.t. $X_i \neq Y_i$. Suppose there are T episodes, then

$$P\left(\sum_{i=1}^{T} \mathbb{1}_{\{a_1 \text{ is taken at } s_1\}} \mathbb{1}_{\{X_i \neq Y_i\}} \geq \epsilon T\right) \leq P(\sum_{i=1}^{T} \mathbb{1}_{\{X_i \neq Y_i\}} \geq T\epsilon) \leq \frac{1}{2} \tag{A.3}$$

because the median of Binomial(n, p) is at most $\lceil np \rceil$. Therefore, the probability that the adaptive adversary can simulate the oblivious adversary throughout T episodes is at least $1/2$. Assuming that when the adversary fails to simulate, the learner automatically succeed in finding the optimal policy, then we've established that the learner will still fail to find an $\left(\frac{\epsilon}{2(1-\gamma)}\right)$-optimal policy with probability $1/4$ on both MDPs. ■

A.3 Property of $\hat{Q}(s, a)$ sampled from Algorithm 1

To prepare for the analysis that follows, we first show that the $\hat{Q}(s, a)$ sampled from Algorithm 1 is unbiased and has bounded variance.

Lemma A.3.1. $\mathbb{E}\left[\hat{Q}^\pi(s, a)\right] = Q^\pi(s, a)$, *Var*$(\hat{Q}^\pi(s, a)) \leq \frac{\gamma}{(1-\gamma)^2} + \frac{\sigma^2}{1-\gamma}$. *The bound for variance is tight.*

Proof of Lemma A.3.1. In the following, we treat (s_0, a_0) as deterministic.

$$\begin{aligned}
\mathbb{E}\left[\hat{Q}^\pi(s_0, a_0)\right] &= \sum_{k=0}^{\infty} \mathbb{E}\left[\sum_{t=0}^{T} r(s_t, a_t) \middle| T = k\right] P(T = k) \quad \text{(by law of total expectation)} \\
&= \sum_{k=0}^{\infty} \mathbb{E}\left[\sum_{t=0}^{k} r(s_t, a_t)\right] (1-\gamma)\gamma^k \quad \text{(each } r(s, a) \text{ is independent of } T\text{)}
\end{aligned}$$

$$
\begin{aligned}
=&(1-\gamma)\sum_{k=0}^{\infty}\frac{\gamma^k}{1-\gamma}\mathbb{E}\left[r(a_k,s_k)\right]\\
=&Q^{\pi}(s_0,a_0)
\end{aligned}
$$

Now, we upperbound the variance. Let $\bar{r}(s,a) := r(s,a) - e(s,a)$ be the expected reward over the zero-mean noise. Because the zero-mean noise is independent of state transition, we observe that:

$$
\begin{aligned}
\mathbb{E}\left[r(s,a)\right] =&\mathbb{E}\left[\bar{r}(s,a)\right]\\
\mathbb{E}\left[r(s,a)^2\right] =&\mathbb{E}\left[(\bar{r}(s,a)+e(s,a))^2\right] = \mathbb{E}\left[\bar{r}(s,a)^2\right] + \mathbb{E}\left[e(s,a)^2\right]\\
\leq&\mathbb{E}\left[\bar{r}(s,a)^2\right] + \sigma^2\\
\mathbb{E}\left[r(s_i,a_i)r(s_j,a_j)\right] =&\mathbb{E}\left[(\bar{r}(s_i,a_i)+e(s_i,a_i))(\bar{r}(s_j,a_j)+e(s_j,a_j))\right]\\
=&\mathbb{E}\left[\bar{r}(s_i,a_i)\bar{r}(s_j,a_j)\right],
\end{aligned}
$$

for $i \neq j$.

Given the above observations, we can bound the variance as follows

$$
\begin{aligned}
&\mathrm{Var}(\hat{Q}^{\pi}(s_0,a_0))\\
\leq\ &\sigma^2 + \mathbb{E}\left[(\hat{Q}^{\pi}(s_0,a_0)-\bar{r}(s_0,a_0))^2\right] - \left(\mathbb{E}\left[\hat{Q}^{\pi}(s_0,a_0)\right]-\bar{r}(s_0,a_0)\right)^2\\
&\text{(separate the variance of } r(s_0,a_0))\\
=\ &\sigma^2 + \sum_{k=1}^{\infty}(1-\gamma)\gamma^k\mathbb{E}\left[\left(\sum_{t=1}^{k} r(s_t,a_t)\right)^2\right] - \left(\mathbb{E}\left[\hat{Q}^{\pi}(s_0,a_0)\right]-\bar{r}(s_0,a_0)\right)^2\\
=\ &\sigma^2 + \sum_{k=1}^{\infty}(1-\gamma)\gamma^k\left(\sum_{t=1}^{k}\mathbb{E}\left[r(s_t,a_t)^2\right] + 2\sum_{i=1}^{k}\sum_{j=i+1}^{k}\mathbb{E}\left[r(s_i,a_i)r(s_j,a_j)\right]\right)\\
&- \left(\mathbb{E}\left[\hat{Q}^{\pi}(s_0,a_0)\right]-\bar{r}(s_0,a_0)\right)^2\\
=\ &\sigma^2 + \sum_{t=1}^{\infty}\gamma^t\mathbb{E}\left[r(s_t,a_t)^2\right] + 2\sum_{i=1}^{\infty}\sum_{j=i+1}^{\infty}\gamma^j\mathbb{E}\left[r(s_i,a_i)r(s_j,a_j)\right]\\
&- \left(\mathbb{E}\left[\hat{Q}^{\pi}(s_0,a_0)\right]-\bar{r}(s_0,a_0)\right)^2
\end{aligned}
$$

$$
\begin{aligned}
&\leq \frac{\sigma^2}{1-\gamma} + \sum_{t=1}^{\infty} \gamma^t \mathbb{E}\left[\bar{r}(s_t,a_t)^2\right] + 2\sum_{i=1}^{\infty}\sum_{j=i+1}^{\infty} \gamma^j \mathbb{E}\left[\bar{r}(s_i,a_i)\bar{r}(s_j,a_j)\right] \\
&\quad - \left(\mathbb{E}\left[\hat{Q}^{\pi}(s_0,a_0)\right] - \bar{r}(s_0,a_0)\right)^2 \\
&\leq \frac{\sigma^2}{1-\gamma} + \sum_{t=1}^{\infty} \gamma^t \mathbb{E}\left[\bar{r}(s_t,a_t)\right] + 2\sum_{i=1}^{\infty}\sum_{j=i+1}^{\infty} \gamma^j \mathbb{E}\left[\bar{r}(s_i,a_i)\right] \\
&\quad - \left(\mathbb{E}\left[\hat{Q}^{\pi}(s_0,a_0)\right] - \bar{r}(s_0,a_0)\right)^2 \\
&= \frac{\sigma^2}{1-\gamma} + \sum_{t=1}^{\infty} \gamma^t \mathbb{E}\left[\bar{r}(s_t,a_t)\right] + 2\sum_{i=1}^{\infty} \frac{\gamma^{i+1}}{1-\gamma} \mathbb{E}\left[\bar{r}(s_i,a_i)\right] \\
&\quad - \left(\mathbb{E}\left[\hat{Q}^{\pi}(s_0,a_0)\right] - \bar{r}(s_0,a_0)\right)^2 \\
&= \frac{\sigma^2}{1-\gamma} + \frac{1+\gamma}{1-\gamma}\sum_{t=1}^{\infty} \gamma^t \mathbb{E}\left[\bar{r}(s_t,a_t)\right] - \left(\sum_{t=1}^{\infty} \gamma^t \mathbb{E}\left[\bar{r}(s_t,a_t)\right]\right)^2 \\
&= -\left(\sum_{t=1}^{\infty} \gamma^t \mathbb{E}\left[\bar{r}(s_t,a_t)\right] - \frac{1+\gamma}{2(1-\gamma)}\right)^2 + \frac{(1+\gamma)^2}{4(1-\gamma)^2} + \frac{\sigma^2}{1-\gamma} \\
&\leq -\left(\sum_{t=1}^{\infty} \gamma^t - \frac{1+\gamma}{2(1-\gamma)}\right)^2 + \frac{(1+\gamma)^2}{4(1-\gamma)^2} + \frac{\sigma^2}{1-\gamma} = \frac{\gamma}{(1-\gamma)^2} + \frac{\sigma^2}{1-\gamma}
\end{aligned}
$$

The last line is because:

$$
\sum_{t=1}^{\infty} \gamma^t \mathbb{E}\left[\bar{r}(s_t,a_t)\right] \leq \sum_{t=1}^{\infty} \gamma^t = \frac{\gamma}{1-\gamma} \leq \frac{1+\gamma}{2(1-\gamma)}.
$$

The equality can be reached by the following reward setting: let $P(1 = \bar{r}(s_1,a_1) = \cdots = \bar{r}(s_t,a_t) = \cdots) = 1$ and therefore is tight. ■

A.4 Proofs for Section 3.4.

Lemma A.4.1 (Lemma 3.4.2)**.** *Suppose the adversarial rewards are bounded in* $[0,1]$*, and in a particular iteration* t*, the adversary contaminates* $\epsilon^{(t)}$ *fraction of the episodes, then given* M *episodes, it is guaranteed that with probability at least* $1-\delta$*,*

$$
\mathbb{E}_{s,a\sim d^{(t)}}\left[\left(Q^{\pi^{(t)}}(s,a) - \phi(s,a)^{\top} w^{(t)}\right)^2\right] \tag{A.4}
$$

$$\leq 4\left(W^2 + WH\right)\left(\epsilon^{(t)} + \sqrt{\frac{8}{M}\log\frac{4d}{\delta}}\right).$$

where $H = (\log\delta - \log M)/\log\gamma$ *is the effective horizon.*

Proof of Lemma A.4.1. First of all, observe that since the adversarial reward is bounded in $[0,1]$, with probability $1-\delta$, the $\hat{Q}(s,a)$ estimates collected in the adversarial episodes are bounded by $H \triangleq (\log\delta - \log M)/\log\gamma$.

Conditioned on the above event, consider three loss functions $\hat{f}$, $f^\dagger$ and f, representing the loss w.r.t. clean data, corrupted data and underlying distribution respectively, i.e.

$$\hat{f} = \frac{1}{M}\sum_{i=1}^{M}(y_i - x_i^\top w)^2 \tag{A.5}$$

$$f^\dagger = \frac{1}{M}\left[\sum_{i\in C}(y_i^\dagger - x_i^{\dagger\top} w)^2 + \sum_{i\notin C}(y_i - x_i^\top w)^2\right] \tag{A.6}$$

$$f = \mathbb{E}(y_i - x_i^\top w)^2 \tag{A.7}$$

Then, for all w, we can make the following decomposition

$$||\nabla_w f^\dagger - \nabla_w f|| \leq ||\nabla_w f^\dagger - \nabla_w \hat{f}|| + ||\nabla_w \hat{f} - \nabla_w f||. \tag{A.8}$$

We next bound each of the two terms in equation A.8. For the first term,

$$\|\nabla_w f^\dagger - \nabla_w \hat{f}\| \tag{A.9}$$

$$= \left\|\frac{2}{M}\sum_{i\in C}\left[(x_i^\dagger x_i^{\dagger\top} - x_i x_i^\top)w + (y_i^\dagger x_i^\dagger - y_i x_i)\right]\right\| \tag{A.10}$$

$$\leq 4\left(W + H\right)\epsilon^{(t)} \tag{A.11}$$

where the last step uses the fact that $|C|/M \leq \epsilon^{(t)}$, and $\|x\| \leq 1$, $|y^\dagger| \leq H$ and $\|w\| \leq W$. For the second term

$$||\nabla_w \hat{f} - \nabla_w f|| \tag{A.12}$$

$$\leq 2\left\|\left(\mathbb{E}[xx^\top] - \frac{1}{M}\sum_{i=1}^{M} x_i x_i^\top\right) w - \left(\mathbb{E}[yx] - \frac{1}{M}\sum_{i=1}^{M} y_i x_i\right)\right\| \tag{A.13}$$

$$\leq 2\left(\frac{2}{3M}\log\frac{4d}{\delta} + \sqrt{\frac{2}{M}\log\frac{4d}{\delta}}\right) W + 2\sqrt{\frac{2}{M}\log\frac{4d}{\delta}} \cdot 2H \tag{A.14}$$

$$\leq 4\sqrt{\frac{8}{M}\log\frac{4d}{\delta}}\,(W+H)\,, \text{for } M \geq 2\log\frac{4d}{\delta}. \tag{A.15}$$

where in step (A.14) we apply Matrix Bernstein inequality Tropp (2015) on the first term and vector Hoeffding's inequality Jin et al. (2019) on the second term. The constant in Corollary 7 of Jin et al. (2019) is instantiated to be $c = 1$, because boundedness means we always have condition 2 in Lemma 2 of Jin et al. (2019). This condition is all we need throughout the proof for the vector Hoeffding.

Now, let M be sufficiently large, and instantiate w to be w^t, i.e. the constrained linear regression solution w.r.t $f^\dagger$, then our result above implies that for any vector v such that $||w + v|| \leq W$, we have $\nabla_w f^\dagger(w^t)^\top v/||v|| \geq 0$, and thus

$$\nabla_w f(w^t)^\top v/||v|| \geq -4\,(W+H)\left(\epsilon^{(t)} + \sqrt{\frac{8}{M}\log\frac{4d}{\delta}}\right) \tag{A.16}$$

which by Lemma B.8 of Diakonikolas et al. (2019b) implies that

$$\epsilon^{(t)}_{stat} \leq 4\left(W^2 + HW\right)\left(\epsilon^{(t)} + \sqrt{\frac{8}{M}\log\frac{4d}{\delta}}\right), \text{w.p. } 1 - 2\delta. \tag{A.17}$$

■

Theorem A.4.1 (Theorem 3.4.1)**.** *Under assumptions 4.2.1 (linear Q function) and 3.3.2 (reset distribution with small κ), given a desired optimality gap α, there exists a set of hyperparameters agnostic to the contamination level ϵ, such that Algorithm 2 guarantees with a $poly(1/\alpha, 1/(1-\gamma), |\mathcal{A}|, W, \sigma, \kappa)$ sample complexity that under ϵ-contamination*

with adversarial rewards bounded in $[0,1]$*, we have*

$$\mathbb{E}\left[V^*(\mu_0) - V^{\hat{\pi}}(\mu_0)\right] \leq \tilde{O}\left(\max\left[\alpha, W\sqrt{\frac{|\mathcal{A}|\kappa\epsilon}{(1-\gamma)^3}}\right]\right)$$

where $\hat{\pi}$ *is the uniform mixture of* $\pi^{(1)}$ *through* $\pi^{(T)}$*.*

Proof of Theorem A.4.1. First note that $\epsilon_{stat} = \mathbb{E}_{s,a\sim d^{(t)}}[\left(\phi(s,a)^\top(w^{(t)} - w^*)\right)^2] \leq 4W^2$, because $\|\phi(s,a)\| \leq 1$ and $\|w^{(t)}\|, \|w^*\| \leq W$. As a result, the high probability bound in Lemma 3.4.2 can be ready translate into an expected bound:

$$\mathbb{E}\left[\mathbb{E}_{s,a\sim d^{(t)}}\left[\left(Q^{\pi^{(t)}}(s,a) - \phi(s,a)^\top w^{(t)}\right)^2\right]\right] \tag{A.18}$$

$$\leq \; 4\left(W^2 + HW\right)\left(\epsilon^{(t)} + \sqrt{\frac{8}{M}\log\frac{4d}{\delta}}\right) + 8\delta W^2 \tag{A.19}$$

where δ becomes a free parameter. Plugging this into Lemma 3.4.1, we get

$$\begin{aligned}
&\mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T}\{V^*(\mu_0) - V^{(t)}(\mu_0)\}\right] \\
\leq\;& \frac{W}{1-\gamma}\sqrt{\frac{2\log|\mathcal{A}|}{T}} + \frac{1}{T}\sum_{t=1}^{T}\sqrt{\frac{4|\mathcal{A}|\kappa\epsilon_{stat}^{(t)}}{(1-\gamma)^3}} \\
\leq\;& \frac{W}{1-\gamma}\sqrt{\frac{2\log|\mathcal{A}|}{T}} + \frac{1}{T}\sum_{t=1}^{T}\sqrt{\frac{16|\mathcal{A}|\kappa\left(\left(W^2+HW\right)\left(\epsilon^{(t)} + \sqrt{\frac{8}{M}\log\frac{4d}{\delta}}\right) + 2\delta W^2\right)}{(1-\gamma)^3}} \\
\leq\;& \frac{W}{1-\gamma}\sqrt{\frac{2\log|\mathcal{A}|}{T}} + \frac{1}{T}\sum_{t=1}^{T}\sqrt{\frac{16|\mathcal{A}|\kappa\left(\left(W^2+HW\right)\sqrt{\frac{8}{M}\log\frac{4d}{\delta}} + 2\delta W^2\right)}{(1-\gamma)^3}} \\
&+\frac{1}{T}\sum_{t=1}^{T}\sqrt{\frac{16|\mathcal{A}|\kappa\left(W^2+HW\right)\epsilon^{(t)}}{(1-\gamma)^3}} \\
\leq\;& \frac{W}{1-\gamma}\sqrt{\frac{2\log|\mathcal{A}|}{T}} + \sqrt{\frac{16|\mathcal{A}|\kappa\left(\left(W^2+HW\right)\sqrt{\frac{8}{M}\log\frac{4d}{\delta}} + 2\delta W^2\right)}{(1-\gamma)^3}}
\end{aligned}$$

$$+\sqrt{\frac{16|\mathcal{A}|\kappa\left(W^2+HW\right)\epsilon}{(1-\gamma)^3}}$$

where the last step is by Cauchy Schwarz and the fact that the attacker only has ϵ budget to distribute, which implies that $\sum_{t=1}^{T}\epsilon^{(t)}=T\epsilon$. Setting

$$T = \frac{2W^2\log|\mathcal{A}|}{\alpha^2(1-\gamma)^2} \tag{A.20}$$

$$\delta = \frac{\alpha^2(1-\gamma)^3}{32W^2|\mathcal{A}|\kappa} \tag{A.21}$$

$$M = \frac{512|\mathcal{A}|^2W^2(W+H)^2\kappa^2}{\alpha^4(1-\gamma)^6}\log\frac{4d}{\delta}, \tag{A.22}$$

we get

$$\mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T}\{V^*(\mu_0)-V^{(t)}(\mu_0)\}\right]\leq 3\alpha+\sqrt{\frac{16|\mathcal{A}|\kappa\left(W^2+HW\right)\epsilon}{(1-\gamma)^3}}. \tag{A.23}$$

with sample complexity

$$TM=\frac{1024|\mathcal{A}|^2\log|\mathcal{A}|W^4(W+H)^2\kappa^2}{\alpha^6(1-\gamma)^8}\log\frac{128W^2|\mathcal{A}|\kappa d}{\alpha^2(1-\gamma)^3}. \tag{A.24}$$

■

A.5 A modified analysis for SEVER

In this section, we will derive an expected error bound for SEVER Diakonikolas et al. (2019b) when applied to a linear regression problem. The high level idea is to use the results of Diakonikolas et al. (2020) to show the existence of a stable set and change the probabilistic argument in Diakonikolas et al. (2019b) to an expectation argument. We note that the original result in Diakonikolas et al. (2019b) works only with probability $9/10$, and there is no direct way of translating it into either a high-probability argument or an expectation argument.

In the following, we consider a robust linear regression problem. We observe pairs $(X_i, Y_i) \in \mathbb{R}^d \times \mathbb{R}$ for $i \in [n]$, where X_i's are drawn i.i.d. from a distribution D_x and $Y_i = w^{*\top} X_i + e_i$ for some unknown $w^* \in \mathbb{R}^d$. e_i's are i.i.d, noise from some distribution $D_{e|x}$. Note that here e_i and X_i may not be independent. We let D_{xy} be the joint distribution of (X, Y). Let $f_i(w) = (Y_i - w^\top X_i)^2$. Given a multiset of observations $\{(X_i, Y_i)\}_{i=1}^n$, our goal is to minimize the objective function

$$\bar{f}(w) = \mathbb{E}_{(X,Y)\sim D_{xy}}[(Y - w^\top X)^2] \tag{A.25}$$

on a convex feasible set $\mathcal{H}$. Let $r := \max_{w\in\mathcal{H}} \|w\|$ be the ℓ_2-radius of $\mathcal{H}$. In the following, we use $\|\cdot\|$ to denote the spectral norm of a matrix and the 2-norm of a vector. We use Cov to denote the covariance matrix of a random vector: $\mathrm{Cov}[X] = \mathbb{E}\left[(X - \mathbb{E}X)(X - \mathbb{E}X)^\top\right]$. When S is a set, we use $\mathbb{E}_S$ and Cov_S to denote the expectation and covariance over the empirical distribution on S. We allow for an ϵ-fraction of the observations to be arbitrary outliers. The ϵ-corruption model is defined in more detail in the Appendix A of Diakonikolas et al. (2019b).

Due to our application, we make assumptions on the linear regression model that is slight different from Assumption E.1 in Diakonikolas et al. (2019b):

Assumption A.5.1. *Given the model for linear regression described above, assume the following conditions for $D_{e|x}$ and D_x:*

- $\mathbb{E}[e|X] = 0$;
- $\mathbb{E}[e^2|X] \le \xi$;
- $\mathbb{E}_{X\sim D_x}[XX^\top] \preceq s^2 I$ *for some* $s > 0$;
- *There is a constant* $C > 0$, *such that for all unit vectors* v, $\mathbb{E}_{X\sim D_x}[\langle v, X\rangle^4] \le Cs^4$.

In Diakonikolas et al. (2019b), the noise term e and X are independent. We weaken the assumption on e and bound its first and second moments conditional on X.

Stability with subgaussian rate

We first note that the gradient of f_i, $\nabla f_i(w)$ has bounded covariance matrix. We will show this by following the proof of Lemma E.3 in Diakonikolas et al. (2019b), but make minor changes as we do not assume e and X are independent:

Lemma A.5.1 (A variant of Lemma E.3 in Diakonikolas et al. (2019b))**.** *Suppose D_{xy} satisfies the conditions of Assumption A.5.1. Then for all unit vectors $v \in \mathbb{R}^d$, we have*

$$v^\top \operatorname*{Cov}_{(X_i,Y_i)\sim D_{xy}} [\nabla f_i(w)] v \leq 4s^2\xi + 4Cs^4\|w^* - w\|^2. \tag{A.26}$$

Proof of Lemma A.5.1. We first note that $f_i(w) = (Y_i - w^\top X_i)^2$ and $\nabla f_i(w) = -2((w^* - w)^\top X_i + e_i)X_i$. By the property of conditional expectation, for any function $g(\cdot), h(\cdot)$, we have

$$\mathbb{E}\left[g(X)h(e)\right] = \mathbb{E}_X\left[\mathbb{E}_{h(e)|X}\left[g(X)h(e)|X\right]\right] = \mathbb{E}_X\left[g(X)\mathbb{E}_{h(e)|X}\left[h(e)|X\right]\right].$$

Then

$$\begin{aligned}
&\mathbb{E}\left[\nabla f_i(w)\nabla f_i(w)^\top\right] = 4\mathbb{E}\left[((w^* - w)^\top X_i + e_i)^2 X_i X_i^\top\right] \\
= \;& 4\mathbb{E}\left[((w^* - w)^\top X_i)^2 X_i X_i^\top\right] + 4\mathbb{E}\left[e_i^2 X_i X_i^\top\right] + 4\mathbb{E}\left[2(w^* - w)^\top X_i e_i X_i X_i^\top\right] \\
= \;& 4\mathbb{E}\left[((w^* - w)^\top X_i)^2 X_i X_i^\top\right] + 4\mathbb{E}\left[X_i X_i^\top \mathbb{E}\left[e_i^2 \middle| X_i\right]\right]
\end{aligned}$$

By Assumption A.5.1, for all unit vectors $v \in \mathbb{R}^d$, we have

$$\begin{aligned}
v^\top \mathbb{E}\left[((w^* - w)^\top X_i)^2 X_i X_i^\top\right] v &= \mathbb{E}\left[((w^* - w)^\top X_i)^2 (v^\top X_i)^2\right] && \text{(A.27)} \\
&\leq \sqrt{\mathbb{E}\left[((w^* - w)^\top X_i)^4\right]\mathbb{E}\left[(v^\top X_i)^4\right]} && \text{(A.28)} \\
&\leq Cs^4\|w^* - w\|^2 && \text{(A.29)}
\end{aligned}$$

and

$$v^\top \mathbb{E}\left[X_i X_i^\top \mathbb{E}\left[e_i^2 \middle| X_i\right]\right] v \leq \xi v^\top \mathbb{E}\left[X_i X_i^\top\right] v \leq s^2\xi \tag{A.30}$$

Thus for all unit vectors $v \in \mathbb{R}^d$, we have

$$v^\top \operatorname*{Cov}_{(X_i,Y_i)\sim D_{xy}}[\nabla f_i(w)]v \leq v^\top \mathbb{E}\left[\nabla f_i(w)\nabla f_i(w)^\top\right] v \leq 4s^2\xi + 4Cs^4\|w^* - w\|^2 \quad \text{(A.31)}$$

■

We then use the following Theorem A.5.1 to show that the observations $f_1, \ldots, f_n$ satisfies the Assumption A.5.2 with high probability:

Theorem A.5.1 (Theorem 1.4 in Diakonikolas et al. (2020)). *Fix any $0 < \tau < 1$. Let S be a multiset of n i.i.d. samples from a distribution on $\mathbb{R}^d$ with mean μ and covariance Σ. Let $\epsilon' = \tilde{C}\left(\log(1/\tau)/n + \epsilon\right) = O(1)$, for some constant $\tilde{C} > 0$. Then, with probability at least $1 - \tau$, there exists a subset $S' \subseteq S$ such that $|S'| \geq (1 - \epsilon')n$ and for every $S'' \subseteq S'$ with $|S''| \geq (1 - 2\epsilon')|S'|$, the following conditions hold: (i) $\|\mu_{S''} - \mu\| \leq \sqrt{\|\Sigma\|}\delta$, and (ii) $\|\overline{\Sigma}_{S''} - \|\Sigma\|I\| \leq \|\Sigma\|\delta^2/(2\epsilon')$, for $\delta = O\left(\sqrt{(d\log d)/n} + \sqrt{\epsilon} + \sqrt{\log(1/\tau)/n}\right)$.*

where $\mu_{S''} = \frac{1}{|S''|}\sum_{x\in S''} x$ and $\overline{\Sigma}_{S''} = \frac{1}{|S''|}\sum_{x\in S''}(x-\mu)(x-\mu)^\top$.

We use a notion of stability similar to that in Diakonikolas et al. (2019b) but allow the parameter to depend on the confidence level and sample size:

Assumption A.5.2 (A variant of Assumption B.1 in Diakonikolas et al. (2019b)). *Fix $0 < \epsilon < 1/2$. With probability at least $1 - \tau$, there exists an unknown set $I_{good} \subseteq [n]$ with $|I_{good}| \geq (1 - \epsilon)n$ of "good" functions $\{f_i\}_{i\in I_{good}}$ and parameters σ, $\alpha(\epsilon, n, \tau), \beta(\epsilon, n, \tau) \in \mathbb{R}_+$ such that for all $w \in \mathcal{H}$:*

$$\left\| \frac{1}{|I_{good}|} \sum_{i\in I_{good}} \nabla f_i(w) - \nabla \bar{f}(w) \right\| \leq \sigma\alpha(\epsilon, n, \tau) \quad \text{(A.32)}$$

and

$$\left\| \frac{1}{|I_{good}|} (\nabla f_i(w) - \nabla \bar{f}(w))(\nabla f_i(w) - \nabla \bar{f}(w))^\top \right\| \leq \sigma^2\beta(\epsilon, n, \tau) \quad \text{(A.33)}$$

We can then equivalently write Theorem A.5.1 as the following Proposition:

Proposition A.5.1. Given a linear regression model $f_i(w) = (Y_i - w^\top X_i)^2$ satisfying Assumption A.5.1, $X_i \sim D_x$, $D_e \sim D_e$, with probability at least $1-\tau$, $\{f_i\}_{i\in[n]}$ satisfies Assumption A.5.2 with $\sigma = 2s\sqrt{\xi} + 2\sqrt{C}s^2\|w^* - w\|$, $\alpha(\epsilon, n, \tau) = O\left(\sqrt{(d\log d)/n} + \sqrt{\epsilon} + \sqrt{\log(1/\tau)/n}\right)$ and $\beta(\epsilon, n, \tau) = \left(\frac{d\log d}{\log(1/\tau)+n\epsilon} + 1\right)$.

Proof of Proposition A.5.1. By Theorem A.5.1 and Lemma A.5.1, with probability at least $1-\tau$, there exist an unknown set $I_{good} \subseteq [n]$ with $|I_{good}| \geq (1-\epsilon')n$, s.t.

$$
\begin{aligned}
& \left\| \frac{1}{|I_{good}|} (\nabla f_i(w) - \nabla \bar{f}(w))(\nabla f_i(w) - \nabla \bar{f}(w))^\top \right\| \\
\leq & \left\| \frac{1}{|I_{good}|} (\nabla f_i(w) - \nabla \bar{f}(w))(\nabla f_i(w) - \nabla \bar{f}(w))^\top - \left\| \operatorname*{Cov}_{f\in p^*}[\nabla f] \right\| I \right\| + \left\| \operatorname*{Cov}_{f\in p^*}[\nabla f] \right\| \\
\leq & \left(4s^2\xi + 4Cs^4\|w^* - w\|^2\right) O\left(\frac{d\log d}{\log(1/\tau) + n\epsilon} + 1\right) \\
\leq & \left(2s\sqrt{\xi} + 2\sqrt{C}s^2\|w^* - w\|\right)^2 O\left(\frac{d\log d}{\log(1/\tau) + n\epsilon} + 1\right) =: \sigma^2\beta(\epsilon, n, \tau).
\end{aligned}
$$

$$
\|\nabla \hat{f}(w) - \nabla \bar{f}(w)\| \leq \sigma O\left(\sqrt{(d\log d)/n} + \sqrt{\epsilon} + \sqrt{\log(1/\tau)/n}\right) =: \sigma\alpha(\epsilon, n, \tau).
$$

■

The expected optimality gap

In order to prove the expected optimality gap, we first state a slightly modified version of the main theorem in Diakonikolas et al. (2019b) by specifying the probability of success;

Theorem A.5.2 (Theorem B.2 in Diakonikolas et al. (2019b))**.** *Let the corruption level $\epsilon \in [0, c]$, for some small enough $c > 0$. Suppose that the functions $f_1, \ldots, f_n, \bar{f} : \mathcal{H} \to \mathbb{R}$ are bounded below, and that Assumption A.5.2 is satisfied. Then SEVER applied to $f_1, \ldots, f_n$ returns a point $w \in \mathcal{H}$ that, fix $p \geq \sqrt{\epsilon}$, with probability at least $1-p$, is a $O\left(\sigma\left(\alpha(\epsilon, n, \tau) + \sqrt{\alpha(\epsilon, n, \tau)^2 + \beta(\epsilon, n, \tau)}\sqrt{\epsilon/p}\right)\right)$-approximate critical point of $\bar{f}$, i.e.*

for all unit vectors v where $w + \lambda v \in \mathcal{H}$ for arbitrarily small positive λ, we have that $v \cdot \nabla f(w) \geq -O\left(\sigma\left(\alpha(\epsilon, n, \tau) + \sqrt{\alpha(\epsilon, n, \tau)^2 + \beta(\epsilon, n, \tau)}\sqrt{\epsilon/p}\right)\right)$.

if $\bar{f}$ is convex, we have the following optimality gap. Recall r is the radius of the convex set $\mathcal{H}$ where w^* belongs.

Corollary A.5.1 (Corollary B.3 in Diakonikolas et al. (2019b))**.** *Let the corruption level $\epsilon \in [0, c]$, for some small enough $c > 0$. For functions $f_1, \ldots, f_n : \mathcal{H} \to \mathbb{R}$, suppose that Assumption A.5.2 holds and that $\mathcal{H}$ is convex. Then, fix $p \geq \sqrt{\epsilon}$, with probability at least $1 - p$, the output of SEVER satisfies the following: if $\bar{f}$ is convex, the algorithm finds a $w \in \mathcal{H}$ such that $\bar{f}(w) - \bar{f}(w^*) = O\left(r\sigma\left(\alpha(\epsilon, n, \tau) + \sqrt{\alpha(\epsilon, n, \tau)^2 + \beta(\epsilon, n, \tau)}\sqrt{\epsilon/p}\right)\right)$*

Given Theorem A.5.1, we can prove the following expected optimality gap:

Theorem A.5.3 (expected optimality gap)**.** *Let the corruption level $\epsilon \in [0, c]$, for some small enough $c > 0$. Let $\mathcal{H}$ be a convex set. Given n samples from a linear regression model $f(w) = (Y - w^\top X)^2$ satisfying Assumption A.5.1, where $X \sim D_x$, $e \sim D_e$, $Y = w^{*\top} X + e$ for some unknown $w^* \in \mathcal{H}$, SEVER will find a $w \in \mathcal{H}$, such that*

$$\mathbb{E}\left[\bar{f}(w) - \bar{f}(w^*)\right] = O\left(\left(sr\sqrt{\xi} + s^2 r^2\right)\left(\tau + \sqrt{(d \log d)/n} + \sqrt{\epsilon} + \sqrt{\log(1/\tau)/n}\right)\right).$$

where the expectation above is over both the randomness of SEVER and (X_i, Y_i) pairs.

Proof of Theorem A.5.3. In the following, we use α and β as shorthands of $\alpha(\epsilon, n, \tau)$ and $\beta(\epsilon, n, \tau)$. We first show that $\bar{f}(w) - \bar{f}(w^*)$ is upper bounded:

$$\begin{aligned}
\bar{f}(w) - \bar{f}(w^*) &= \mathbb{E}_{X,Y}\left[(Y - w^\top X)^2 - (Y - w^{*\top} X)^2\right] && \text{(A.34)} \\
&= \mathbb{E}_{X,e}\left[(w^* - w)^\top X + e)^2 - e^2\right] && \text{(A.35)} \\
&= (w^* - w)^\top \mathbb{E}_X[XX^\top](w^* - w) \leq s^2 (w - w^*)^2 \leq 4s^2 r^2 && \text{(A.36)}
\end{aligned}$$

For some constant $M > 0$, define $x_1 := Mr\sigma\left(\alpha/\sqrt{\epsilon} + \sqrt{\alpha^2 + \beta}\right)\sqrt{\epsilon}$. Let A_1, A_2, A_3 be the following events

$$A_1 = \{\text{Assumption A.5.2 holds}\}$$

$$A_2 = \{\text{SEVER removes less than } (1 + 1/\sqrt{\epsilon})\epsilon n \text{ points}\}$$
$$A_3 = \left\{\bar{f}(w) - \bar{f}(w^*) > Mr\sigma\left(\alpha + \sqrt{\alpha^2 + \beta}\sqrt{\epsilon/p}\right)\right\}.$$

Then, $\forall 0 \leq p < \sqrt{\epsilon}$

$$P(A_2, A_3(p) \mid A_1) = 0. \tag{A.37}$$

By Corollary A.5.1, $\forall \sqrt{\epsilon} \leq p \leq 1$

$$P(A_2, A_3(p) \mid A_1) \leq p. \tag{A.38}$$

By Proposition A.5.1,

$$P(A_1) \geq 1 - \tau. \tag{A.39}$$

By Lemma A.5.3,

$$P(A_2 \mid A_1) \geq 1 - \sqrt{\epsilon}, \tag{A.40}$$

and thus

$$1 - P(A_1, A_2) = 1 - P(A_2 \mid A_1)P(A_1) \leq \tau + \sqrt{\epsilon}. \tag{A.41}$$

Then, we have:

$$P\left(\bar{f}(w) - \bar{f}(w^*) > x_1/\sqrt{p} \mid A_1, A_2\right) \tag{A.42}$$
$$\leq P\left(A_3(p) \mid A_1, A_2\right) = P(A_2, A_3(p) \mid A_1)/P(A_2 \mid A_1) \tag{A.43}$$
$$\leq \begin{cases} 0 & 0 \leq p < \sqrt{\epsilon} \\ \frac{p}{1-\sqrt{\epsilon}} & \sqrt{\epsilon} \leq p \leq 1 \end{cases}. \tag{A.44}$$

Let $x = x_1/\sqrt{p}$, we have:

$$P\left(\bar{f}(w) - \bar{f}(w^*) > x \middle| A_1, A_2\right) \leq \begin{cases} 0 & x \geq x_1\epsilon^{-1/4} \\ \frac{1}{1-\sqrt{\epsilon}}\frac{x_1^2}{x^2} & x_1 \leq x < x_1\epsilon^{-1/4} \\ 1 & 0 \leq x < x_1 \end{cases}. \tag{A.45}$$

By Proposition A.5.1 and law of total expectation, we can bound the expected

optimality gap by:

$$\begin{aligned}
\mathbb{E}\left[\bar{f}(w) - \bar{f}(w^*)\right] &\leq \mathbb{E}\left[\bar{f}(w) - \bar{f}(w^*)\middle| A_1, A_2\right] P(A_1, A_2) + 4s^2r^2(1 - P(A_1, A_2)) \\
&\leq \int_0^\infty P\left(\bar{f}(w) - \bar{f}(w^*) > x \middle| A_1, A_2\right) dx + 4s^2r^2(\tau + \sqrt{\epsilon}) \\
&= \int_0^{x_1} 1 dx + \frac{1}{1 - \sqrt{\epsilon}} \int_{x_1}^{x_1\epsilon^{-1/4}} \frac{x_1^2}{x^2} dx + 4s^2r^2(\tau + \sqrt{\epsilon}) \\
&\leq 2x_1 + 4s^2r^2(\tau + \sqrt{\epsilon}) \\
&= 2Mr\sigma\left(\alpha/\sqrt{\epsilon} + \sqrt{\alpha^2 + \beta}\right)\sqrt{\epsilon} + 4s^2r^2(\tau + \sqrt{\epsilon}) \\
&= O\left(\left(sr\sqrt{\xi} + s^2r^2\right)\left(\tau + \sqrt{(d\log d)/n} + \sqrt{\epsilon} + \sqrt{\log(1/\tau)/n}\right)\right)
\end{aligned}$$

Note that the expectation above is over both the randomness of SEVER and (X_i, Y_i) pairs. ■

Proof of Theorem A.5.2

In this proof, we mainly follow the steps in Diakonikolas et al. (2019b) but use our notion of stability in Assumption A.5.2. We also allow the success probability to vary, so that we can give an expected error bound later on.

We first restate the SEVER algorithm in Algorithm 11 and Algorithm 12. Throughout this proof we let I_{good} be as in Assumption A.5.2. We require the following three lemmas. Roughly speaking, the first states that with high probability, we will not remove too many points throughtout the process, the second states that on average, we remove more corrupted points than uncorrupted points, and the third states that at termination, and if we have not removed too many points, then we have reached a point at which the empirical gradient is close to the true gradient. Formally:

Lemma A.5.2. *If the samples satisfy Assumption A.5.2, $|S| \geq c_1 n$, and the filtering threshold is at least*

$$\frac{2(1-\epsilon)\sigma^2}{c_1 - 2\epsilon}\left(\alpha(\epsilon, n, \tau)^2 + \beta(\epsilon, n, \tau)\right) \tag{A.46}$$

Algorithm 11 SEVER$(f_{1:n}, \mathcal{L}, \sigma)$

Input: Sample functions $f_1, \ldots, f_n : \mathcal{H} \to \mathbb{R}$, bounded below on a closed domain $\mathcal{H}$, γ-approximate learner $\mathcal{L}$, and parameter $\sigma \in \mathbb{R}_+$.
Initialize $S \leftarrow \{1, \ldots, n\}$.
repeat
  $w \leftarrow \mathcal{L}(\{f_i\}_{i \in S})$. ▷ Run approximate learner on points in S.
  Let $\widehat{\nabla} = \frac{1}{|S|} \sum_{i \in S} \nabla f_i(w)$.
  Let $G = [\nabla f_i(w) - \widehat{\nabla}]_{i \in S}$ be the $|S| \times d$ matrix of centered gradients.
  Let v be the top right singular vector of G.
  Compute the vector τ of *outlier scores* defined via $\tau_i = \left((\nabla f_i(w) - \widehat{\nabla}) \cdot v\right)^2$.
  $S' \leftarrow S$
  $S \leftarrow$ FILTER(S', τ, σ) ▷ Remove some i's with the largest scores τ_i from S; see Algorithm 12.
until $S = S'$.
Return w.

Algorithm 12 FILTER(S, τ, σ)

Input: Set $S \subseteq [n]$, vector τ of outlier scores, and parameter $\sigma \in \mathbb{R}_+$.
If $\frac{1}{|S|} \sum_{i \in S} \tau_i \leq c_0 \cdot \sigma^2$, for some constant $c_0 > 1$, return S ▷ We only filter out points if the variance is larger than an appropriately chosen threshold.
Draw T from the uniform distribution on $[0, \max_i \tau_i]$.
Return $\{i \in S : \tau_i < T\}$.

then if S' is the output of FILTER(S, τ, σ)*, we have that*

$$\mathbb{E}[|I_{\text{good}} \cap (S \backslash S')|] \leq \mathbb{E}[|([n] \backslash I_{\text{good}}) \cap (S \backslash S')|]. \tag{A.47}$$

Lemma A.5.3 (Revised version of Lemma 6 in Diakonikolas et al. (2019b))**.** *Assume filtering threshold is $4(\alpha(\epsilon, n, \tau)^2 + \beta(\epsilon, n, \tau))\sigma^2$, $\epsilon \leq 1/16$, then we have that for any given $p \geq \sqrt{\epsilon}$, with probability at least $1 - p$, $n - |S| \leq (1 + 1/p)\epsilon n$ when the filtering algorithm terminates.*

Lemma A.5.4. *If the samples satisfy Assumption A.5.2, FILTER$(S, \tau, \sigma) = S$, and $n - |S| \leq$*

$(1+1/p)\epsilon n$, *for* $p \geq \sqrt{\epsilon}$, *then*

$$\left\|\nabla \bar{f}(w) - \frac{1}{|I_{\text{good}}|}\sum_{i\in S}\nabla f_i(w)\right\|_2 \leq O\left(\sigma\left(\alpha(\epsilon,n,\tau) + \sqrt{\alpha(\epsilon,n,\tau)^2 + \beta(\epsilon,n,\tau)}\sqrt{\epsilon/p}\right)\right) \tag{A.48}$$

Before we prove these lemmata, we show how together they imply Theorem A.5.2.

Proof of Theorem A.5.2 assuming Lemma A.5.3 and Lemma A.5.4. First, we note that the algorithm must terminate in at most n iterations. This is easy to see as each iteration of the main loop except for the last must decrease the size of S by at least 1.

It thus suffices to prove correctness. Note that Lemma A.5.3 says that with probability at least $1-p$, SEVER will not remove too many points, this will allow us to apply Lemma A.5.4 to complete the proof, using the fact that w is a critical point of $\frac{1}{|I_{\text{good}}|}\sum_{i\in S}\nabla f_i(w)$. ■

Thus it suffices to prove these three lemmata.

Proof of Lemma A.5.2. Let $S_{\text{good}} = S \cap I_{\text{good}}$ and $S_{\text{bad}} = S \backslash I_{\text{good}}$. We wish to show that the expected number of elements thrown out of S_{bad} is at least the expected number thrown out of S_{good}. We note that our result holds trivially if $\textsc{Filter}(S,\tau,\sigma) = S$. Thus, we can assume that $\mathbb{E}_{i\in S}[\tau_i] \geq \frac{2(1-\epsilon)\sigma^2}{c_1-2\epsilon}\left(\alpha(\epsilon,n,\tau)^2 + \beta(\epsilon,n,\tau)\right)$.

It is easy to see that the expected number of elements thrown out of S_{bad} is proportional to $\sum_{i\in S_{\text{bad}}}\tau_i$, while the number removed from S_{good} is proportional to $\sum_{i\in S_{\text{good}}}\tau_i$ (with the same proportionality). Hence, it suffices to show that $\sum_{i\in S_{\text{bad}}}\tau_i \geq \sum_{i\in S_{\text{good}}}\tau_i$.

We first note that since $\text{Cov}_{i\in I_{\text{good}}}[\nabla f_i(w)] \preceq \sigma^2 I$, we have that

$$\underset{i\in S_{\text{good}}}{\text{Cov}}[v\cdot\nabla f_i(w)] \leq \frac{1-\epsilon}{c_1-\epsilon}\underset{i\in I_{\text{good}}}{\text{Cov}}[v\cdot\nabla f_i(w)] \quad (\text{since } |S_{\text{good}}| \geq \tfrac{c_1-\epsilon}{1-\epsilon}|I_{\text{good}}|) \tag{A.49}$$

$$= \frac{1-\epsilon}{c_1-\epsilon}\left(\frac{1}{|I_{\text{good}}|}\sum_{i\in I_{\text{good}}}(v\cdot(\nabla f_i(w) - \bar{f}(w)))^2 - (\bar{f}(w) - \mathbb{E}_{i\in I_{\text{good}}}[v\cdot\nabla f_i(w)])^2\right) \tag{A.50}$$

$$\leq \frac{(1-\epsilon)\sigma^2}{c_1-\epsilon}\left(\alpha(\epsilon,n,\tau)^2+\beta(\epsilon,n,\tau)\right) \quad \text{(By Assumption A.5.2)}, \tag{A.51}$$

Let $\mu_{\text{good}} = \mathbb{E}_{i\in S_{\text{good}}}[v\cdot\nabla f_i(w)]$ and $\mu = \mathbb{E}_{i\in S}[v\cdot\nabla f_i(w)]$. Note that

$$\mathbb{E}_{i\in S_{\text{good}}}[\tau_i] = \operatorname*{Cov}_{i\in S_{\text{good}}}[v\cdot\nabla f_i(w)] + (\mu-\mu_{\text{good}})^2 \tag{A.52}$$

$$\leq \frac{(1-\epsilon)\sigma^2}{c_1-\epsilon}\left(\alpha(\epsilon,n,\tau)^2+\beta(\epsilon,n,\tau)\right) + (\mu-\mu_{\text{good}})^2\,. \tag{A.53}$$

We now split into two cases.

Firstly, if

$$(\mu-\mu_{\text{good}})^2 \geq \frac{\epsilon}{c_1-2\epsilon}\frac{(1-\epsilon)\sigma^2}{c_1-\epsilon}\left(\alpha(\epsilon,n,\tau)^2+\beta(\epsilon,n,\tau)\right), \tag{A.54}$$

we let $\mu_{\text{bad}} = \mathbb{E}_{i\in S_{\text{bad}}}[v\cdot\nabla f_i(w)]$, and note that $|\mu-\mu_{\text{bad}}||S_{\text{bad}}| = |\mu-\mu_{\text{good}}||S_{\text{good}}|$. We then have that

$$\mathbb{E}_{i\in S_{\text{bad}}}[\tau_i] = \operatorname*{Cov}_{i\in S_{\text{bad}}}[v\cdot\nabla f_i(w)] + (\mu-\mu_{\text{bad}})^2 \geq (\mu-\mu_{\text{bad}})^2 \tag{A.55}$$

$$= (\mu-\mu_{\text{good}})^2\left(\frac{|S_{\text{good}}|}{|S_{\text{bad}}|}\right)^2 \tag{A.56}$$

$$\geq \frac{|S_{\text{good}}|}{|S_{\text{bad}}|}\frac{c_1-\epsilon}{\epsilon}(\mu-\mu_{\text{good}})^2 \quad \text{(because } |S_{\text{good}}|\geq (c_1-\epsilon)n \text{ and } |S_{\text{bad}}|\leq \epsilon n\text{)} \tag{A.57}$$

$$= \frac{|S_{\text{good}}|}{|S_{\text{bad}}|}\left(\frac{c_1-2\epsilon}{\epsilon}(\mu-\mu_{\text{good}})^2 + (\mu-\mu_{\text{good}})^2\right) \tag{A.58}$$

$$\geq \frac{|S_{\text{good}}|}{|S_{\text{bad}}|}\left(\frac{(1-\epsilon)\sigma^2}{c_1-\epsilon}\left(\alpha(\epsilon,n,\tau)^2+\beta(\epsilon,n,\tau)\right) + (\mu-\mu_{\text{good}})^2\right) \tag{A.59}$$

(by (A.54)) $\tag{A.60}$

$$\geq \frac{|S_{\text{good}}|}{|S_{\text{bad}}|}\mathbb{E}_{i\in S_{\text{good}}}[\tau_i] \quad \text{(by (A.52))}. \tag{A.61}$$

Hence, $\sum_{i\in S_{\text{bad}}}\tau_i \geq \sum_{i\in S_{\text{good}}}\tau_i$.

On the other hand, if $(\mu-\mu_{\text{good}})^2 \leq \frac{\epsilon}{c_1-2\epsilon}\frac{(1-\epsilon)\sigma^2}{c_1-\epsilon}\left(\alpha(\epsilon,n,\tau)^2+\beta(\epsilon,n,\tau)\right)$, then

$\mathbb{E}_{i\in S_{\text{good}}}[\tau_i] \leq \left(1+\frac{\epsilon}{c-2\epsilon}\right)\frac{(1-\epsilon)\sigma^2}{c_1-\epsilon}\left(\alpha(\epsilon,n,\tau)^2+\beta(\epsilon,n,\tau)\right) \leq \mathbb{E}_{i\in S}[\tau_i]/2$. Therefore

$$\sum_{i\in S_{\text{bad}}}\tau_i \geq \sum_{i\in S_{\text{good}}}\tau_i$$

once again. This completes our proof. ■

Proof of Lemma A.5.3. Define the event

$$A = \{n-|S| \leq (1+1/p)\epsilon n\}, \tag{A.62}$$

and we want to lower-bound $P(A)$. Given that $\epsilon \leq 1/16$, the threshold is $4(\alpha(\epsilon,n,\tau)^2+\beta(\epsilon,n,\tau))\sigma^2$ and $p \geq \sqrt{\epsilon}$, and conditioned on the event A, it can be verified that the asusumption of Lemma A.5.2 is satisfied. In particular, simple calculation shows that given $c_1 = 1-(1+1/p)\epsilon$, $\epsilon \leq 1/16$, $p \geq \sqrt{\epsilon}$, we have

$$4\sigma^2 \geq \frac{2(1-\epsilon)\sigma^2}{c_1-2\epsilon} \tag{A.63}$$

And Lemma A.5.2 implies that $|([n]\backslash I_{\text{good}})\cap S| + |I_{\text{good}}\backslash S|$ is a supermartingale. Since its initial size is at most ϵn, with probability at least $1-p$, it never exceeds $\epsilon n/p$, and therefore at the end of the algorithm, we must have that $n-|S| \leq \epsilon n + |I_{\text{good}}\backslash S| \leq (1+1/p)\epsilon n$. ■

We now prove Lemma A.5.4.

Proof of Lemma A.5.4. We note that

$$\left\|\sum_{i\in S}(\nabla f_i(w)-\nabla\bar{f}(w))\right\|_2 \tag{A.64}$$

$$\leq \left\|\sum_{i\in I_{\text{good}}}(\nabla f_i(w)-\nabla\bar{f}(w))\right\|_2 + \left\|\sum_{i\in(I_{\text{good}}\backslash S)}(\nabla f_i(w)-\nabla\bar{f}(w))\right\|_2 \tag{A.65}$$

$$+\left\|\sum_{i\in(S\backslash I_{\text{good}})}(\nabla f_i(w)-\nabla\bar{f}(w))\right\|_2 \tag{A.66}$$

$$\leq \left\| \sum_{i \in (I_{\text{good}} \setminus S)} (\nabla f_i(w) - \nabla \bar{f}(w)) \right\|_2 + \left\| \sum_{i \in (S \setminus I_{\text{good}})} (\nabla f_i(w) - \nabla \bar{f}(w)) \right\|_2 + n\sigma\alpha(\epsilon, n, \tau). \tag{A.67}$$

First we analyze

$$\left\| \sum_{i \in (I_{\text{good}} \setminus S)} (\nabla f_i(w) - \nabla \bar{f}(w)) \right\|_2. \tag{A.68}$$

This is the supremum over unit vectors v of

$$\sum_{i \in (I_{\text{good}} \setminus S)} v \cdot (\nabla f_i(w) - \nabla \bar{f}(w)). \tag{A.69}$$

However, we note that

$$\sum_{i \in I_{\text{good}}} (v \cdot (\nabla f_i(w) - \nabla \bar{f}(w)))^2 \leq n\sigma^2\beta(\epsilon, n, \tau). \tag{A.70}$$

Since $|I_{\text{good}} \setminus S| \leq (1 + 1/p)\epsilon n$, we have by Cauchy-Schwarz that

$$\sum_{i \in (I_{\text{good}} \setminus S)} v \cdot (\nabla f_i(w) - \nabla \bar{f}(w)) = \sqrt{(n\sigma^2\beta(\epsilon, n, \tau))((1 + 1/p)\epsilon n)} \tag{A.71}$$

$$= n\sigma\sqrt{\beta(\epsilon, n, \tau)(1 + 1/p)\epsilon}, \tag{A.72}$$

as desired.

Let

$$\Delta := \left\| \sum_{i \in S} (\nabla f_i(w) - \nabla \bar{f}(w)) \right\|_2. \tag{A.73}$$

Because our Filter algorithm terminates with $n - |S| \leq (1 + 1/p)\epsilon n$, and the stopping condition is set as $\| \frac{1}{|S|} \sum_{i \in S} (\nabla f_i(w) - \nabla \hat{f}(w))(\nabla f_i(w) - \nabla \hat{f}(w))^\top \| \leq 4(\alpha(\epsilon, n, \tau)^2 + \beta(\epsilon, n, \tau))\sigma^2$, we note that since for any such v that

$$\sum_{i \in S} (v \cdot (\nabla f_i(w) - \nabla \bar{f}(w)))^2 \tag{A.74}$$

$$= \sum_{i\in S}(v\cdot(\nabla f_i(w)-\nabla\hat{f}(w)))^2+|S|(v\cdot(\nabla\hat{f}(w)-\nabla\bar{f}(w)))^2 \quad \text{(A.75)}$$

$$\leq \sum_{i\in S}(v\cdot(\nabla f_i(w)-\nabla\hat{f}(w)))^2+\Delta^2/|S| \quad \text{(A.76)}$$

$$\leq n4(\alpha(\epsilon,n,\tau)^2+\beta(\epsilon,n,\tau))\sigma^2+\Delta^2/((1-(1+1/p)\epsilon)n) \quad \text{(A.77)}$$

and since $|S\backslash I_{\text{good}}|\leq(1+1/p)\epsilon n$, and so we have similarly that

$$\left\|\sum_{i\in(S\backslash I_{\text{good}})}\nabla f_i(w)-\nabla\bar{f}(w)\right\|_2 \quad \text{(A.78)}$$

$$\leq 2n\sigma\sqrt{\alpha(\epsilon,n,\tau)^2+\beta(\epsilon,n,\tau)}\sqrt{(1+1/p)\epsilon}+\Delta\sqrt{\frac{(1+1/p)\epsilon}{1-(1+1/p)\epsilon}}. \quad \text{(A.79)}$$

Combining with the above we have that

$$\frac{\Delta}{n}\leq\sigma\alpha(\epsilon,n,\tau)+\sigma\sqrt{\beta(\epsilon,n,\tau)(1+1/p)\epsilon}+2\sigma\sqrt{\alpha(\epsilon,n,\tau)^2+\beta(\epsilon,n,\tau)}\sqrt{(1+1/p)\epsilon} \quad \text{(A.80)}$$

$$+\frac{\Delta}{n}\sqrt{\frac{(1+1/p)\epsilon}{1-(1+1/p)\epsilon}}, \quad \text{(A.81)}$$

Thus

$$\frac{\Delta}{n}\leq\frac{1}{1-\sqrt{\frac{(1+1/p)\epsilon}{1-(1+1/p)\epsilon}}}\left(\sigma\alpha(\epsilon,n,\tau)+6\sigma\sqrt{\alpha(\epsilon,n,\tau)^2+\beta(\epsilon,n,\tau)}\sqrt{\epsilon/p}\right) \quad \text{(A.82)}$$

and therefore, $\frac{\Delta}{n}=O\left(\sigma\left(\alpha(\epsilon,n,\tau)+\sqrt{\alpha(\epsilon,n,\tau)^2+\beta(\epsilon,n,\tau)}\sqrt{\epsilon/p}\right)\right)$ as desired. ■

A.6 Proofs for Section 3.5

Lemma A.6.1 (Lemma 3.5.1)**.** *Suppose the adversarial rewards are unbounded, and in a particular iteration t, the adversarial contaminate $\epsilon^{(t)}$ fraction of the episodes, then given M episodes, it is guaranteed that if $\epsilon^{(t)}\leq c$, for some absolute constant c, and any constant*

$\tau \in [0,1]$, *we have*

$$\begin{aligned}&\mathbb{E}\left[\mathbb{E}_{s,a\sim d^{(t)}}\left[\left(Q^{\pi^{(t)}}(s,a)-\phi(s,a)^\top w^{(t)}\right)^2\right]\right] \\ &\leq O\left(\left(W^2+\frac{\sigma W}{1-\gamma}\right)\left(\sqrt{\epsilon^{(t)}}+f(d,\tau)M^{-\frac{1}{2}}+\tau\right)\right).\end{aligned} \tag{A.83}$$

where $f(d,\tau)=\sqrt{d\log d}+\sqrt{\log(1/\tau)}$.

Proof of Lemma A.6.1. The proof of Lemma 3.5.1 follows by instantiating Theorem A.5.3 to our specific linear regression problem instance. To specify the constants in Theorem A.5.3, we make the following observations

1. By Lemma A.3.1, we have that $\xi=\frac{1}{(1-\gamma)^2}+\frac{\sigma^2}{1-\gamma}$.
2. Since $\|X\|\leq 1$, $\mathbb{E}_{X\sim D_x}\left[XX^\top\right]\leq I$, and thus $s=1$.
3. $\max_{\|v\|=1}\mathbb{E}\left[(v^\top X)^4\right]\leq\mathbb{E}\left[\|v\|^4\|X\|^4\right]\leq 1$, thus $C=1$.

Plugging in the above instantiation to Theorem A.5.3 concludes the proof. ∎

Theorem A.6.1 (Theorem 3.5.1)**.** *Under assumptions 4.2.1 and 3.3.2, given a desired optimality gap* α*, there exists a set of hyperparameters agnostic to the contamination level* ϵ*, such that Algorithm 2, using Algorithm 3 as the linear regression solver, guarantees with a* $poly(1/\alpha,1/(1-\gamma),|\mathcal{A}|,W,\sigma,\kappa)$ *sample complexity that under* ϵ*-contamination, we have*

$$\begin{aligned}&\mathbb{E}\left[V^*(\mu_0)-V^{\hat{\pi}}(\mu_0)\right] \\ &\leq\tilde{O}\left(\max\left[\alpha,\sqrt{\frac{|\mathcal{A}|\kappa\left(W^2+\sigma W\right)}{(1-\gamma)^4}}\epsilon^{1/4}\right]\right).\end{aligned} \tag{A.84}$$

where $\hat{\pi}$ *is the uniform mixture of* $\pi^{(1)}$ *through* $\pi^{(T)}$.

Proof of Theorem A.6.1. Denote $z:=2W$ and again $\epsilon_{stat}\leq(2W)^2=z^2$. Denote $\left(W^2+\frac{\sigma W}{1-\gamma}\right)=b$. Notice that Lemma 3.5.1 only holds when $\epsilon^{(t)}\leq c$ for some absolute constant c, and there are at most $\epsilon T/c$ iterations in which $\epsilon^{(t)}>c$, which

incurs at most $\epsilon_{stat} \leq z^2$ error. Given this observation we can now plugging Lemma 3.5.1 into Lemma 3.4.1, and we get

$$
\begin{aligned}
& \mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T}\{V^*(\mu_0) - V^{(t)}(\mu_0)\}\right] \\
\leq\ & \frac{W}{1-\gamma}\sqrt{\frac{2\log|\mathcal{A}|}{T}} + \frac{1}{T}\sum_{t=1}^{T}\sqrt{\frac{4|\mathcal{A}|\kappa\epsilon_{stat}^{(t)}}{(1-\gamma)^3}} \\
\leq\ & \frac{W}{1-\gamma}\sqrt{\frac{2\log|\mathcal{A}|}{T}} + \frac{z^2}{c}\epsilon \\
& + \frac{1}{T}\sum_{t=1}^{T}\sqrt{\frac{4|\mathcal{A}|\kappa b\left(\sqrt{\epsilon^{(t)}} + \sqrt{(d\log d)/M} + \sqrt{\log(1/\tau)/M} + \tau\right)}{(1-\gamma)^3}} \\
\leq\ & \frac{W}{1-\gamma}\sqrt{\frac{2\log|\mathcal{A}|}{T}} + \frac{z^2}{c}\epsilon + \sqrt{\frac{4|\mathcal{A}|\kappa b\left(\sqrt{(d\log d)/M} + \sqrt{\log(1/\tau)/M} + \tau\right)}{(1-\gamma)^3}} \\
& + \frac{1}{T}\sum_{t=1}^{T}\sqrt{\frac{4|\mathcal{A}|\kappa b\sqrt{\epsilon^{(t)}}}{(1-\gamma)^3}} \\
\leq\ & \frac{W}{1-\gamma}\sqrt{\frac{2\log|\mathcal{A}|}{T}} + \frac{z^2}{c}\epsilon + \sqrt{\frac{4|\mathcal{A}|\kappa b\left(\sqrt{(d\log d)/M} + \sqrt{\log(1/\tau)/M} + \tau\right)}{(1-\gamma)^3}} \\
& + \sqrt{\frac{4|\mathcal{A}|\kappa b}{(1-\gamma)^3}}\epsilon^{1/4}
\end{aligned}
$$

where the last two steps are by Cauchy Schwarz and the fact that the attacker only has ϵ budget to distribute, which implies that $\sum_{t=1}^{T}\epsilon^{(t)} = T\epsilon$. Setting

$$
\begin{aligned}
T &= \frac{2W^2\log|\mathcal{A}|}{\alpha^2(1-\gamma)^2} && \text{(A.85)} \\
\tau &= \frac{\alpha^2(1-\gamma)^3}{4|\mathcal{A}|b\kappa} && \text{(A.86)} \\
M &= \frac{16|\mathcal{A}|^2b^2\kappa^2}{\alpha^4(1-\gamma)^6}\max\left[d\log d, \log(1/\tau)\right] && \text{(A.87)}
\end{aligned}
$$

we get

$$\mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T}\{V^*(\mu_0) - V^{(t)}(\mu_0)\}\right] \leq O\left(\alpha + \sqrt{\frac{|\mathcal{A}|\kappa b}{(1-\gamma)^3}\epsilon^{1/4}}\right). \tag{A.88}$$

with sample complexity

$$TM = \frac{32W^2|\mathcal{A}|^2\log|\mathcal{A}|b^2\kappa^2}{\alpha^6(1-\gamma)^8}\max\left[d\log d, \log(1/\tau)\right]. \tag{A.89}$$

■

A.7 Implementation Details of FPG-TRPO

In the experiment, we use a TRPO variant of FPG implementation, which differs from Alg. 2 in several ways:

1. Most existing TRPO implementation uses the conjugate gradient (CG) method instead of linear regression to solve for the matrix inverse vector product problem. We follow this convention and design FPG-TRPO to use a filtered conjugate gradient (FCG) subroutine to replace the standard CG produce. The FPG procedure is detailed in Alg. 14. At a high level FCG performs a filtering algorithm (a.k.a. outlier removal) on the residues of CG with respect to each data point.

2. Again following existing TRPO implementations, FPG-TRPO builds another network to estimate the value function for the purpose of variance reduction, effectively resulting in an actor-critic algorithm. Instead of performing robust learning procedure on both policy and value function learning, we perform the main filtering algorithm on the policy learning procedure (the CG step discussed above), which also returns a filtered subset of data as a by-product. We then use this filtered subset of data to perform the rest of the learning procedure, including value function update and the sample loss estimation

Algorithm 13 FPG-TRPO

Input: initial policy parameter θ_0; initial value function parameter ϕ_0.
Hyperparameters: KL-divergence limit δ; backtracking coefficient α; maximum number of backtracking steps K; upper-bound of corruption level ϵ; episode length H; batch size M.
for $k = 0, 1, \ldots$ **do**
  Collect set of M trajectories $D_k = \{\tau_i\}_{1:M}$ by running policy $\pi_k = \pi(\theta_k)$ in the environment.
  Compute rewards-to-go $\hat{R}_{t,i} = \sum_{h=t}^{H} \gamma^{h-t} r_{h,i}$.
  Using GAE to compute advantage estimate $\hat{A}_{t,i}$ based on the current value function V_{ϕ_k}.
  Compute and save $\hat{g}_{t,i} = \nabla_\theta \log \pi_\theta(a_{t,i}, s_{t,i})|_{\theta_k}$ for all $t = 1 : H$ and $i = 1 : M$.
  Call the filtered conjugate gradient algorithm in Alg. 14 to get $S_k \subset [M] \times [H], \hat{x}_k = FCG(\hat{g}_{t,i}, \hat{A}_{t,i})$.
  Compute policy gradient estimate $\hat{g}_k = \frac{1}{|S_k|} \sum_{(t,i)\in S_k} \hat{g}_{t,i} \hat{A}_{t,i}$.
  Update the policy by backtracking line search with

$$\theta_{k+1} = \theta_k + \alpha^j \sqrt{\frac{2\delta}{\hat{x}_k \hat{g}_k}} \hat{x}_k \tag{A.90}$$

  where $j \in \{0, 1, 2, ..., K\}$ is the smallest value which improves the sample loss and satisfies the sample KL-divergence constraint.
  Fit the value function by regression on mean-squared error on the filtered trajectories S_k:

$$\phi_{k+1} = \underset{\phi}{\operatorname{argmin}} \frac{1}{|S_k|} \sum_{(t,i)\in S_k} \left(V_\phi(s_{t,i}) - \hat{R}_{t,i} \right)^2 \tag{A.91}$$

  In practice, one often only take a few gradient steps in each iteration k, instead of optimizing to convergence.
end for

in backtracking line search. This allows us to perform the robust learning procedure only once per PG iteration.

3. FPG-TRPO uses a deterministic variant of the filtering algorithm suggested in Diakonikolas et al. (2019b), which empirically performs better and is simpler

Algorithm 14 Filtered Conjugate Gradient (FCG)

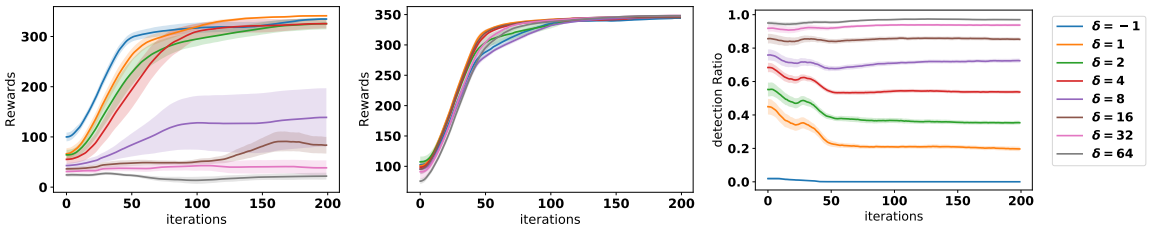
Input: $\hat{g}_{t,i}, \hat{A}_{t,i}$
Hyperparameters: Number of iterations r (default $r = 4$), fraction of data filtered in each iteration p (default $p = \epsilon/2$, i.e. filter out 2ϵ data in total).
Initialize $S = \{1, 2, \ldots, M\}$.
for $k = 1, \ldots, r$ **do**
  Call standard CG to solve for $\hat{x} = \hat{F}^{-1}\hat{g}$, where $\hat{F} = \frac{1}{S}\sum_{(t,i)\in S} \hat{g}_{t,i}\hat{g}_{t,i}^\top$ and $\hat{g} = \frac{1}{S}\sum_{(t,i)\in S} \hat{g}_{t,i}\hat{A}_{t,i}$.
  Compute the residues $r_{t,i} = \hat{g}_{t,i}\hat{g}_{t,i}^\top\hat{x} - \hat{g}_{t,i}\hat{A}_{t,i}$ for $(t,i) \in S$ and save in a matrix G of size $d \times |S|$.
  Let v be the top right singular vector of G.
  Compute the vector τ of *outlier scores* defined via $\tau_{t,i} = \left(r_{t,i}^\top v\right)^2$.
  Remove (HMp) number of (t,i) pair with the largest outlier scores from S.
end for
Call standard CG one more time and return $(S, \hat{x})$.

to implement than the stochastic variant used for theoretical analysis. Specifically, the filtering algorithm will simply remove a fixed fraction of points with the largest deviation along the top singular value direction (step 9 of Alg. 14).

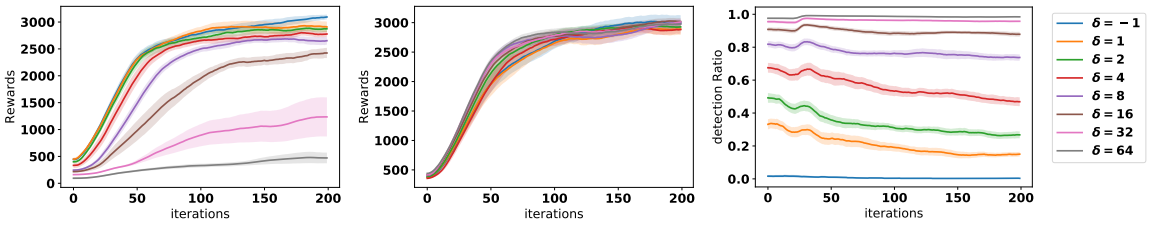
The pseudo-code of `FPG-TRPO` can be found in Alg. 13. Similar to the NPG variant of FPG, the only difference between Alg. 13 and a standard TRPO implementation is the replacement of the CG subroutine with the FCG subroutine. This modular implementation allows one to easily replace Alg. 14 with any state-of-the-art robust CG procedure in the future. Table A.1 lists all the hyper-parameters we used in our experiments, which are taken from open-source implementations of TRPO tuned for the MuJoCo environments. Our code to reproduce the experiment result is included in the supplementary material and will be open-sourced. Finally, Figure A.1 presents the detailed results on all experiments, completing the partial results shown in Figure 3.3.

Parameters	Values	Description
γ	0.995	discounting factor.
λ	0.97	GAE parameter Schulman et al. (2015b).
l2-reg	0.001	L2 regularization weight in value loss.
δ	0.01	KL constraint in TRPO.
damping	0.1	damping factor in conjugate gradient.
batch-size	25000	number of time steps per policy gradient iteration.
α	0.5	backtracking coefficient.
K	10	maximum number of backtracking steps.

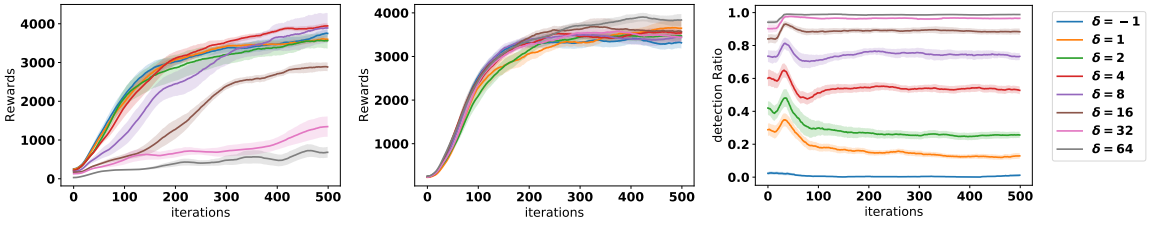
Table A.1: Hyperparameters for FPG-TRPO.



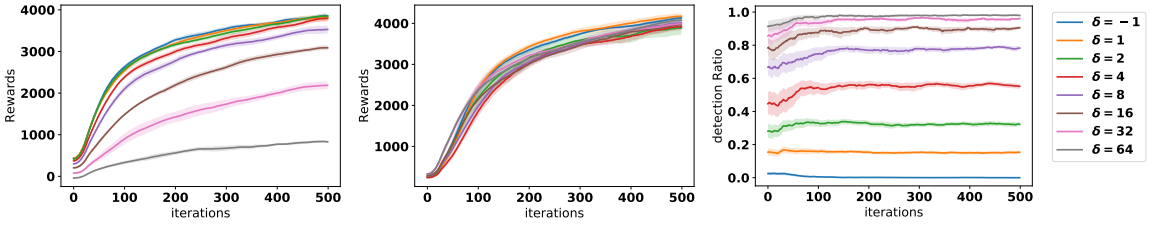
(a) TRPO on Swimmer (b) FPG on Swimmer (c) Detection Ratio on Swimmer



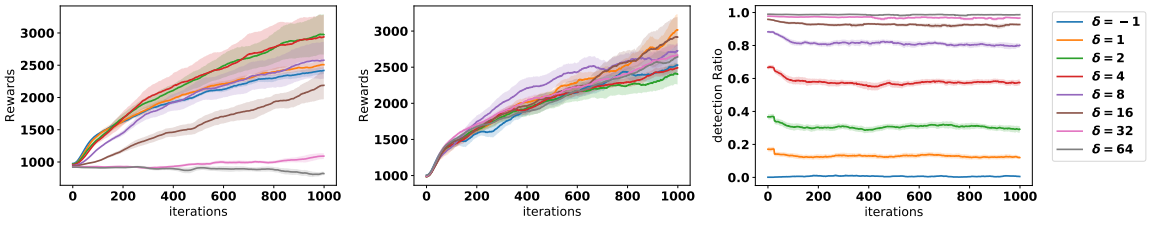
(d) TRPO on Hopper (e) FPG on Hopper (f) Detection Ratio on Hopper



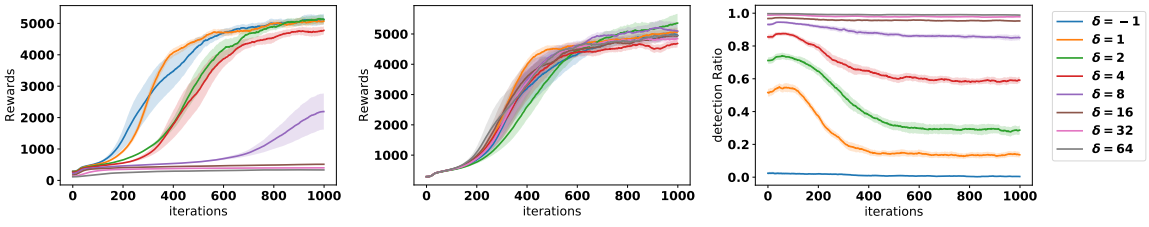
(g) TRPO on Walker2d (h) FPG on Walker2d (i) Detection Ratio on Walker2d



(j) TRPO on HalfCheetah (k) FPG on HalfCheetah (l) Detection Ratio on HalfCheetah



(m) TRPO on Ant (n) FPG on Ant (o) Detection Ratio on Ant



(p) TRPO on Humanoid (q) FPG on Humanoid (r) Detection Ratio on Humanoid

Figure A.1: Detailed Results on the MuJoCo benchmarks.

B.1 Basics

Lemma B.1.1. *$\|w_h^*\| \leq H\sqrt{d}$ for all h.*

Proof. By definition, we have

$$w_h^* = \theta + \int_{\mathcal{S}} \hat{V}_{h+1}(s')\mu_h(s')ds' \tag{B.1}$$

and thus

$$\begin{aligned}
\|w_h^*\| &\leq \|\theta\| + \|\int_{\mathcal{S}} \hat{V}_{h+1}(s')\mu_h(s')ds'\| && \text{(B.2)}\\
&\leq \|\theta\| + \int_{\mathcal{S}} \|\hat{V}_{h+1}(s')\mu_h(s')\|ds' && \text{(B.3)}\\
&\leq \sqrt{d} + (H-h+1)\sqrt{d} && \text{(B.4)}\\
&\leq H\sqrt{d}. && \text{(B.5)}
\end{aligned}$$

■

Lemma B.1.2. *Note that $\mathbb{E}[[(r(s,a) + \hat{V}(s')) - (\mathbf{B}_h\hat{V})(s,a)]^2|s,a] \leq \gamma^2 = (\sigma + H/2)^2$*

Proof.

$$\begin{aligned}
Var(X+Y) =& Var(X) + Var(Y) + 2Cov(X,Y) && \text{(B.6)}\\
\leq& Var(X) + Var(Y) + 2\sqrt{Var(X)Var(Y)} && \text{(B.7)}
\end{aligned}$$

Because $0 \leq \hat{V}(s') \leq H$,

$$\begin{aligned}
&\mathbb{E}[(\hat{V}(s') - \mathbb{E}[\hat{V}(s')|s,a])^2|s,a] = \mathbb{E}[\hat{V}(s')^2|s,a] - \mathbb{E}[\hat{V}(s')|s,a]^2 && \text{(B.8)}\\
\leq& H\mathbb{E}[\hat{V}(s')|s,a] - \mathbb{E}[\hat{V}(s')|s,a]^2 \leq \frac{H^2}{4}. && \text{(B.9)}
\end{aligned}$$

$$\mathbb{E}[[(r(s,a)+\hat{V}(s'))-(\mathbf{B}_h\hat{V})(s,a)]^2|s,a] \tag{B.10}$$
$$=\mathbb{E}[[(r(s,a)+\hat{V}(s'))-\mathbb{E}[r(s,a)+\hat{V}(s')|s,a]]^2|s,a] \tag{B.11}$$
$$=\mathbb{E}[(r(s,a)-\mathbb{E}[r(s,a)|s,a])^2|s,a]+\mathbb{E}[(\hat{V}(s')-\mathbb{E}[\hat{V}(s')|s,a])^2|s,a] \tag{B.12}$$
$$+2\mathbb{E}[(r(s,a)-\mathbb{E}[r(s,a)|s,a])(\hat{V}(s')-\mathbb{E}[\hat{V}(s')|s,a])|s,a] \tag{B.13}$$
$$\leq\mathbb{E}[(r(s,a)-\mathbb{E}[r(s,a)|s,a])^2|s,a]+\mathbb{E}[(\hat{V}(s')-\mathbb{E}[\hat{V}(s')|s,a])^2|s,a] \tag{B.14}$$
$$+2\sqrt{\mathbb{E}[(r(s,a)-\mathbb{E}[r(s,a)|s,a])^2|s,a]\mathbb{E}[(\hat{V}(s')-\mathbb{E}[\hat{V}(s')|s,a])^2|s,a]} \tag{B.15}$$
$$\text{(By Cauchy's Ineq)} \tag{B.16}$$
$$=Var(r(s,a)\mid(s,a))+Var(\hat{V}(s')\mid(s,a)) \tag{B.17}$$
$$+2\sqrt{Var(r(s,a)\mid(s,a))Var(\hat{V}(s')\mid(s,a))} \tag{B.18}$$
$$=\left(\sqrt{Var(r(s,a)\mid(s,a))}+\sqrt{Var(\hat{V}(s')\mid(s,a))}\right)^2\leq(\sigma+H/2)^2 \tag{B.19}$$

■

B.2 Proof of the Minimax Lower-bound

Proof of Theorem 4.3.1. Given any dimension d, time horizon H, consider a tabular MDP with action space size $A>2$ and state space size $S\leq\left(\frac{A}{2}\right)^{H/2}$ s.t. $SA=d$. Consider a "tree" with self-loops, which has S nodes and depth $\lceil\log_{A/2}\left(S\left(\frac{A}{2}-1\right)+1\right)\rceil$. There is 1 node at the first level, $\frac{A}{2}$ nodes at the second level, $\left(\frac{A}{2}\right)^2$ nodes at the third level, ..., $\left(\frac{A}{2}\right)^{\lceil\log_{A/2}\left(S\left(\frac{A}{2}-1\right)+1\right)\rceil-2}$ nodes at the second to last level. The rest nodes are all at the last level. Define the MDP induced by this graph, where each state corresponds to a node, and each action corresponds to an edge. The agent always starts from the first level. For each state at the first $\lceil\log_{A/2}\left(S\left(\frac{A}{2}-1\right)+1\right)\rceil-2$ levels, there are $A/2$ actions that lead to child nodes, and the rest leads back to that state, i.e. self-loops. The leaf states are absorbing state, i.e. all actions lead to self-loops. Denote this transition structure as P. Let's consider two MDPs with the same transition structure and different reward function, i.e. $M=(P,R)$, $M'=(P,R')$.

For MDP M, define $R(s^*,a^*)=\text{Bernoulli}(SA\epsilon/2)$ on one particular (s^*,a^*) pair,

where s^* is a leaf state at the last level, a^* is a self-loop action. Every other (s, a) pair receive reward 0. Let $(s', a') = \operatorname{argmin}_{(s,a)} \nu(s, a)$ be the state-action pair appears least often in the data collecting distribution. For MCP M', define $R'(s^*, a^*) =$ Bernoulli$(SA\epsilon/2)$, $R'(s', a') =$ Bernoulli$(SA\epsilon)$ and 0 everywhere else. Then, it can be easily verified that: on M, the expected cumulative reward of the optimal policy is $\left(H - \lceil \log_{A/2}\left(S\left(\frac{A}{2} - 1\right) + 1\right)\rceil\right) SA\epsilon/2$; on M', the expected cumulative reward of the optimal policy is at least $\left(H - \lceil \log_{A/2}\left(S\left(\frac{A}{2} - 1\right) + 1\right)\rceil\right) SA\epsilon$; no policy can be simultaneously better than $\left(H - \lceil \log_{A/2}\left(S\left(\frac{A}{2} - 1\right) + 1\right)\rceil\right) SA\epsilon/4$-optimal on both M and M'. Note that because $S \leq \left(\frac{A}{2}\right)^{H/2}$,

$$\left(H - \lceil \log_{A/2}\left(S\left(\frac{A}{2} - 1\right) + 1\right)\rceil\right) SA\epsilon/4 = \Omega(HSA\epsilon). \tag{B.20}$$

With probability at least $1/2$, we have $N(s', a') \leq T\nu(s', a') \leq T/SA$ by the pigeonhole principle. Conditioning on $N(s', a') \leq T/SA$, with probability at least $1/2$, the amount of positive reward $r(s', a')$ will not exceed $SA\epsilon N(s', a') \leq \epsilon T$, and thus an ϵ-contamination adversary can perturb all the positive rewards on (s', a') to 0. In other words, with probability $1/4$, the learner will observe a dataset whose likelihood under M and ($M' + \epsilon$-contamination) are exactly the same, and thus the learner must suffer at least $\Omega(HSA\epsilon)$ regret on one of the MDPs. ■

B.3 Proof of Upper-bounds

Proof of Lemma C.4.1. Applying Lemma B.6.2 with $\pi = \hat{\pi}$, $\pi' = \tilde{\pi}$, and $\{\hat{Q}_h\}_{h=1}^H$ being the Q-functions constructed by the meta-algorithm, we have

$$\begin{aligned}\hat{V}_1(s) - V_1^{\tilde{\pi}}(s) = &\sum_{h=1}^{H} \mathbb{E}_{\tilde{\pi}}\left[\langle \hat{Q}_h(s_h, \cdot), \hat{\pi}_h(\cdot|s_h) - \tilde{\pi}_h(\cdot|s_h)\rangle_{\mathcal{A}} | s_1 = s\right] \\ &+ \sum_{h=1}^{H} \mathbb{E}_{\tilde{\pi}}\left[\hat{Q}_h(s_h, a_h) - (\mathbf{B}_h \hat{V}_{h+1})(s_h, a_h) | s_1 = s\right]\end{aligned} \tag{B.21}$$

Similarly, applying Lemma B.6.2 with $\pi = \pi' = \hat{\pi}$, we have

$$\hat{V}_1(s) - V_1^{\hat{\pi}}(s) = \sum_{h=1}^{H} \mathbb{E}_{\hat{\pi}} \left[\hat{Q}_h(s_h, a_h) - (\mathbf{B}_h \hat{V}_{h+1})(s_h, a_h) | s_1 = s \right] \tag{B.22}$$

Then, we have

$$\begin{aligned}
\mathrm{SubOpt}(\hat{\pi}, \tilde{\pi}) =& \left(V_1^{\tilde{\pi}}(\mu) - \hat{V}_1(\mu) \right) + \left(\hat{V}_1(\mu) - V_1^{\hat{\pi}}(\mu) \right) && \text{(B.23)} \\
=& - \sum_{h=1}^{H} \mathbb{E}_{\tilde{\pi}} \left[(\mathbf{B}_h \hat{V}_{h+1}) - \hat{Q}_h \right] + \sum_{h=1}^{H} \mathbb{E}_{\tilde{\pi}} \left[(\mathbf{B}_h \hat{V}_{h+1}) - \hat{Q}_h \right] && \text{(B.24)} \\
&+ \sum_{h=1}^{H} \mathbb{E}_{\tilde{\pi}} \left[\langle \hat{Q}_h(s_h, \cdot), \tilde{\pi}_h(\cdot|s_h) - \hat{\pi}_h(\cdot|s_h) \rangle_{\mathcal{A}} \right] && \text{(B.25)} \\
\leq& 0 + 2 \sum_{h=1}^{H} \mathbb{E}_{\tilde{\pi}} [\Gamma_h(s, a)] + 0 && \text{(B.26)} \\
=& 2 \sum_{h=1}^{H} \mathbb{E}_{\tilde{\pi}} [\Gamma_h(s, a)] && \text{(B.27)}
\end{aligned}$$

as needed. ■

Proof of Theorem 4.3.3. To simplify the notation, below we use M for the number of data points per time step, i.e. $M \triangleq N/H$. We first show that

$$|\hat{Q}_h(s, a) - (\mathbf{B}_h \hat{V}_{h+1})(s, a)| \leq \Gamma(s, a). \tag{B.28}$$

The robust least-square oracle guarantees

$$\begin{aligned}
\mathbb{E}_\nu \left(\|x^\top (\hat{w} - w^*)\|_2^2 \right) \leq c_2(\delta) \cdot \left(\frac{\gamma^2 \mathrm{poly}(d)}{M} + \gamma^2 \epsilon \right) && \text{(B.29)} \\
\implies \|\hat{w}_h - w_h^*\|_\Sigma^2 \leq c_2(\delta) \cdot \left(\frac{\gamma^2 \mathrm{poly}(d)}{M} + \gamma^2 \epsilon \right) && \text{(B.30)} \\
\implies \|\hat{w}_h - w_h^*\|_{\Sigma + (2\epsilon + \lambda) I}^2 \leq c_2(\delta) \cdot \left(\frac{\gamma^2 \mathrm{poly}(d)}{M} + \gamma^2 \epsilon + (2\epsilon + \lambda) H^2 d \right) && \text{(B.31)}
\end{aligned}$$

Then,

$$|\hat{Q}_h(s,a) - (\mathbf{B}_h\hat{V}_{h+1})(s,a)| = |\phi(s,a)(\hat{w}_h - w_h^*)| \tag{B.32}$$

$$\leq \|\hat{w}_h - w_h^*\|_{(\Sigma+(2\epsilon+\lambda)I)}\|\phi(s,a)\|_{(\Sigma+(2\epsilon+\lambda)I)^{-1}} \tag{B.33}$$

$$\leq \sqrt{c_2(\delta)\cdot\left(\frac{\gamma^2\mathrm{poly}(d)}{M} + \gamma^2\epsilon + (2\epsilon+\lambda)H^2 d\right)}\|\phi(s,a)\|_{(\Sigma+(2\epsilon+\lambda)I)^{-1}} \tag{B.34}$$

$$\leq \sqrt{c_2(\delta)}\cdot\left(\frac{\gamma\mathrm{poly}(d)}{\sqrt{M}} + (\gamma + 2H\sqrt{d})\sqrt{\epsilon} + H\sqrt{d\lambda}\right)\|\phi(s,a)\|_{\Lambda^{-1}} \tag{B.35}$$

where the last step are due to $W \leq H\sqrt{d}$ and

$$\Lambda = \frac{3}{5}\left(\frac{1}{M}\sum_{i=1}^{M}\phi_i\phi_i^\top + (\epsilon+\lambda)\cdot I\right) \tag{B.36}$$

$$\preceq \frac{3}{5}\left(\frac{1}{M}\sum_{i=1}^{M}\tilde{\phi}_i\tilde{\phi}_i^\top + (2\epsilon+\lambda)\cdot I\right) \tag{B.37}$$

$$\preceq (\Sigma + (2\epsilon+\lambda)\cdot I) \tag{B.38}$$

where $\tilde{\phi}$ denotes the clean data and the last step applies Lemma B.6.3 because $M(2\epsilon+\lambda) \geq \Omega(d\log(M/\delta))$ due to the definition of λ and $\epsilon \geq 0$.

Next, we show that Algorithm 4 achieves the desired optimality gap. By Lemma C.4.1, we have

$$\mathrm{SubOpt}(\hat{\pi}) \leq 2H\mathbb{E}_{\pi^*}[\Gamma(s,a)] \tag{B.39}$$

$$\leq \sqrt{c_2(\delta)}\cdot\left(\frac{\gamma H\mathrm{poly}(d)}{\sqrt{N}} + (H\gamma + 2H^2\sqrt{d})\sqrt{\epsilon} + H^2\sqrt{d\lambda}\right)\mathbb{E}_{\pi^*}\left[\|\phi(s,a)\|_{\Lambda^{-1}}\right] \tag{B.40}$$

Focusing on the last term, applying Lemma B.6.3 again, we have

$$\mathbb{E}_{d^*}[\|\phi(s,a)\|_{\Lambda^{-1}}] \leq \mathbb{E}_{d^*}[\|\phi(s,a)\|_{(\frac{1}{5}(\Sigma+\lambda I))^{-1}}] \tag{B.41}$$

$$= \mathbb{E}_{d^*}\left[\sqrt{\phi^\top(\frac{1}{5}(\Sigma+\lambda I))^{-1}\phi}\right] \tag{B.42}$$

$$\leq \sqrt{\mathbb{E}_{d^*}[\phi^\top (\frac{1}{5}(\Sigma + \lambda I))^{-1}\phi]} \tag{B.43}$$

$$\leq \sqrt{tr\left(\Sigma_* (\frac{1}{5}(\Sigma + \lambda I))^{-1}\right)} \tag{B.44}$$

$$\leq \sqrt{\kappa tr\left(\Sigma (\frac{1}{5}(\Sigma + \lambda I))^{-1}\right)} \tag{B.45}$$

$$\leq \sqrt{5\kappa \sum_{i=1}^{d} \frac{\sigma_i}{\sigma_i + \lambda}} \tag{B.46}$$

$$\leq \sqrt{5d\kappa} \tag{B.47}$$

Combining the two terms give the desired results. ■

B.4 Proof of uncorrupted learning results

In this section, we prove the conclusion in Corollary 4.3.1 and 4.3.2. The proof follows closely the classic analysis of Least Squared Value Iteration (LSVI) methods with the only difference being the data splitting, which allows us to ditch the covering argument and obtain a tighter bound. Such a trick is only possible in the offline setting where the data are assumed to be i.i.d. For completeness, we specify the uncorrupted algorithm in Alg. 15.

Algorithm 15 Uncorrupted Least-Square Value Iteration (LSVI)

Input: Dataset $D = \{(s_i, a_i, r_i, s_i')\}_{1:N}$; pessimism bonus $\Gamma_h(s,a) \geq 0$, $\lambda > 0$.
Split the dataset randomly into H subset: $D_h = \{(s_i^h, a_i^h, r_i^h, s_i'^h)\}_{1:(N/H)}$, for $h \in [H]$.
Initialization: Set $\hat{V}_{H+1}(s) \leftarrow 0$.
for step $h = H, H-1, \ldots, 1$ **do**
  Set $\Lambda_h \leftarrow \frac{H}{M} \sum_{i=1}^{N/H} \phi(s_i^h, a_i^h)\phi(s_i^h, a_i^h)^\top + \lambda \cdot I$.
  Set $\hat{w}_h \leftarrow \Lambda_h^{-1}(\frac{H}{N} \sum_{i=1}^{N/H} \phi(s_i^h, a_i^h) \cdot (r_i^h + \hat{V}_{h+1}(s_i^{h+1})))$.
  Set $\hat{Q}_h(s,a) \leftarrow \phi(s,a)^\top \hat{w}_h - \Gamma_h(s,a)$, clipped within $[0, H-h+1]$.
  Set $\hat{\pi}_h(a|s) \leftarrow \operatorname{argmax}_a \hat{Q}_h(s,a)$ and $\hat{V}_h(s) \leftarrow \max_a \hat{Q}_h(s,a)$.
end for
Output: $\{\hat{\pi}_h\}_{h=1}^H$.

We first prove the following lemma:

Lemma B.4.1 (Bound on the Bellman Error)**.** *Under assumption 4.2.1, given a dataset of size N, Algorithm 4 achieves*

$$|(\mathbb{B}_h \hat{V}_{h+1})(s,a) - \hat{Q}_h(s,a)| \leq H \left(\sqrt{d \cdot \lambda} + \sqrt{\frac{Hd \log(N/\delta\lambda)}{N}} \right) \cdot \sqrt{\phi(x,a)^\top \Lambda_h^{-1} \phi(x,a)}$$

for all $(s,a,h) \in \mathcal{S} \times \mathcal{A} \times [H]$, with probability at least $1-\delta$.

Proof. We start by applying the following decomposition

$$\begin{aligned}
&(\mathbb{B}_h \hat{V}_{h+1})(s,a) - \hat{Q}_h(s,a) && \text{(B.48)}\\
=&(\mathbb{B}_h \hat{V}_{h+1})(s,a) - (\hat{\mathbb{B}}_h \hat{V}_{h+1})(s,a) && \text{(B.49)}\\
=&\underbrace{\phi(s,a)^\top w_h - \phi(s,a)^\top \Lambda_h^{-1} \left(\frac{H}{N} \sum_{i=1}^{N/H} \phi(s_i,a_i) \cdot (\mathbb{B}_h \hat{V}_{h+1})(s_i,a_i) \right)}_{\text{(i)}} - && \text{(B.50)}\\
&\underbrace{\phi(s,a)^\top \Lambda_h^{-1} \left(\frac{H}{N} \sum_{i=1}^{N/H} \phi(s_i,a_i) \cdot \left(r_i + \hat{V}_{h+1}(s_i') - (\mathbb{B}_h \hat{V}_{h+1})(s_i,a_i) \right) \right)}_{\text{(ii)}} && \text{(B.51)}
\end{aligned}$$

Therefore, by triangle inequality we have

$$|(\mathbb{B}_h \hat{V}_{h+1})(s,a) - \hat{Q}_h(s,a)| \leq |(\mathrm{i})| + |(\mathrm{ii})| \tag{B.52}$$

Then, we bound the two terms separately:

$$\begin{aligned}
|(\mathrm{i})| &= \left| \phi(s,a)^\top w_h - \phi(s,a)^\top \Lambda_h^{-1} \left(\frac{H}{N} \sum_{i=1}^{N/H} \phi(s_i,a_i) \cdot \phi(s_i,a_i)^\top w_h \right) \right| \\
&= \left| \phi(s,a)^\top w_h - \phi(s,a)^\top \Lambda_h^{-1} (\Lambda_h - \lambda \cdot I) w_h \right| = \lambda \cdot |\phi(s,a)^\top \Lambda_h^{-1} w_h| \\
&\leq \lambda \cdot \|w_h\|_{\Lambda_h^{-1}} \cdot \|\phi(s,a)\|_{\Lambda_h^{-1}} \leq H\sqrt{d \cdot \lambda} \cdot \sqrt{\phi(s,a)^\top \Lambda_h^{-1} \phi(s,a)}.
\end{aligned}$$

For the second term, define

$$\epsilon_i^h(V) = r_i^h + V(s_i^{h\prime}) - (\mathbf{B}_h V)(s_i^h, a_i^h) \tag{B.53}$$

Then, we have

$$\begin{aligned}
|(\mathrm{ii})| &= \left| \phi(s,a)^\top \Lambda_h^{-1} \left(\frac{H}{N} \sum_{i=1}^{N/H} \phi(s_i,a_i) \cdot \epsilon_i^h(\hat{V}_{h+1}) \right) \right| \\
&\leq \underbrace{\left\| \frac{H}{N} \sum_{i=1}^{N/H} \phi(s_i,a_i) \cdot \epsilon_i^h(\hat{V}_{h+1}) \right\|_{\Lambda_h^{-1}}}_{(\mathrm{iii})} \cdot \sqrt{\phi(x,a)^\top \Lambda_h^{-1} \phi(x,a)}.
\end{aligned} \tag{B.54}$$

From here, because of our data splitting, $\hat{V}_{h+1}$ is independent from D_h, and thus we can bypass the covering argument and directly apply matrix concentrations. In particular, by applying Lemma B.6.1, we have that with probability at least $1-\delta$

$$(\mathrm{iii}) \leq H\sqrt{\frac{Hd\log(1+N/H\lambda) + 2H\log(1/\delta)}{N}} \tag{B.55}$$

Combining the two terms gives

$$|(\mathbb{B}_h\hat{V}_{h+1})(s,a) - \hat{Q}_h(s,a)| \leq H\left(\sqrt{d\cdot\lambda} + \sqrt{\frac{Hd\log(N/\delta\lambda)}{N}}\right)\cdot\sqrt{\phi(x,a)^\top\Lambda_h^{-1}\phi(x,a)} \tag{B.56}$$

■

Now, given Lemma B.4.1, applying Lemma C.4.1, we have

$$\begin{aligned}
\mathrm{SubOpt}(\hat{\pi},\tilde{\pi}) \leq & 2\sum_{h=1}^{H}\mathbb{E}_{d^{\tilde{\pi}}}[\Gamma_h(s,a)] && \text{(B.57)}\\
\leq & 2H^2\left(\sqrt{d\cdot\lambda} + \sqrt{\frac{Hd\log(N/\delta\lambda)}{N}}\right)\cdot\mathbb{E}_{d^{\tilde{\pi}}}[\sqrt{\phi(x,a)^\top\Lambda_h^{-1}\phi(x,a)}] && \text{(B.58)}
\end{aligned}$$

The last step would be to bound $\mathbb{E}_{d^{\tilde{\pi}}}[\sqrt{\phi(x,a)^\top\Lambda_h^{-1}\phi(x,a)}]$, similar to the last section. In particular, applying Lemma B.6.3, we have

$$\begin{aligned}
\mathbb{E}_{d^{\tilde{\pi}}}\left[\sqrt{\phi(x,a)^\top\Lambda_h^{-1}\phi(x,a)}\right] \leq & \mathbb{E}_{d^{\tilde{\pi}}}\left[\sqrt{3\phi(x,a)^\top(\Sigma+\lambda I)\phi(x,a)}\right] && \text{(B.59)}\\
\leq & \sqrt{3\mathbb{E}_{d^{\tilde{\pi}}}\left[\phi(x,a)^\top(\Sigma+\lambda\cdot I)\phi(x,a)\right]} && \text{(B.60)}\\
\leq & \sqrt{3d\kappa} && \text{(B.61)}
\end{aligned}$$

where step B.59 requires $\lambda \geq H\Omega(d\log(N/\delta))/N$. Thus,

$$\begin{aligned}
\mathrm{SubOpt}(\hat{\pi},\tilde{\pi}) \leq & 2H^2\left(\sqrt{d\cdot\lambda} + \sqrt{Hd\log(N/\delta\lambda)}\right)\sqrt{\frac{3d\kappa}{N}} && \text{(B.62)}\\
\leq & \tilde{O}\left(H^2\left(d\sqrt{\log(N/\delta)} + \sqrt{Hd\log(N/(d\delta))}\right)\sqrt{\frac{3d\kappa}{N}}\right) && \text{(B.63)}
\end{aligned}$$

B.5 Lower-bound on best-of-both-world results

Proof of Theorem 4.3.4. Consider two instances of the offline RL problem, with two MDPs, M and M', both of which are actually simple two-arm bandit problems,

along with their data generating distribution ν and ν', defined below.

1. Instance 1: Bandit M has $r_1 = \text{Bernoulli}(\frac{1}{2} + \frac{\epsilon}{2p})$ and $r_2 = \text{Bernoulli}(\frac{1}{2})$. The data generating distribution is $\nu(a_1) = p$ and $\nu(a_2) = 1 - p$. The relative condition number is $1/p$.

2. Instance 2: Bandit M has $r_1 = \text{Bernoulli}(\frac{1}{2} - \frac{\epsilon}{2p})$ and $r_2 = \text{Bernoulli}(\frac{1}{2})$. The data generating distribution is $\nu(a_1) = p$ and $\nu(a_2) = 1 - p$, same as instance 1. The relative condition number is $1/(1-p)$.

Let D and D' be i.i.d. datasets of size N generated by instances 1 and 2, respectively, generated by the following *coupling* process. First, the actions are sampled from ν and shared across instances, e.g. $N_D(a_1) = N_{D'}(a_1)$ and $N_D(a_2) = N_{D'}(a_2)$. Then, the rewards of a_2 are sampled from $\text{Bernoulli}(\frac{1}{2})$ and shared across tasks, e.g. $N_D(a_2, 0) = N_{D'}(a_2, 0)$ and $N_D(a_2, 1) = N_{D'}(a_2, 1)$.

Finally, let X_i, Y_i be Bernoulli random variables s.t. $X_i = \begin{cases} 0 & U \leq \frac{1}{2} - \frac{\epsilon}{2p} \\ 1 & \text{o.w.} \end{cases}$, $Y_i = \begin{cases} 0 & U \leq \frac{1}{2} + \frac{\epsilon}{2p} \\ 1 & \text{o.w.} \end{cases}$, where U is picked uniformly random in $[0, 1]$. Then (X_i, Y_i) is a coupling with law: $P((X_i, Y_i) = (0, 0)) = \frac{1}{2} - \frac{\epsilon}{2p}$, $P((X_i, Y_i) = (1, 0)) = 0$, $P((X_i, Y_i) = (0, 1)) = \frac{\epsilon}{2p}$, $P((X_i, Y_i) = (s_3, s_3)) = \frac{1}{2} - \frac{\epsilon}{2p}$, X_i and Y_i can be thought as the outcome of $\text{Bernoulli}(\frac{1}{2} + \frac{\epsilon}{2p})$, $\text{Bernoulli}(\frac{1}{2} + \frac{\epsilon}{2p})$ respectively. Then, let the rewards of a_1 of the two instances be generated by Y_i and X_i respectively. We then have

$$P(\sum_{i=1}^{N(a_1)} \mathbb{1} X_i \neq Y_i) \geq P(N(a_1) \leq pN) \cdot P(\sum_{i=1}^{pN} \mathbb{1} X_i \neq Y_i) \geq \frac{1}{2} \cdot \frac{1}{2} = \frac{1}{4} \tag{B.64}$$

In other words, with probability at least $\frac{1}{4}$, instance 1 and 2 are indistinguishable under ϵ-contamination, in particular the adversary can replace at most ϵN of $(a_1, 0)$ with $(a_1, 1)$ in D' to replicate D. Therefore, instance 1 and (instance 2 + ϵ-contamination) are with probability at least $1/4$ indistinguishable. Now, if an algorithm wants to achieve best of both world guarantee, it must return a_1 as the

optimal arm with high probability when observing a dataset generated as above, in which case it will suffer a suboptimality of $\frac{\epsilon}{2p}$ if the data is generated by (instance 2 + ϵ-contamination). As $p \geq \epsilon \geq 0$ goes to 0, this gap blows up, while the relative condition number $1/(1-p)$ remains bounded, thus contradiction.

■

B.6 Technical Lemmas

Lemma B.6.1 (Concentration of Self-Normalized Processes (Abbasi-Yadkori et al., 2011)). *Let $\{\epsilon_t\}_{t=1}^{\infty}$ be a real-valued stochastic process that is adaptive to a filtration $\{\mathcal{F}_t\}_{t=0}^{\infty}$. That is, ϵ_t is $\mathcal{F}_t$-measurable for all $t \geq 1$. Moreover, we assume that, for any $t \geq 1$, conditioning on $\mathcal{F}_{t-1}$, ϵ_t is a zero-mean and σ-subGaussian random variable such that*

$$\mathbb{E}[\epsilon_t|\mathcal{F}_{t-1}] = 0 \qquad \textit{and} \qquad \mathbb{E}[\exp(\lambda\epsilon_t)|\mathcal{F}_{t-1}] \leq \exp(\lambda^2\sigma^2/2), \quad \forall\lambda \in \mathbb{R}. \tag{B.65}$$

Besides, let $\{\phi_t\}_{t=1}^{\infty}$ be an $\mathbb{R}^d$-valued stochastic process such that ϕ_t is $\mathcal{F}_{t-1}$-measurable for all $t \geq 1$. Let $M_0 \in \mathbb{R}^{d\times d}$ be a deterministic and positive-definite matrix, and we define $M_t = M_0 + \sum_{s=1}^{t} \phi_s\phi_s^\top$ for all $t \geq 1$. Then for any $\delta > 0$, with probability at least $1-\delta$, we have for all $t \geq 1$ that

$$\Big\|\sum_{s=1}^{t} \phi_s \cdot \epsilon_s\Big\|_{M_t^{-1}}^2 \leq 2\sigma^2 \cdot \log\Big(\frac{\det(M_t)^{1/2}\det(M_0)^{-1/2}}{\delta}\Big).$$

Lemma B.6.2 (Extended Value Difference (Cai et al., 2020)). *Let $\pi = \{\pi_h\}_{h=1}^{H}$ and $\pi' = \{\pi'_h\}_{h=1}^{H}$ be two arbitrary policies and let $\{\hat{Q}_h\}_{h=1}^{H}$ be any given Q-functions. For any $h \in [H]$, we define a value function $\hat{V}_h\colon \mathcal{S} \to \mathbb{R}$ by letting $\hat{V}_h(x) = \langle \hat{Q}_h(x,\cdot), \pi_h(\cdot|x)\rangle_{\mathcal{A}}$ for all $s \in \mathcal{S}$. Then for all $s \in \mathcal{S}$, we have*

$$\hat{V}_1(s) - V_1^{\pi'}(s) = \sum_{h=1}^{H} \mathbb{E}_{\pi'}\Big[\langle \hat{Q}_h(s_h,\cdot), \pi_h(\cdot|s_h) - \pi'_h(\cdot|s_h)\rangle_{\mathcal{A}}|s_1 = s\Big] \tag{B.66}$$

$$+\sum_{h=1}^{H}\mathbb{E}_{\pi'}\left[\hat{Q}_h(s_h,a_h)-(\mathbf{B}_h\hat{V}_{h+1})(s_h,a_h)|s_1=s\right], \tag{B.67}$$

where the expectation $\mathbb{E}_{\pi'}$ is taken with respect to the trajectory generated by π', and $\mathbf{B}_h$ is the Bellman operator.

Lemma B.6.3 (Concentration of Covariances (Zanette et al., 2021))**.** *Let $\{\phi_i\}_{1:N}\subset\mathbb{R}^d$ be i.i.d. samples from an underlying bounded distribution ν, with $\|\phi_i\|_i\leq 1$ and covariance Σ. Define*

$$\Lambda=\sum_{i=1}^{N}\phi_i\phi_i^\top+\lambda\cdot I \tag{B.68}$$

for some $\lambda\geq\Omega(d\log(N/\delta))$. Then, we have that with probability at least $(1-\delta)$,

$$\frac{1}{3}(N\Sigma+\lambda I)\preceq\Lambda\preceq\frac{5}{3}(N\Sigma+\lambda I) \tag{B.69}$$

Proof. See (Zanette et al., 2021) Lemma 39 for a detailed proof. ∎

C.1 More Discussion on page 56:COW

Impossibility Result

Theorem C.1.1 (impossibility result)**.** *There exists a distribution $\mathcal{D}$, s.t. given m data batches $\left\{\left\{x_j^i\right\}_{i=1}^{n_j}\right\}_{j\in[m]}$ generated under page 54, every robust mean estimation algorithm $\mathscr{A}$ suffers an error of at least*

$$\Omega\left(\frac{1}{\sqrt{N}}\right) \tag{C.1}$$

even $\mathscr{A}$ knows some of the batches are clean, where N is the sum of sizes of the smallest $(1-2\alpha)m$ good batches.

Proof of Theorem C.1.1. Let $\mathcal{D}$ be Bernoulli distribution with parameter $\frac{1}{2}$. W.l.o.g., assume $\mathcal{G} = [(1-\alpha)m]$, $n_1 \leq \cdots \leq n_{(1-\alpha)m}$ and $\mathcal{B} = \{(1-\alpha)m+1, \ldots, m\}$. We assume algorithm $\mathscr{A}$ knows $[(1-2\alpha)m]$ is a subset of the good batches.

Let $\eta = \frac{1}{2\sqrt{N}} = \frac{1}{2\sqrt{\sum_{j=1}^{(1-2\alpha)m} n_j}}$. Let the bad batches $\mathcal{B}$ be i.i.d. samples from $\mathcal{D}'$, a Bernoulli distribution with parameter $\frac{1}{2}+\eta$. By Theorem 4 of (Paninski, 2008; Chan et al., 2014), no algorithm can distinguish if the batches

$$\left\{x_1^i\right\}_{i=1}^{n_1}, \ldots, \left\{x_{(1-2\alpha)m}^i\right\}_{i=1}^{n_{(1-2\alpha)m}}$$

are sampled from $\mathcal{D}$ or $\mathcal{D}'$. I.e. no algorithm can distinguish if

$$\{(1-2\alpha)m+1, \ldots, (1-\alpha)m\}$$

are good batches or $\mathcal{B}$ are good batches.

This means, given m data batches $\left\{\left\{x_j^i\right\}_{i=1}^{n_j}\right\}_{j\in[m]}$, every robust mean estimation algorithm suffers an error at least $\Omega\left(\frac{1}{\sqrt{N}}\right)$. ∎

Adaption To Good Batch Perturbation And Distributed Learning

Compared to page 56, page 197 enlarges the confidence interval by ϵ on both endpoints due to the perturbation and only requires some sufficient statistics from the batches, instead of the whole dataset. When $n^{\text{cut}} > 0$, meaning there are at least $2\alpha m + 1$ non-empty batches, page 197 runs a modified COW algorithm to calculate the mean estimation and the error upper bound . When $n^{\text{cut}} = 0$, page 197 returns 0 and a trivial error upper bound.

Algorithm 16 PERT-COW

Require: Batch empirical means: $\hat{\mu}_1, \ldots, \hat{\mu}_m$; batch sizes: $n_1, \ldots, n_m$; subGaussian parameter σ; corruption level α; confidence level δ

$n^{\text{cut}} \leftarrow$ the $(2\alpha m + 1)$-th largest batch size
if $n^{\text{cut}} \leq 0$ **then**
  Error $\leftarrow \infty$
  return $\hat{\mu} \leftarrow 0$, Error
end if
$I_j \leftarrow \left[\hat{\mu}_j - \frac{\sigma}{\sqrt{\tilde{n}_j}}\sqrt{2\log\frac{2m}{\delta}} - \epsilon, \hat{\mu}_j + \frac{\sigma}{\sqrt{\tilde{n}_j}}\sqrt{2\log\frac{2m}{\delta}} + \epsilon\right], \forall j \in [m]$
$C^* \leftarrow \text{argmax}_{C \subseteq [m]: \bigcap_{j \in C} I_j \neq \varnothing} |C|$
$\tilde{n}_j \leftarrow \min(n_j, n^{\text{cut}}), \forall j \in [m]$
$\hat{\mu} \leftarrow \frac{1}{\sum_{j \in C^*} \tilde{n}_j} \sum_{j \in C^*} \tilde{n}_j \hat{\mu}_j$,
Error $\leftarrow$ RHS of page 199
return $\hat{\mu}$, Error

C.2 Proof of Theorem 5.3.1

To prove Theorem 5.3.1, we show page 56 holds under some concentration event while the event happens with high probability. We consider a slightly more general setting where there could be perturbations to even good batches:

Definition C.2.1 (Robust mean estimation from batches)**.** There are m data providers indexed by: $\{1, 2, \ldots, m\} =: [m]$. Among these providers, we denote the indexes of uncorrupted providers by $\mathcal{G}$ and the indexes of corrupted providers by $\mathcal{B}$, where

$\mathcal{B} \cup \mathcal{G} = [m]$, $\mathcal{B} \cap \mathcal{G} = \varnothing$, $|\mathcal{B}| = \alpha m$. Any uncorrupted providers have access to **perturbed** samples from a sub-Gaussian distribution $\mathcal{D}$ with mean μ and variance proxy σ^2 (i.e. $\mathbb{E}_{X\sim\mathcal{D}}[X] = \mu$ and $\mathbb{E}_{X\sim\mathcal{D}}\left[\exp\left(s\left(X-\mu\right)\right)\right] \leq \exp\left(\sigma^2 s^2/2\right), \forall s \in \mathbb{R}.$). **For each $j \in \mathcal{G}$, a data batch $\left\{\tilde{x}_j^i\right\}_{i=1}^{n_j}$ is drawn from $\mathcal{D}$, while a perturbed version $\left\{x_j^i\right\}_{i=1}^{n_j}$ is sent to the learner, where n_j can be arbitrary and $\left|x_j^i - \tilde{x}_j^i\right| \leq \epsilon$ for some $\epsilon \geq 0$.** For $j \in \mathcal{B}$, $\left\{x_j^i\right\}_{i=1}^{n_j}$ can be arbitrary.

One can easily recover page 54 by letting $\epsilon = 0$. page 197 only requires the empirical mean $\hat{\mu}_j := \frac{1}{n_j}\sum_{i=1}^{n_j} x_j^i$ and size n_j of each batch $j \in [m]$.. We first define the concentration event as follows:

Definition C.2.2 (Concentration event)**.** For all $j \in \mathcal{G}$, define the event that the empirical mean of clean batches is close to the population mean as:

$$\mathcal{E}_j := \left\{|\hat{\mu}_j - \mu| \leq \frac{\sigma}{\sqrt{n_j}}\sqrt{2\log\frac{2m}{\delta}} + \epsilon\right\} \tag{C.2}$$

Define the event that the weighted average of empirical means of clean batches is close to the population mean as:

$$\mathcal{E}_{wa} := \left\{\left|\frac{1}{\sum_{j\in\mathcal{G}}\tilde{n}_j}\sum_{j\in\mathcal{G}}\tilde{n}_j\hat{\mu}_j - \mu\right| \leq \frac{\sigma}{\sqrt{\sum_{j\in\mathcal{G}}\tilde{n}_j}}\sqrt{2\log\frac{2}{\delta}} + \epsilon\right\} \tag{C.3}$$

Let $\mathcal{E}_{conc}$ be the event that the events above happen together:

$$\mathcal{E}_{conc} := \mathcal{E}_{wa} \cap \bigcap_{j\in\mathcal{G}} \mathcal{E}_j \tag{C.4}$$

We can show $\mathcal{E}_{conc}$ happens with high probability using Hoeffding's inequality:

Lemma C.2.1. $\mathbb{P}\left(\mathcal{E}_{conc}\right) \geq 1 - 2\delta$.

Proof. See proof in page 199. ∎

Under event $\mathcal{E}_{conc}$, we can give an upper bound on the estimation error:

Lemma C.2.2. *Under event $\mathcal{E}_{conc}$, if $n^{\text{cut}} > 0$, page 197 outputs a $\hat{\mu}$ with*

$$|\hat{\mu} - \mu| \leq \frac{2}{\sqrt{\sum_{j\in[m]} \tilde{n}_j}}\sigma\sqrt{2\log\frac{2}{\delta}} + \frac{8\alpha m\sqrt{n^{\text{cut}}}}{\sum_{j\in[m]} \tilde{n}_j}\sigma\sqrt{2\log\frac{2m}{\delta}} + 5\epsilon \tag{C.5}$$

Proof. See proof in page 202. ■

Proof of Theorem 5.3.1. Consider $\epsilon = 0$, i.e. no perturbation involved. By page 198 and page 199, with probability at least $1 - 2\delta$,

$$|\hat{\mu} - \mu| \leq \frac{2}{\sqrt{\sum_{j\in[m]} \tilde{n}_j}}\sigma\sqrt{2\log\frac{2}{\delta}} + \frac{8\alpha m\sqrt{n^{\text{cut}}}}{\sum_{j\in[m]} \tilde{n}_j}\sigma\sqrt{2\log\frac{2m}{\delta}} \tag{C.6}$$

■

Proof of Lemma C.2.1

To prove page 198,

1. we first show that the perturbation changes the empirical mean of batches by at most ϵ;
2. we can show the concentration bound of empirical means and weighted means for the **unperturbed** samples;
3. we can conclude by using the two results above and triangular inequality.

The Probability Of Event $\bigcap_{j\in\mathcal{G}} \mathcal{E}_j$: For all $j \in \mathcal{G}$, let $\bar{x}_j$ be the empirical mean of **unperturbed** samples in batch j:

$$\bar{\mu}_j := \frac{1}{n_j}\sum_{i=1}^{n_j} \tilde{x}_j^i \tag{C.7}$$

By triangular inequality:

$$|\bar{\mu}_j - \hat{\mu}_j| = \left| \frac{1}{n_j} \sum_{i=1}^{n_j} (x_j^i - \tilde{x}_j^i) \right| \leq \frac{1}{n_j} \sum_{i=1}^{n_j} \epsilon = \epsilon \tag{C.8}$$

Since $\mathcal{D}$ is sub-Gaussian distribution, we can show the concentration of unperturbed samples mean $\bar{\mu}_j$: for all good batch $j \in \mathcal{G}$,

$$\mathbb{P}\left(|\bar{\mu}_j - \mu| > t\right) \leq 2 \exp\left(-\frac{n_j t^2}{2\sigma^2}\right) \tag{C.9}$$

By union bound, with probability at least $1 - \delta$, $\forall j \in \mathcal{G}$,

$$|\bar{\mu}_j - \mu| \leq \frac{\sigma}{\sqrt{n_j}} \sqrt{2 \log \frac{2\,|\mathcal{G}|}{\delta}} \leq \frac{\sigma}{\sqrt{n_j}} \sqrt{2 \log \frac{2m}{\delta}} \tag{C.10}$$

By triangular inequality, with probability at least $1 - \delta$, $\forall j \in \mathcal{G}$,

$$|\hat{\mu}_j - \mu| \leq |\hat{\mu}_j - \bar{\mu}_j| + |\bar{\mu}_j - \mu| \leq \frac{\sigma}{\sqrt{n_j}} \sqrt{2 \log \frac{2m}{\delta}} + \epsilon \tag{C.11}$$

I.e. $\mathbb{P}\left(\bigcap_{j \in \mathcal{G}} \mathcal{E}_j\right) \geq 1 - \delta$.

The Probability Of Event $\mathcal{E}_{wa}$: We first show the weighted average of empirical mean of the **unperturbed** sample i.e., $\frac{1}{\sum_{j' \in \mathcal{G}} \tilde{n}_{j'}} \sum_{j \in \mathcal{G}} \tilde{n}_j \bar{\mu}_j$ is a sub-Gaussian random variable: firstly, note that the mean of the weighted average is μ, i.e. $\mathbb{E}\left[\frac{1}{\sum_{j' \in \mathcal{G}} \tilde{n}_{j'}} \sum_{j \in \mathcal{G}} \tilde{n}_j \bar{\mu}_j\right] = \mu$. By definition, we know for good batch $j \in \mathcal{G}$, $\tilde{x}_j^1, \ldots, \tilde{x}_j^{n_j}$ are i.i.d. sub-Gaussian random variable with mean μ and variance proxy σ^2, i.e.

$$\mathbb{E}\left[\exp\left(s\left(\tilde{x}_j^i - \mu\right)\right)\right] \leq \exp\left(\frac{\sigma^2 s^2}{2}\right) \quad \forall s \in \mathbb{R}. \tag{C.12}$$

Since $\bar{\mu}_j = \frac{1}{n_j}\sum_{i=1}^{n_j} \tilde{x}_j^i$: for all $s \in \mathbb{R}$,

$$\mathbb{E}\left[\exp\left(s\left(\frac{1}{\sum_{j'\in\mathcal{G}}\tilde{n}_{j'}}\sum_{j\in\mathcal{G}}\tilde{n}_j\bar{\mu}_j - \mu\right)\right)\right] = \prod_{j\in\mathcal{G}}\mathbb{E}\left[\exp\left(s\left(\frac{1}{\sum_{j'\in\mathcal{G}}\tilde{n}_{j'}}\tilde{n}_j(\bar{\mu}_j - \mu)\right)\right)\right] \tag{C.13}$$

$$= \prod_{j\in\mathcal{G}}\prod_{i\in[n_j]}\mathbb{E}\left[\exp\left(\frac{s}{\sum_{j'\in\mathcal{G}}\tilde{n}_{j'}}\frac{\tilde{n}_j}{n_j}(\tilde{x}_j^i - \mu)\right)\right] \leq \prod_{j\in\mathcal{G}}\prod_{i\in[n_j]}\exp\left(\frac{\sigma^2}{2}\left(\frac{s}{\sum_{j'\in\mathcal{G}}\tilde{n}_{j'}}\frac{\tilde{n}_j}{n_j}\right)^2\right) \tag{C.14}$$

$$\leq \exp\left(\frac{\sigma^2}{2}\left(\frac{s}{\sum_{j'\in\mathcal{G}}\tilde{n}_{j'}}\right)^2\sum_{j\in\mathcal{G}}\sum_{i\in[n_j]}\left(\frac{\tilde{n}_j}{n_j}\right)^2\right) = \exp\left(\frac{\sigma^2}{2}\left(\frac{s}{\sum_{j'\in\mathcal{G}}\tilde{n}_{j'}}\right)^2\sum_{j\in\mathcal{G}}\frac{\tilde{n}_j}{n_j}\tilde{n}_j\right) \tag{C.15}$$

$$\leq \exp\left(\frac{\sigma^2}{2}\left(\frac{s}{\sum_{j'\in\mathcal{G}}\tilde{n}_{j'}}\right)^2\sum_{j\in\mathcal{G}}\tilde{n}_j\right) = \exp\left(\frac{s^2}{2}\left(\frac{\sigma}{\sqrt{\sum_{j'\in\mathcal{G}}\tilde{n}_{j'}}}\right)^2\right) \tag{C.16}$$

This means $\frac{1}{\sum_{j'\in\mathcal{G}}\tilde{n}_{j'}}\sum_{j'\in\mathcal{G}}\tilde{n}_{j'}\bar{\mu}_j$ is a sub-Gaussian random variable with variance proxy $\frac{\sigma^2}{\sum_{j'\in\mathcal{G}}\tilde{n}_{j'}}$. Thus $\forall t > 0$,

$$\mathbb{P}\left(\left|\frac{1}{\sum_{j'\in\mathcal{G}}\tilde{n}_{j'}}\sum_{j\in\mathcal{G}}\tilde{n}_j\bar{\mu}_j - \mu\right| > t\right) \leq 2\exp\left(-\frac{\sum_{j'\in\mathcal{G}}\tilde{n}_{j'}t^2}{2\sigma^2}\right) \tag{C.17}$$

Thus with probability at least $1-\delta$:

$$\left|\frac{1}{\sum_{j'\in\mathcal{G}}\tilde{n}_{j'}}\sum_{j\in\mathcal{G}}\tilde{n}_j\bar{\mu}_j - \mu\right| \leq \frac{\sigma}{\sqrt{\sum_{j'\in\mathcal{G}}\tilde{n}_{j'}}}\sqrt{2\log\frac{2}{\delta}} \tag{C.18}$$

This means:

$$\left|\frac{1}{\sum_{j'\in\mathcal{G}}\tilde{n}_{j'}}\sum_{j\in\mathcal{G}}\tilde{n}_j\hat{\mu}_j - \mu\right| \tag{C.19}$$

$$\leq \left| \frac{1}{\sum_{j' \in \mathcal{G}} \tilde{n}_{j'}} \sum_{j \in \mathcal{G}} \tilde{n}_j \bar{\mu}_j - \mu \right| + \left| \frac{1}{\sum_{j' \in \mathcal{G}} \tilde{n}_{j'}} \sum_{j \in \mathcal{G}} \tilde{n}_j \bar{\mu}_j - \frac{1}{\sum_{j' \in \mathcal{G}} \tilde{n}_{j'}} \sum_{j \in \mathcal{G}} \tilde{n}_j \hat{\mu}_j \right| \tag{C.20}$$

$$\leq \frac{\sigma}{\sqrt{\sum_{j' \in \mathcal{G}} \tilde{n}_{j'}}} \sqrt{2 \log \frac{2}{\delta}} + \epsilon \tag{C.21}$$

I.e. $\mathbb{P}(\mathcal{E}_{wa}) \geq 1 - \delta$.

By union bound $\mathbb{P}(\mathcal{E}_{conc}) = \mathbb{P}\left(\mathcal{E}_{wa} \cap \bigcap_{j \in \mathcal{G}} \mathcal{E}_j\right) \geq 1 - 2\delta$.

Proof of Lemma C.2.2

By page 198, we know the weighted average of the empirical mean of good batches is a proper estimation for the population mean. Compared to $\mathcal{G}$, the C^* returned in page 197 in page 197 may remove some good batches and include some bad batches. Even though, as long as we can show:

1. page 197 will not remove too many good batches and will not include too many bad batches;
2. the bad batches included in C^* will not be significant

then we can show that the $\hat{x}$ returned in page 197 is a reasonable estimation for μ.

The Structure Of C^*: C^* is the largest subset of batches with confidence interval intersection. The confidence intervals of all the good batches intersect under event $\bigcap_{j \in \mathcal{G}} \mathcal{E}_j$, thus C^* should be at least as large as $\mathcal{G}$, thus it is not possible to remove too many good batches. Furthermore, we can also show that we will significantly reduce the total number of samples. Later on, we can show that the statistical rate will not be affected too much. We make these ideas precise below.

Under event $\bigcap_{j \in \mathcal{G}} \mathcal{E}_j$,

$$\mu \in \bigcap_{j \in \mathcal{G}} I_j, \tag{C.22}$$

where I_j is the confidence interval defined in page 197. Thus $\bigcap_{j \in \mathcal{G}} I_j \neq \varnothing$.

Because C^* maximizes

$$\max_{C \text{ s.t. } \varnothing \neq \bigcap_{j \in C} I_j} |C|, \tag{C.23}$$

we know $|C^*| \geq |\mathcal{G}| = (1-\alpha)m$. Furthermore, C^* can include at most αm batches, this means C^* includes at least $(1-2\alpha m)$ good batches. Formally:

$$|C^* \cap \mathcal{G}| = |C^* \setminus \mathcal{B}| \geq |C^*| - |\mathcal{B}| \geq (1-2\alpha)m. \tag{C.24}$$

Now we show C^* is not losing too much information, i.e. $\sum_{j \in C^*} \tilde{n}_j \geq \frac{1}{2} \sum_{j \in [m]} \tilde{n}_j$. By definition of n^{cut}, there are at least $2\alpha m + 1$ batches in $[m]$ such that $\tilde{n}_j = n^{\text{cut}}$. Because C^* removes at more αm batches, there are at least $\alpha m + 1$ batches in C^* such that $\tilde{n}_j = n^{\text{cut}}$. I.e.

$$\left|\left\{j \in C^* : \tilde{n}_j = n^{\text{cut}}\right\}\right| = \left|\left\{j \in [m] : \tilde{n}_j = n^{\text{cut}}\right\}\right| - \left|\left\{j \in [m] \setminus C^* : \tilde{n}_j = n^{\text{cut}}\right\}\right| \tag{C.25}$$

$$\geq \left|\left\{j \in [m] : \tilde{n}_j = n^{\text{cut}}\right\}\right| - |[m] \setminus C^*| \tag{C.26}$$

$$\geq 2\alpha m + 1 - \alpha m = \alpha m + 1 \tag{C.27}$$

This means the information loss $\sum_{j \in [m] \setminus \mathcal{G}} \tilde{n}_j$ can be bounded by $\sum_{j \in C^*} \tilde{n}_j$, formally:

$$2 \sum_{j \in C^*} \tilde{n}_j - \sum_{j \in [m]} \tilde{n}_j = \sum_{j \in C^*} \tilde{n}_j + \sum_{j \in C^*} \tilde{n}_j - \sum_{j \in [m] \cap C^*} \tilde{n}_j - \sum_{j \in [m] \setminus C^*} \tilde{n}_j \tag{C.28}$$

$$= \sum_{j \in C^*} \tilde{n}_j - \sum_{j \in [m] \setminus C^*} \tilde{n}_j \geq (\alpha m + 1) n^{\text{cut}} - \alpha m n^{\text{cut}} \geq 0 \tag{C.29}$$

Thus we have:

$$\sum_{j \in C^*} \tilde{n}_j \geq \frac{1}{2} \sum_{j \in [m]} \tilde{n}_j. \tag{C.30}$$

Bad Batches In C^*: In order for a bad batch i to survive in C^*, its confidence interval I_i must intersect with each good batch's confidence interval in C^*. In particular, I_i must intersect with the good batch in C^* with the largest $\tilde{n}_j$. By definition, there are at least $\alpha m + 1$ good batches with $\tilde{n}_j = n^{\text{cut}}$. Because C^*

excludes at most αm good batches, there is at least one good batch (denote by j^*), s.t. $\tilde{n}_{j^*} = n^{\text{cut}}$.

Thus $\forall j \in C^* \cap \mathcal{B}$, $I_i \cap I_{j^*} \neq \varnothing$. This means, there exists some point x, s.t. $x \in I_i \cap I_{j^*}$, thus

$$|\hat{\mu}_i - \hat{\mu}_{j^*}| \leq |\hat{\mu}_i - x| + |x - \hat{\mu}_{j^*}| \tag{C.31}$$

$$\leq \frac{\sigma}{\sqrt{n_i}}\sqrt{2\log\frac{2m}{\delta}} + \epsilon + \frac{\sigma}{\sqrt{n_{j^*}}}\sqrt{2\log\frac{2m}{\delta}} + \epsilon \tag{C.32}$$

$$\leq \left(\frac{1}{\sqrt{\tilde{n}_i}} + \frac{1}{\sqrt{n^{\text{cut}}}}\right)\sigma\sqrt{2\log\frac{2m}{\delta}} + 2\epsilon. \tag{C.33}$$

Furthermore, under event $\bigcap_{j\in\mathcal{G}} \mathcal{E}_j$,

$$|\hat{\mu}_{j^*} - \mu| \leq \frac{\sigma}{\sqrt{n_{j^*}}}\sqrt{2\log\frac{2m}{\delta}} + \epsilon \leq \frac{\sigma}{\sqrt{n^{\text{cut}}}}\sqrt{2\log\frac{2m}{\delta}} + \epsilon \tag{C.34}$$

By triangular inequality, $\hat{\mu}_i$ will not be too far away from μ:

$$|\hat{\mu}_i - \mu| \leq |\hat{\mu}_i - \hat{\mu}_{j^*}| + |\hat{\mu}_{j^*} - \mu| = \left(\frac{1}{\sqrt{\tilde{n}_i}} + \frac{2}{\sqrt{n^{\text{cut}}}}\right)\sigma\sqrt{2\log\frac{2m}{\delta}} + 3\epsilon \tag{C.35}$$

Error Decomposition: As mentioned earlier, we can decompose the estimation of $\hat{\mu}$ returned by page 197 by: statistical error (with potential information loss), term $\mathscr{A}_1$ in page 205; error coming from including bad batches, term $\mathscr{A}_2$ in page 205; error coming from removing good batches, term $\mathscr{A}_3$ in page 205. Specifically:

$$|\hat{\mu} - \mu| = \frac{1}{\sum_{j\in C^*}\tilde{n}_j}\left|\sum_{j\in C^*}\tilde{n}_j(\hat{\mu}_j - \mu)\right| \tag{C.36}$$

$$= \frac{1}{\sum_{j\in C^*}\tilde{n}_j}\left|\left(\sum_{j\in\mathcal{G}} + \sum_{j\in C^*\cap\mathcal{B}} - \sum_{j\in\mathcal{G}\setminus C^*}\right)\tilde{n}_j(\hat{\mu}_j - \mu)\right| \tag{C.37}$$

$$\leq \frac{1}{\sum_{j\in C^*}\tilde{n}_j}\left(\left|\sum_{j\in\mathcal{G}}\tilde{n}_j(\hat{\mu}_j - \mu)\right| + \left|\sum_{j\in C^*\cap\mathcal{B}}\tilde{n}_j(\hat{\mu}_j - \mu)\right| + \left|\sum_{j\in\mathcal{G}\setminus C^*}\tilde{n}_j(\hat{\mu}_j - \mu)\right|\right) \tag{C.38}$$

$$\text{(this is by triangular inequality)} \tag{C.39}$$

$$=:\mathscr{A}_1+\mathscr{A}_2+\mathscr{A}_3 \tag{C.40}$$

We can bound the first term $\mathscr{A}_1$ by page 203 under event $\mathcal{E}_{wa}$:

$$\mathscr{A}_1 = \frac{1}{\sum_{j\in C^*}\tilde{n}_j}\left|\sum_{j\in\mathcal{G}}\tilde{n}_j(\hat{\mu}_j-\mu)\right| = \frac{\sum_{j\in\mathcal{G}}\tilde{n}_j}{\sum_{j\in C^*}\tilde{n}_j}\left|\frac{1}{\sum_{j\in\mathcal{G}}\tilde{n}_j}\sum_{j\in\mathcal{G}}\tilde{n}_j(\hat{\mu}_j-\mu)\right| \tag{C.41}$$

$$= \frac{\sum_{j\in\mathcal{G}}\tilde{n}_j}{\sum_{j\in C^*}\tilde{n}_j}\left|\frac{1}{\sum_{j\in\mathcal{G}}\tilde{n}_j}\sum_{j\in\mathcal{G}}\tilde{n}_j\hat{\mu}_j-\mu\right| \tag{C.42}$$

$$\le \frac{\sum_{j\in\mathcal{G}}\tilde{n}_j}{\sum_{j\in C^*}\tilde{n}_j}\left(\frac{\sigma}{\sqrt{\sum_{j\in\mathcal{G}}\tilde{n}_j}}\sqrt{2\log\frac{2}{\delta}}+\epsilon\right) \quad \text{(By event } \mathcal{E}_{wa}\text{)} \tag{C.43}$$

$$= \frac{\sqrt{\sum_{j\in\mathcal{G}}\tilde{n}_j}}{\sum_{j\in C^*}\tilde{n}_j}\sigma\sqrt{2\log\frac{2}{\delta}}+\frac{\sum_{j\in\mathcal{G}}\tilde{n}_j}{\sum_{j\in C^*}\tilde{n}_j}\epsilon \tag{C.44}$$

$$\le 2\frac{\sqrt{\sum_{j\in\mathcal{G}}\tilde{n}_j}}{\sum_{j\in[m]}\tilde{n}_j}\sigma\sqrt{2\log\frac{2}{\delta}}+\frac{\sum_{j\in\mathcal{G}}\tilde{n}_j}{\sum_{j\in C^*}\tilde{n}_j}\epsilon \quad \text{(By page 203)} \tag{C.45}$$

$$\le 2\frac{\sqrt{\sum_{j\in[m]}\tilde{n}_j}}{\sum_{j\in[m]}\tilde{n}_j}\sigma\sqrt{2\log\frac{2}{\delta}}+\frac{\sum_{j\in\mathcal{G}}\tilde{n}_j}{\sum_{j\in C^*}\tilde{n}_j}\epsilon \quad \text{(By } \mathcal{G}\subseteq[m]\text{)} \tag{C.46}$$

$$= \frac{2}{\sqrt{\sum_{j\in[m]}\tilde{n}_j}}\sigma\sqrt{2\log\frac{2}{\delta}}+\frac{\sum_{j\in\mathcal{G}}\tilde{n}_j}{\sum_{j\in C^*}\tilde{n}_j}\epsilon \tag{C.47}$$

By page 204, we can bound the second term $\mathscr{A}_2$ by:

$$\mathscr{A}_2 = \frac{1}{\sum_{j\in C^*}\tilde{n}_j}\left|\sum_{j\in C^*\cap\mathcal{B}}\tilde{n}_j(\hat{\mu}_j-\mu)\right| \le \frac{1}{\sum_{j\in C^*}\tilde{n}_j}\sum_{j\in C^*\cap\mathcal{B}}\tilde{n}_j\left|\hat{\mu}_j-\mu\right| \tag{C.48}$$

$$\text{(By triangular ineq)} \tag{C.49}$$

$$\le \frac{1}{\sum_{j\in C^*}\tilde{n}_j}\sum_{j\in C^*\cap\mathcal{B}}\tilde{n}_j\left(\left(\frac{1}{\sqrt{\tilde{n}_j}}+\frac{2}{\sqrt{n^{\text{cut}}}}\right)\sigma\sqrt{2\log\frac{2m}{\delta}}+3\epsilon\right) \quad \text{(By page 204)} \tag{C.50}$$

$$\le \frac{1}{\sum_{j\in C^*}\tilde{n}_j}\sum_{j\in C^*\cap\mathcal{B}}\left(\sqrt{\tilde{n}_j}+\frac{2\tilde{n}_j}{\sqrt{n^{\text{cut}}}}\right)\sigma\sqrt{2\log\frac{2m}{\delta}}+\frac{\sum_{j\in C^*\cap\mathcal{B}}\tilde{n}_j}{\sum_{j\in C^*}\tilde{n}_j}3\epsilon \tag{C.51}$$

$$\leq \frac{1}{\sum_{j\in C^*}\tilde{n}_j}\sum_{j\in C^*\cap\mathcal{B}} 3\sqrt{n^{\text{cut}}}\sigma\sqrt{2\log\frac{2m}{\delta}} + \frac{\sum_{j\in C^*\cap\mathcal{B}}\tilde{n}_j}{\sum_{j\in C^*}\tilde{n}_j}3\epsilon \quad (\text{By } \tilde{n}_j \leq n^{\text{cut}}) \tag{C.52}$$

$$\leq \frac{3\alpha m\sqrt{n^{\text{cut}}}}{\sum_{j\in C^*}\tilde{n}_j}\sigma\sqrt{2\log\frac{2m}{\delta}} + \frac{\sum_{j\in C^*\cap\mathcal{B}}\tilde{n}_j}{\sum_{j\in C^*}\tilde{n}_j}3\epsilon \quad (C^* \text{ includes at most } \alpha m \text{ bad batches}) \tag{C.53}$$

We can bound the third term $\mathscr{A}_3$ by:

$$\mathscr{A}_3 = \frac{1}{\sum_{j\in C^*}\tilde{n}_j}\left|\sum_{j\in\mathcal{G}\setminus C^*}\tilde{n}_j(\hat{\mu}_j-\mu)\right| \leq \frac{1}{\sum_{j\in C^*}\tilde{n}_j}\sum_{j\in\mathcal{G}\setminus C^*}\tilde{n}_j\left|\hat{\mu}_j-\mu\right| \tag{C.54}$$

(By triangular ineq) (C.55)

$$\leq \frac{1}{\sum_{j\in C^*}\tilde{n}_j}\sum_{j\in\mathcal{G}\setminus C^*}\tilde{n}_j\left(\frac{\sigma}{\sqrt{n_j}}\sqrt{2\log\frac{2m}{\delta}}+\epsilon\right) \quad \left(\text{By event } \bigcap_{j\in\mathcal{G}}\mathcal{E}_j\right) \tag{C.56}$$

$$\leq \frac{1}{\sum_{j\in C^*}\tilde{n}_j}\sum_{j\in\mathcal{G}\setminus C^*}\tilde{n}_j\left(\frac{\sigma}{\sqrt{\tilde{n}_j}}\sqrt{2\log\frac{2m}{\delta}}+\epsilon\right) \tag{C.57}$$

$$= \frac{1}{\sum_{j\in C^*}\tilde{n}_j}\sum_{j\in\mathcal{G}\setminus C^*}\sigma\sqrt{\tilde{n}_j}\sqrt{2\log\frac{2m}{\delta}} + \frac{\sum_{j\in\mathcal{G}\setminus C^*}\tilde{n}_j}{\sum_{j\in C^*}\tilde{n}_j}\epsilon \tag{C.58}$$

$$\leq \frac{\alpha m\sqrt{n^{\text{cut}}}}{\sum_{j\in C^*}\tilde{n}_j}\sigma\sqrt{2\log\frac{2m}{\delta}} + \frac{\sum_{j\in\mathcal{G}\setminus C^*}\tilde{n}_j}{\sum_{j\in C^*}\tilde{n}_j}\epsilon \tag{C.59}$$

(Because C^* excludes at most αm good batches and $\tilde{n}_j \leq n^{\text{cut}}$) (C.60)

Note that the above upper bounds for $\mathscr{A}_2$ and $\mathscr{A}_3$ are still valid even if some of the $\tilde{n}_j$'s are zero.

In conclusion, we can bound the estimation error by:

$$|\hat{\mu}-\mu| \leq \mathscr{A}_1+\mathscr{A}_2+\mathscr{A}_3 \tag{C.61}$$

$$\leq \left(\frac{2}{\sqrt{\sum_{j\in[m]}\tilde{n}_j}}\sigma\sqrt{2\log\frac{2}{\delta}} + \frac{\sum_{j\in\mathcal{G}}\tilde{n}_j}{\sum_{j\in C^*}\tilde{n}_j}\epsilon\right) \tag{C.62}$$

$$+ \left(\frac{3\alpha m\sqrt{n^{\text{cut}}}}{\sum_{j\in C^*}\tilde{n}_j}\sigma\sqrt{2\log\frac{2m}{\delta}} + \frac{\sum_{j\in C^*\cap\mathcal{B}}\tilde{n}_j}{\sum_{j\in C^*}\tilde{n}_j}3\epsilon\right) \tag{C.63}$$

$$+\left(\frac{\alpha m\sqrt{n^{\text{cut}}}}{\sum_{j\in C^*}\tilde{n}_j}\sigma\sqrt{2\log\frac{2m}{\delta}}+\frac{\sum_{j\in\mathcal{G}\setminus C^*}\tilde{n}_j}{\sum_{j\in C^*}\tilde{n}_j}\epsilon\right) \tag{C.64}$$

$$=\frac{2}{\sqrt{\sum_{j\in[m]}\tilde{n}_j}}\sigma\sqrt{2\log\frac{2}{\delta}}+\frac{4\alpha m\sqrt{n^{\text{cut}}}}{\sum_{j\in C^*}\tilde{n}_j}\sigma\sqrt{2\log\frac{2m}{\delta}} \tag{C.65}$$

$$+\frac{\left(\sum_{j\in\mathcal{G}}+\sum_{j\in C^*\cap\mathcal{B}}\right)\tilde{n}_j}{\sum_{j\in C^*}\tilde{n}_j}\epsilon+\frac{\sum_{j\in C^*\cap\mathcal{B}}\tilde{n}_j}{\sum_{j\in C^*}\tilde{n}_j}2\epsilon+\frac{\sum_{j\in\mathcal{G}\setminus C^*}\tilde{n}_j}{\sum_{j\in C^*}\tilde{n}_j}\epsilon \tag{C.66}$$

$$\leq\frac{2}{\sqrt{\sum_{j\in[m]}\tilde{n}_j}}\sigma\sqrt{2\log\frac{2}{\delta}}+\frac{4\alpha m\sqrt{n^{\text{cut}}}}{\sum_{j\in C^*}\tilde{n}_j}\sigma\sqrt{2\log\frac{2m}{\delta}} \tag{C.67}$$

$$+\frac{\sum_{j\in[m]}\tilde{n}_j}{\sum_{j\in C^*}\tilde{n}_j}\epsilon+\frac{\alpha m n^{\text{cut}}}{\sum_{j\in C^*}\tilde{n}_j}2\epsilon+\frac{\alpha m n^{\text{cut}}}{\sum_{j\in C^*}\tilde{n}_j}\epsilon \tag{C.68}$$

$$(\text{By } \mathcal{G}\cup(C^*\cap\mathcal{B})\subseteq[m],\ |C^*\cap\mathcal{B}|\leq\alpha m,\ |\mathcal{G}\setminus C^*|\leq\alpha m) \tag{C.69}$$

$$\leq\frac{2}{\sqrt{\sum_{j\in[m]}\tilde{n}_j}}\sigma\sqrt{2\log\frac{2}{\delta}}+\frac{8\alpha m\sqrt{n^{\text{cut}}}}{\sum_{j\in[m]}\tilde{n}_j}\sigma\sqrt{2\log\frac{2m}{\delta}}+5\epsilon \tag{C.70}$$

$$(\text{By page 203 and page 203}) \tag{C.71}$$

C.3 Proof of Theorem 5.5.1

By following standard regret decomposition for UCB type of algorithm (see (Jin et al., 2020b)), under the event that the estimation error of the Bellman operator is bounded by bonus terms, we can decompose the regret by:

1. the cumulative bonus term occurred in the trajectories of each good agent
2. a term that can be easier bounded by Azuma-Hoeffding's inequalities.

By page 199 and replacing page 198 with a variant for martingale, we can show the event mentioned above happens with high probability. Unlike standard regret bound for tabular settings, we cannot directly use the telescoping series to estimate the cumulative bonuses. Instead, we first need to show that because each good agent is using the same policy in every episode, their trajectories have a lot of

overlaps, meaning the (s, a, h) counts of all good agents do not differ by too much. Given that, we can simplify the bound in page 199 and use the telescoping series.

We start by restating page 61:

Theorem C.3.1 (Regret bound, page 61)**.** *If* $\alpha \leq \frac{1}{3}\left(1 - \frac{1}{m}\right)$*, for all* $\delta < \frac{1}{4}$*, with probability at least* $1 - 3\delta$*:*

$$\sum_{k=1}^{K} \sum_{j \in \mathcal{G}} \left(V_1^*(s_1) - V_1^{\hat{\pi}^k}(s_1)\right) = \tilde{O}\left((1 + \alpha\sqrt{m})SH^2\sqrt{AKm \log \frac{1}{\delta}}\right) \tag{C.72}$$

We first give the high-level idea of our proof:

1. We give an analysis under the intersection of three "good events":
   - event $\mathcal{E}$: the estimation error of Bellman operator is upper-bounded by bonus (See page 215, page 216);
   - event $\mathcal{E}_{\text{even}}$: if the total count $\sum_{j \in \mathcal{G}} N_h^{j,k}(s, a, h)$ on some (s, a, h) is large, then the counts of each agent differ by at most 2 times (See page 230, page 230);
   - event $\mathcal{E}_{\text{Azmua}}$: an error term in the regret decomposition is bounded by Azmua-Hoeffding bound.
2. Under event $\mathcal{E}$, we can decompose the regret into two terms (see page 225, page 227):
   - a martingale with bounded difference which is controlled by Hoeffding bound under event $\mathcal{E}_{\text{Azmua}}$;
   - the cumulative bonus term, which can be bounded by telescoping series under event $\mathcal{E}_{\text{even}}$.

We use $\bar{Q}_h^k$, $\hat{Q}_h^k$, $\hat{\pi}_h^k$, $\hat{V}_h^k$, $\hat{\mathbb{B}}_h^k$, Γ_h^k to denote the variables used in the k-th episode. When synchronization happens in episode k, those variables are the updated ones after the synchronization; when there is no synchronization in episode k, those variables remain the same as in the last episode. Let $N_h^{j,k}(s, a)$ be the counts on

(s, a, h) tuples in episode k after the counts update. Formally, We start by restating the data collection process and counts on (s, a, h) tuples of each good agent $j \in \mathcal{G}$: during the data collection process, we allow all of the agents to collect data together. In the k-th episode, agent j collects a multi-set of transition tuples using policy $d^{\hat{\pi}^k}$: $\left\{\left(s_h^{j,k}, a_h^{j,k}, r_h^{j,k}, s_{h+1}^{j,k}\right)\right\}_{h\in[H]}$.

$$D_{j,k} := \bigcup_{h\in[H]} D_{j,k}^h := \bigcup_{h\in[H]} \bigcup_{k'\le k} \left\{\left(s_h^{j,k'}, a_h^{j,k'}, r_h^{j,k'}, s_{h+1}^{j,k'}\right)\right\} \tag{C.73}$$

$N_h^{j,k}(s, a)$ is given by:

$$N_h^{j,k}(s,a) = \sum_{h=1}^{H} \sum_{(\tilde{s},\tilde{a},\tilde{r},\tilde{s}')\in D_{j,k}^h} \mathbf{1}\left\{(s,a) = (\tilde{s},\tilde{a})\right\} \tag{C.74}$$

We give the formal definition of good events below:

Definition C.3.1.

$$\mathcal{E}_{\text{Azmua}} := \Bigg\{ \sum_{k=1}^{K} \sum_{j\in\mathcal{G}} \sum_{h=1}^{H} \left(\mathbb{E}_{s'\sim P_h(\cdot|s_h^{j,k}, a_h^{j,k})}\left[\hat{V}_{h+1}^k(s') - V_{h+1}^{\hat{\pi}^k}(s')\right]\right. \tag{C.75}$$

$$\left. - \left(\hat{V}_{h+1}^k(s_{h+1}^{j,k}) - V_{h+1}^{\hat{\pi}^k}(s_{h+1}^{j,k})\right)\right) \le \sqrt{8mKH^3 \log\frac{2}{\delta}} \Bigg\} \tag{C.76}$$

$$\mathcal{E} := \left\{ \bigcap_{(s,a,h,k,f)\in\mathcal{S}\times\mathcal{A}\times H\times K\times[0,1]^{\mathcal{S}}} \left\{ \left| \left(\hat{\mathbb{B}}_h^k f\right)(s,a) - \left(\mathbb{B}_h f\right)(s,a) \right| \le \Gamma_h^k(s,a) \right\} \right\} \tag{C.77}$$

For any $(s, a, h, k) \in \mathcal{S} \times \mathcal{A} \times [H] \times [K]$, we define the following event:

$$\mathcal{E}_{\text{even}}(s,a,h,k) \tag{C.78}$$

$$:= \left\{ \text{if } \textstyle\sum_{j\in\mathcal{G}} N_h^{j,k}(s,a) \ge 400m \log\frac{2mKSAH}{\delta}, \text{ then } \max_{i,j\in\mathcal{G}} \frac{N_h^{j,k}(s,a)}{N_h^{i,k}(s,a)} \le 2 \right\} \tag{C.79}$$

And define

$$\mathcal{E}_{\text{even}} := \bigcap_{s,a,h,K} \mathcal{E}_{\text{even}}(s,a,h,k). \tag{C.80}$$

Proof of Theorem 5.5.1. By Azuma-Hoeffding inequality:

$$\mathbb{P}\left(\overline{\mathcal{E}_{\text{Azmua}}}\right) \leq \delta \tag{C.81}$$

Then by union bound: Lemma C.3.1 and Lemma C.3.7 together implies for all $0 < \delta < \frac{1}{4}$:

$$\mathbb{P}\left(\overline{\mathcal{E}} \cup \overline{\mathcal{E}_{\text{even}}} \cup \overline{\mathcal{E}_{\text{Azmua}}}\right) \leq \mathbb{P}\left(\overline{\mathcal{E}}\right) + \mathbb{P}\left(\overline{\mathcal{E}_{\text{even}}}\right) + \mathbb{P}\left(\overline{\mathcal{E}_{\text{Azmua}}}\right) \leq 3\delta \tag{C.82}$$

which means $\mathcal{E} \cap \mathcal{E}_{\text{even}} \cap \mathcal{E}_{\text{Azmua}}$ happens with probability at least $1 - 3\delta$.

We now upper bound the regret under event $\mathcal{E} \cap \mathcal{E}_{\text{even}} \cap \mathcal{E}_{\text{Azmua}}$. By Lemma C.3.6 we can decompose the regret by:

$$\sum_{k=1}^{K} \sum_{j \in \mathcal{G}} \left(V_1^*(s_1) - V_1^{\hat{\pi}^k}(s_1)\right) \tag{C.83}$$

$$\leq 2 \sum_{k=1}^{K} \sum_{j \in \mathcal{G}} \sum_{h=1}^{H} \Gamma_h^k(s_h^{j,k}, a_h^{j,k}) \tag{C.84}$$

$$+ \sum_{k=1}^{K} \sum_{j \in \mathcal{G}} \sum_{h=1}^{H} \left(\mathbb{E}_{s' \sim P_h(\cdot|s_h^k, a_h^k)}\left[\hat{V}_{h+1}^k(s') - V_{h+1}^{\hat{\pi}^k}(s')\right] - \left(\hat{V}_{h+1}^k(s_{h+1}^{j,k}) - V_{h+1}^{\hat{\pi}^k}(s_{h+1}^{j,k})\right)\right) \tag{C.85}$$

$$(\text{Under event } \mathcal{E}) \tag{C.86}$$

$$\leq 2 \sum_{k=1}^{K} \sum_{j \in \mathcal{G}} \sum_{h=1}^{H} \Gamma_h^k(s_h^{j,k}, a_h^{j,k}) + \sqrt{8mKH^3 \log \frac{2}{\delta}} \tag{C.87}$$

$$\left(\text{Under event } \mathcal{E}_{\text{Azmua}}\right) \tag{C.88}$$

We only need to upper bound the cumulative bonus. Suppose the policy is updated at the beginning of $k_0+1, k_1+1, k_2+1, \ldots, k_l+1$-th episodes, with the data collected in the first $k_0, k_1, k_2, \ldots, k_l$-th episodes, with $k_1 = 1$. To simplify the notation, we

define $k_0 = 0$, $k_{l+1} = K$.

For convenience, in the following, we use $N_h^k(s,a)$ to denote the total count on (s,a,h) tuples up to episode k over all good agents:

$$N_h^k(s,a) := \sum_{j \in \mathcal{G}} N_h^{j,k}(s,a), \tag{C.89}$$

where $N_h^0(s,a) = 0$. We can rearrange the cumulative bonus by summing over (s,a) pairs:

$$\sum_{k=1}^{K} \sum_{j \in \mathcal{G}} \sum_{h=1}^{H} \Gamma_h^k(s_h^{j,k}, a_h^{j,k}) = \sum_{h=1}^{H} \sum_{(s,a) \in \mathcal{S} \times \mathcal{A}} \sum_{t=1}^{l+1} \Gamma_h^{k_{t-1}+1}(s,a) \left(N_h^{k_t}(s,a) - N_h^{k_{t-1}}(s,a) \right) \tag{C.90}$$

When there are less than $(2\alpha m + 1)$ agents have coverage on some (s,a,h) tuple, the bonus term $\Gamma_h^k(s,a)$ is trivially set to be $H - h + 1$. In the following, we show that under the event $\mathcal{E}_{\text{even}}$, in (C.90), for each (s,a,h) tuple, there are at most $2N_0$ bonus term such that $\Gamma_h(s,a) = H - h + 1$, where

$$N_0 := 400m \log \frac{2mKSAH}{\delta}. \tag{C.91}$$

For any (s,a,h), let $l_0(s,a,h)$ be such that:

$$N_h^{k_{l_0(s,a,h)-1}}(s,a) < N_0 \leq N_h^{k_{l_0(s,a,h)}}(s,a). \tag{C.92}$$

This means when running the policy update at episode $k_{l_0(s,a,h)} + 1$, the total counts for (s,a,h), i.e. $N_h^{k_{l_0(s,a,h)}}(s,a)$, is larger than N_0. For any $k \geq k_{l_0(s,a,h)}$, we have

$$\sum_{j \in \mathcal{G}} N_h^{j,k}(s,a) = N_h^k(s,a) \geq N_h^{k_{l_0(s,a,h)}}(s,a) \geq N_0. \tag{C.93}$$

By definition of $\mathcal{E}_{\text{even}}$, for any $k \geq k_{l_0(s,a,h)}$

$$\max_{i,j \in \mathcal{G}} \frac{N_h^{j,k}(s,a)}{N_h^{i,k}(s,a)} \leq 2 \tag{C.94}$$

this means for any $k \geq k_{l_0(s,a,h)}$, $N_h^{j,k}(s,a) > 0, \forall j \in \mathcal{G}$, meaning all of the good agents have coverage on (s,a,h), this means there are at least $(1-\alpha)m \geq 2\alpha m + 1$ agents have coverage, and thus:

- Trivial bonus can only happens at $k \leq k_{l_0(s,a,h)}$, i.e.

  $$\Gamma_h^k(s,a) = H - h + 1 \text{ only if } k \leq k_{l_0(s,a,h)}. \tag{C.95}$$

  Furthermore, in the algorithm, the agents synchronize and update their policy when or before any (s,a,h) counts for a good agent doubles. I.e.: for all $(s,a,h,j,i) \in \mathcal{S} \times \mathcal{A} \times [H] \times \mathcal{G} \times [l]$:

  $$N_h^{k_t}(s,a) \leq 2N_h^{k_{t-1}}(s,a) \tag{C.96}$$

  This means

  $$N_h^{k_{l_0(s,a,h)}}(s,a) \leq 2N_h^{k_{l_0(s,a,h)-1}}(s,a) < 2N_0. \tag{C.97}$$

  Thus for each (s,a,h) tuple, there are at most $2N_0$ bonus terms such that $\Gamma_h(s,a) = H - h + 1$.

- for any $k \geq k_{l_0(s,a,h)} + 1$

  $$\Gamma_h^k(s,a) = \frac{6}{SAHKm} + \frac{2(H-h+1)}{\sqrt{\sum_{j\in[m]} \tilde{N}_h^{j,k-1}(s,a)}} \sqrt{2\log\frac{2(SAHKm)^{3S}}{\delta}} \tag{C.98}$$

  $$+ \frac{8\alpha m\sqrt{N_h^{\text{cut},k-1}(s,a)}}{\sum_{j\in[m]} \tilde{N}_h^{j,k-1}(s,a)}(H-h+1)\sqrt{2\log\frac{2m(SAHKm)^{3S}}{\delta}} \tag{C.99}$$

Where $N_h^{\text{cut},k-1}(s,a)$ is the $(2\alpha m+1)$-largest among $\left\{N_h^{j,k-1}(s,a)\right\}$ and

$$\tilde{N}_h^{j,k-1}(s,a) = \max\left(N_h^{\text{cut},k-1}(s,a), N_h^{j,k-1}(s,a)\right); \tag{C.100}$$

For any $k-1 \geq k_{l_0(s,a,h)}$, $\max_{i,j\in\mathcal{G}} \frac{N_h^{j,k-1}(s,a)}{N_h^{i,k-1}(s,a)} \leq 2$ implies $\forall j$, $\tilde{N}_h^{j,k-1}(s,a) \geq \frac{1}{2}N_h^{j,k-1}(s,a)$ and $\tilde{N}_h^{j,k-1}(s,a) \geq \frac{1}{2}N_h^{\text{cut},k-1}(s,a)$.

This means for any $k \geq k_{l_0(s,a,h)}+1$

$$\frac{1}{\sqrt{\sum_{j\in[m]}\tilde{N}_h^{j,k-1}(s,a)}} \leq \frac{\sqrt{2}}{\sqrt{\sum_{j\in[m]}N_h^{j,k-1}(s,a)}} = \frac{\sqrt{2}}{\sqrt{N_h^{k-1}(s,a)}} \tag{C.101}$$

$$\frac{m\sqrt{N_h^{\text{cut},k-1}(s,a)}}{\sum_{j\in[m]}\tilde{N}_h^{j,k-1}(s,a)} = \frac{\sqrt{m}\sqrt{\sum_{j\in[m]}N_h^{\text{cut},k-1}(s,a)}}{\sum_{j\in[m]}\tilde{N}_h^{j,k-1}(s,a)} \tag{C.102}$$

$$\leq \frac{\sqrt{m}\sqrt{2\sum_{j\in[m]}\tilde{N}_h^{j,k-1}(s,a)}}{\sum_{j\in[m]}\tilde{N}_h^{j,k-1}(s,a)} \leq \frac{2\sqrt{m}}{\sqrt{N_h^{k-1}(s,a)}} \tag{C.103}$$

Thus

$$\Gamma_h^k(s,a) \leq \frac{4+16\sqrt{2}\alpha\sqrt{m}}{\sqrt{N_h^{k-1}(s,a)}} H\sqrt{\log\frac{2m(SAHKm)^{3S}}{\delta}} + \frac{6}{SAHKm} \tag{C.104}$$

We are now ready to bound the cumulative bonus:

$$\sum_{k=1}^{K}\sum_{j\in\mathcal{G}}\sum_{h=1}^{H}\Gamma_h^k(s_h^{j,k},a_h^{j,k}) = \sum_{h=1}^{H}\sum_{(s,a)\in\mathcal{S}\times\mathcal{A}}\sum_{t=1}^{l+1}\Gamma_h^{k_{t-1}+1}(s,a)\left(N_h^{k_t}(s,a)-N_h^{k_{t-1}}(s,a)\right) \tag{C.105}$$

$$= \sum_{h=1}^{H}\sum_{(s,a)\in\mathcal{S}\times\mathcal{A}}\left(\sum_{t=1}^{l_0(s,a,h)}\Gamma_h^{k_{t-1}+1}(s,a)\left(N_h^{k_t}(s,a)-N_h^{k_{t-1}}(s,a)\right)\right. \tag{C.106}$$

$$\left. + \sum_{t=l_0(s,a,h)+1}^{l+1}\Gamma_h^{k_{t-1}+1}(s,a)\left(N_h^{k_t}(s,a)-N_h^{k_{t-1}}(s,a)\right)\right) \tag{C.107}$$

$$\leq \sum_{h=1}^{H} \sum_{(s,a)\in\mathcal{S}\times\mathcal{A}} \left(\sum_{t=1}^{l_0(s,a,h)} \Gamma_h^{k_{t-1}+1}(s,a) \left(N_h^{k_t}(s,a) - N_h^{k_{t-1}}(s,a) \right) \right. \tag{C.108}$$

$$+ \sum_{t=l_0(s,a,h)+1}^{l+1} \frac{4+16\sqrt{2}\alpha\sqrt{m}}{\sqrt{N_h^{k-1}(s,a)}} H \sqrt{\log \frac{2m(SAHKm)^{3S}}{\delta}} \left(N_h^{k_t}(s,a) - N_h^{k_{t-1}}(s,a) \right) \tag{C.109}$$

$$\left. + \sum_{t=l_0(s,a,h)+1}^{l+1} \frac{6}{SAHKm} \left(N_h^{k_t}(s,a) - N_h^{k_{t-1}}(s,a) \right) \right) \tag{C.110}$$

(By page 213) (C.111)

$$=: \mathscr{A}_1 + \mathscr{A}_2 + \mathscr{A}_3. \tag{C.112}$$

By (C.95) and (C.97),

$$\mathscr{A}_1 \leq SAH^2 N_h^{k_{l_0(s,a,h)}}(s,a) \leq 2SAH^2 N_0. \tag{C.113}$$

Because $k_{l+1} = K$,

$$\mathscr{A}_3 \leq \frac{6}{SAHKm} \sum_{h=1}^{H} \sum_{(s,a)\in\mathcal{S}\times\mathcal{A}} N_h^K(s,a) = \frac{6}{SA} \tag{C.114}$$

By (C.96),

$$\sum_{t=l_0(s,a,h)+1}^{l+1} \frac{N_h^{k_t}(s,a) - N_h^{k_{t-1}}(s,a)}{\sqrt{N_h^{k_{t-1}}(s,a)}} \tag{C.115}$$

$$\leq (\sqrt{2}+1) \sum_{t=l_0(s,a,h)+1}^{l+1} \frac{N_h^{k_t}(s,a) - N_h^{k_{t-1}}(s,a)}{\sqrt{N_h^{k_t}(s,a)} + \sqrt{N_h^{k_{t-1}}(s,a)}} \tag{C.116}$$

$$= (\sqrt{2}+1) \sum_{t=l_0(s,a,h)+1}^{l+1} \left(\sqrt{N_h^{k_t}(s,a)} - \sqrt{N_h^{k_{t-1}}(s,a)} \right) \leq (\sqrt{2}+1) \sqrt{N_h^K(s,a)} \tag{C.117}$$

By Cauchy–Schwarz inequality,

$$\sum_{(s,a)\in\mathcal{S}\times\mathcal{A}} \sqrt{N_h^K(s,a)} \leq \sqrt{\sum_{(s,a)\in\mathcal{S}\times\mathcal{A}} 1 \sum_{(s,a)\in\mathcal{S}\times\mathcal{A}} N_h^K(s,a)} = \sqrt{SAKm} \tag{C.118}$$

Thus

$$\mathscr{A}_2 \leq (\sqrt{2}+1)(4+16\sqrt{2}\alpha\sqrt{m})H^2\sqrt{SAKm}\sqrt{\log\frac{2m(SAHKm)^{3S}}{\delta}} \tag{C.119}$$

$$= O\left((1+\alpha\sqrt{m})H^2S\sqrt{AKm}\sqrt{\log\frac{SAHKm}{\delta}}\right) \tag{C.120}$$

Thus

$$\mathscr{A}_1+\mathscr{A}_2+\mathscr{A}_3 \leq O\left((1+\alpha\sqrt{m})H^2S\sqrt{AKm}\sqrt{\log\frac{SAHKm}{\delta}}\right) \tag{C.121}$$

$$+ O\left(SAH^2m\log\frac{2mKSAH}{\delta}\right) \tag{C.122}$$

In conclusion:

$$\sum_{k=1}^{K}\sum_{j\in\mathcal{G}}\left(V_1^*(s_1)-V_1^{\hat{\pi}^k}(s_1)\right) \leq 2\sum_{k=1}^{K}\sum_{j\in\mathcal{G}}\sum_{h=1}^{H}\Gamma_h^k(s_h^{j,k},a_h^{j,k})+\sqrt{8mKH^3\log\frac{2}{\delta}} \tag{C.123}$$

$$= \tilde{O}\left((1+\alpha\sqrt{m})SH^2\sqrt{AKm\log\frac{1}{\delta}}\right) \tag{C.124}$$

■

The Good Event $\mathcal{E}$

We first show that our bonus is upper confidence bound for the estimated Bellman operator. Recall that our bonus term used in k-th episode is calculated based on the data collected in the first $k-1$-episodes. The bonus is given by:

- If $|j\in[m]: N_h^{j,k-1}(s,a)>0| < 2\alpha m+1$

$$\Gamma_h^k(s,a) = H-h+1; \tag{C.125}$$

- If $|j \in [m] : N_h^{j,k-1}(s,a) > 0| \geq 2\alpha m + 1$

$$\Gamma_h^k(s,a) := \frac{6}{SAHKm} + \frac{2(H-h+1)}{\sqrt{\sum_{j\in[m]} \tilde{N}_h^{j,k-1}(s,a)}} \sqrt{2\log\frac{2(SAHKm)^{3S}}{\delta}} \tag{C.126}$$

$$+ \frac{8\alpha m \sqrt{N_h^{\mathrm{cut},k-1}(s,a)}}{\sum_{j\in[m]} \tilde{N}_h^{j,k-1}(s,a)} (H-h+1) \sqrt{2\log\frac{2m(SAHKm)^{3S}}{\delta}} \tag{C.127}$$

Where $N_h^{\mathrm{cut},k-1}(s,a)$ is the $(2\alpha m+1)$-largest among $\left\{N_h^{j,k-1}(s,a)\right\}$ and

$$\tilde{N}_h^{j,k-1}(s,a) = \max\left(N_h^{\mathrm{cut},k-1}(s,a), N_h^{j,k-1}(s,a)\right). \tag{C.128}$$

To be precise:

Lemma C.3.1 (Valid bonus)**.** *Let $\mathcal{E}$ be the following event:*

$$\mathcal{E} = \left\{ \bigcap_{(s,a,h,k,f)\in\mathcal{S}\times\mathcal{A}\times H\times K\times[0,1]^{\mathcal{S}}} \left\{ \left| \left(\hat{\mathbb{B}}_h^k f\right)(s,a) - \left(\mathbb{B}_h f\right)(s,a) \right| \leq \Gamma_h^k(s,a) \right\} \right\} \tag{C.129}$$

Then, we have

$$\mathbb{P}(\mathcal{E}) \geq 1-\delta \tag{C.130}$$

To show that $\mathcal{E}$ is a high probability event, we seek to utilize the result of page 55. Since there are two obstacles, we need to make some modifications:

1. Because the transition tuples are collected sequentially, they are no longer i.i.d., which means page 198 does not hold trivially. To resolve this, we use the concentration of martingale (see page 219);

2. Event $\mathcal{E}$ shows the concentration property of $\hat{\mathbb{B}}$ holds uniformly for infinitely many f's. Thus a direct union bound does not apply. Instead, we need to use a cover number argument for all possible f's, which is standard (see (Jin et al., 2020b)).

Proof of Lemma C.3.1. Let $\mathcal{E}'$ be the following event:

$$\mathcal{E}' = \left\{ N_h^{\text{cut},k-1}(s,a) > 0 \right\}. \tag{C.131}$$

In the following, we decompose $\mathcal{E}$ by:

$$\mathcal{E} = \left(\mathcal{E} \cap \overline{\mathcal{E}'}\right) \cup (\mathcal{E} \cap \mathcal{E}') \tag{C.132}$$

and bound $\mathbb{P}(\mathcal{E})$ by law of total probability.

If $N_h^{\text{cut},k-1}(s,a) = 0$, because $\left(\hat{\mathbb{B}}_h^k f\right)(s,a) = 0$ and $(\mathbb{B}_h f)(s,a) \leq H - h + 1$, with probability 1, $\forall (s,a,h,k,f) \in \mathcal{S} \times \mathcal{A} \times H \times K \times [0,1]^{\mathcal{S}}$,

$$\left| \left(\hat{\mathbb{B}}_h^k f\right)(s,a) - (\mathbb{B}_h f)(s,a) \right| \leq \Gamma_h^k(s,a) \tag{C.133}$$

This means

$$\mathbb{P}\left(\mathcal{E} \cap \overline{\mathcal{E}'}\right) = \mathbb{P}\left(\mathcal{E} | \overline{\mathcal{E}'}\right) \mathbb{P}\left(\overline{\mathcal{E}'}\right) = \mathbb{P}\left(\overline{\mathcal{E}'}\right) \tag{C.134}$$

If $N_h^{\text{cut},k-1}(s,a) > 0$, we use a covering number argument and union bound to bound the probability of event $\mathcal{E}$.

Consider $\mathcal{V}_\epsilon := \left\{ \frac{1}{\lceil 1/\epsilon \rceil}, \frac{2}{\lceil 1/\epsilon \rceil}, \ldots, \frac{H\lceil 1/\epsilon \rceil}{\lceil 1/\epsilon \rceil} \right\}^{\mathcal{S}}$, an ϵ cover of $[0,H]^{\mathcal{S}}$, in the sense of ∞-norm. We can bound the cover number by $|\mathcal{V}_\epsilon| \leq \left(H\left(\frac{1}{\epsilon} + 1\right)\right)^S$. This means $\forall f \in [0,H]^{\mathcal{S}}$, we can find an $V_f \in \mathcal{V}_\epsilon$, s.t. $\|f - V_f\|_\infty := \max_{x \in \mathcal{S}} |f(x) - V_f(x)| \leq \epsilon$. In other words,

$$[0,H]^{\mathcal{S}} = \bigcup_{f_\epsilon \in \mathcal{V}_\epsilon} \{ f : \|f - f_\epsilon\|_\infty \leq \epsilon \}. \tag{C.135}$$

Importantly, unlike the model-based method without bad agents, our $\hat{\mathbb{B}}$ is not a linear operator, meaning we cannot trivially upper bound

$$\left| \left(\hat{\mathbb{B}}_h^k f\right)(s,a) - \left(\hat{\mathbb{B}}_h^k V_f\right)(s,a) \right|$$

in the cover number argument. Instead, we need to use the continuity of error bound of our robust mean estimation Algorithm 16, meaning as long as each data point collected by each agent is not perturbed too much, then the estimation error

bound does not increase too much.

Recall that in Algorithm 6, at episode k, if the agents decide to synchronize, then at each step h, given any function f, the clean agents will calculate the empirical mean for

$$\left\{r + f(s') : (s, a, r, s') \in D_h^{j,k}\right\}. \tag{C.136}$$

Let f_ϵ be an element in $\mathcal{V}_\epsilon$, s.t. $\|f_\epsilon - f\|_\infty \leq \epsilon$, this means set (C.136) is a perturbed version (by at most ϵ) of

$$\left\{r + f_\epsilon(s') : (s, a, r, s') \in D_h^{j,k}\right\}. \tag{C.137}$$

This means given an $f_\epsilon \in \mathcal{V}_\epsilon$, for any f, s.t. $\|f - f_\epsilon\|_\infty \leq \epsilon$, Algorithm 16 can be used to robustly estimate $\left(\mathbb{B}_h f_\epsilon\right)(s, a)$, given set (C.136). Furthermore, choosing $\epsilon = \frac{1}{SAHKm}$, by Lemma C.3.2, Lemma C.3.3 and page 199, given any s, a, h, k, f_ϵ, and any f, s.t. $\|f - f_\epsilon\|_\infty \leq \epsilon$, with probability at least $1 - \frac{\delta}{(SAHKm)^{3S}/(2mK)}$,

$$\left|\left(\hat{\mathbb{B}}_h^k f\right)(s, a) - \left(\mathbb{B}_h f_\epsilon\right)(s, a)\right| \leq \Gamma_h^k(s, a) - \frac{1}{SAHKm}. \tag{C.138}$$

We can bound the $\left|\left(\hat{\mathbb{B}}_h^k f\right)(s, a) - \left(\mathbb{B}_h f\right)(s, a)\right|$ by:

$$\begin{aligned}
&\left|\left(\hat{\mathbb{B}}_h^k f\right)(s, a) - \left(\mathbb{B}_h f\right)(s, a)\right| && \text{(C.139)}\\
\leq &\left|\left(\hat{\mathbb{B}}_h^k f\right)(s, a) - \left(\mathbb{B}_h f_\epsilon\right)(s, a)\right| + \left|\left(\mathbb{B}_h f_\epsilon\right)(s, a) - \left(\mathbb{B}_h f\right)(s, a)\right| && \text{(C.140)}\\
\leq &\left|\left(\hat{\mathbb{B}}_h^k f\right)(s, a) - \left(\mathbb{B}_h f_\epsilon\right)(s, a)\right| + \frac{1}{SAHKm} && \text{(C.141)}
\end{aligned}$$

Then

$$\begin{aligned}
&\mathbb{P}\left(\bigcup_{s,a,h,k,f}\left\{\left|\left(\hat{\mathbb{B}}_h^k f\right)(s, a) - \left(\mathbb{B}_h f\right)(s, a)\right| > \Gamma_h^k(s, a)\right\}\right) && \text{(C.142)}\\
\leq &\sum_{s,a,h,k}\mathbb{P}\left(\bigcup_{f\in[0,H]^{\mathcal{S}}}\left\{\left|\left(\hat{\mathbb{B}}_h^k f\right)(s, a) - \left(\mathbb{B}_h f\right)(s, a)\right| > \Gamma_h^k(s, a)\right\}\right) && \text{(C.143)}
\end{aligned}$$

$$\leq \sum_{s,a,h,k} \mathbb{P}\left(\bigcup_{f_\epsilon \in \mathcal{V}_\epsilon} \bigcup_{f:\|f-f_\epsilon\|_\infty \leq \epsilon} \left\{ \left| \left(\hat{\mathbb{B}}_h^k f\right)(s,a) - \left(\mathbb{B}_h f_\epsilon\right)(s,a) \right| + \frac{1}{SAHKm} > \Gamma_h^k(s,a) \right\} \right) \tag{C.144}$$

$$\leq \sum_{s,a,h,k} \sum_{f_\epsilon \in \mathcal{V}_\epsilon} \mathbb{P}\left(\bigcup_{f:\|f-f_\epsilon\|_\infty \leq \epsilon} \left\{ \left| \left(\hat{\mathbb{B}}_h^k f\right)(s,a) - \left(\mathbb{B}_h f_\epsilon\right)(s,a) \right| + \frac{1}{SAHKm} > \Gamma_h^k(s,a) \right\} \right) \tag{C.145}$$

$$\leq SAHK(H(1+HSAKm))^S \frac{\delta}{(SAHKm)^{3S}/(2mK)} \leq \delta \tag{C.146}$$

This means

$$\mathbb{P}\left(\mathcal{E} \cap \mathcal{E}'\right) = \mathbb{P}\left(\mathcal{E}|\mathcal{E}'\right)\mathbb{P}\left(\mathcal{E}'\right) \geq (1-\delta)\,\mathbb{P}\left(\mathcal{E}'\right) \geq \mathbb{P}\left(\mathcal{E}'\right) - \delta \tag{C.147}$$

In conclusion,

$$\mathbb{P}\left(\mathcal{E}\right) = \mathbb{P}\left(\mathcal{E} \cap \mathcal{E}'\right) + \mathbb{P}\left(\mathcal{E} \cap \overline{\mathcal{E}'}\right) \geq \mathbb{P}\left(\mathcal{E}\right) + \mathbb{P}\left(\mathcal{E}'\right) - \delta = 1 - \delta. \tag{C.148}$$

■

Concentration Of Estimation From Good Agents

Lemma C.3.2. *Let:*

$$\left(\hat{\mathbb{B}}_h^{j,k} f\right)(s,a) := \frac{1}{N_h^{j,k}(s,a)} \sum_{(s,a,r,s') \in D_h^{j,k}} r + f(s'), \tag{C.149}$$

where we define $\frac{0}{0} = 0$*. For any* $f : \mathcal{S} \mapsto [H]$*, and for any* $(s,a,h,k) \in \mathcal{S}\times\mathcal{A}\times[H]\times[K]$ *with probability at least* $1-\delta/2$*,* $\mathcal{E}_{conc-seq}(s,a,h,k)$ *happens, where*

$$\mathcal{E}_{conc-seq}(s,a,h,k) = \bigcap_{j \in \mathcal{G}} \mathcal{E}_{c-seq}(s,a,h,j,k), \tag{C.150}$$

and

$$\mathcal{E}_{c-seq}(s,a,h,j,k) := \left\{ \left| \left(\hat{\mathbb{B}}_h^{j,k} f\right)(s,a) - \left(\mathbb{B}_h f\right)(s,a) \right| \leq \frac{H-h+1}{\sqrt{\tilde{N}_h^{j,k}(s,a)}} \sqrt{2\log\frac{4Km}{\delta}} \right\} \tag{C.151}$$

Proof of Lemma C.3.2. We use the martingale stopping time argument in Lemma 4.3 of (Jin et al., 2018).

For each fixed $(s,a,h,j) \in \mathcal{S}\times\mathcal{A}\times[H]\times\mathcal{G}$: for all $t \in [K]$, define

$$\mathcal{F}_t := \sigma\left(\bigcup_{t'\leq t} \bigcup_{j\in[m]} \left\{ \left(s_h^{j,t'}, a_h^{j,t'}, r_h^{j,t'}, s_{h+1}^{j,t'}\right) \right\}_{h=1}^{H} \right). \tag{C.152}$$

Let

$$X_t = \sum_{(s,a,r,s')\in D_h^{j,t}} \left(r + f(s') - \left(\mathbb{B}_h f\right)(s,a)\right) \tag{C.153}$$

Then $\{(\mathcal{F}_t, X_t)\}_{t=1}^K$ is a martingale. One observation is $X_{t_1} = X_{t_2}$ if agent j did not visit (s,a,h) in $t_1+1, t_1+2, \ldots, t_2$-th episodes. Thus we can use the stopping time idea to shorten the martingale sequence.

Define the following sequence of t_i's: $t_0 := 0$,

$$t_i := \min\left(\left\{ t' \in [K] : t' > t_{i-1} \text{ and } (s_h^{j,t'}, a_h^{j,t'}) = (s,a) \right\} \cup \{K+1\} \right). \tag{C.154}$$

Intuitively, t_i is the episode when (s,a,h) is visited by agent j for the i-th time. If agent j visit (s,a,h) for less than i times, then $t_i = K+1$. By definition, t_i is a stopping time w.r.t. $\{\mathcal{F}_t\}_{t=1}^K$.

By optional sampling theorem, $\{(\mathcal{F}_{t_i}, X_{t_i})\}_{i=1}^K$ is a martingale.

By Azuma-Hoeffding's inequality: for any $\tau \leq K$

$$\mathbb{P}\left(|X_{t_\tau}| \geq \beta\right) \leq 2\exp\left(-\frac{2\beta^2}{4\tau(H-h+1)^2}\right) \tag{C.155}$$

Let $\frac{\delta}{2mK} = 2\exp\left(-\frac{2\beta^2}{4\tau(H-h+1)^2}\right)$, we get: for any (s,a,h,j), for any $\tau \leq K$, with

probability at least $1 - \frac{\delta}{2mK}$:

$$\left| \sum_{(s,a,r,s') \in D_h^{j,t_\tau}} (r + f(s') - (\mathbb{B}_h f)(s,a)) \right| < \sqrt{\tau}(H - h + 1)\sqrt{2 \log \frac{4mK}{\delta}}. \tag{C.156}$$

By union bound, for any (s, a, h, j), with probability at least $1 - \frac{\delta}{mK}$, for any $\tau \leq K$:

$$\left| \sum_{(s,a,r,s') \in D_h^{j,t_\tau}} (r + f(s') - (\mathbb{B}_h f)(s,a)) \right| < \sqrt{\tau}(H - h + 1)\sqrt{2 \log \frac{4mK}{\delta}}. \tag{C.157}$$

This means for any (s, a, h, j, k) and any $\tau \leq k$

$$\mathbb{P}\left(\overline{\mathcal{E}_{c-seq}(s,a,h,j,k)} \middle| N_h^{j,k}(s,a) = \tau\right) \tag{C.158}$$

$$\leq \mathbb{P}\left(\left| \left(\hat{\mathbb{B}}_h^{j,k} f\right)(s,a) - (\mathbb{B}_h f)(s,a) \right| \geq \frac{H - h + 1}{\sqrt{N_h^{j,k}(s,a)}} \sqrt{2 \log \frac{4Km}{\delta}} \middle| N_h^{j,k}(s,a) = \tau \right) \tag{C.159}$$

$$\leq \frac{\delta}{mK} \tag{C.160}$$

Thus

$$\mathbb{P}\left(\overline{\mathcal{E}_{c-seq}(s,a,h,j,k)}\right) = \sum_{\tau=0}^{k} \mathbb{P}\left(\overline{\mathcal{E}_{c-seq}(s,a,h,j,k)} \middle| N_h^{j,k}(s,a) = \tau\right) \mathbb{P}\left(N_h^{j,k}(s,a) = \tau\right) \tag{C.161}$$

$$\leq \frac{\delta}{2mK} \tag{C.162}$$

By union bound

$$\mathbb{P}\left(\mathcal{E}_{conc-seq}(s,a,h,k)\right) \geq 1 - \frac{\delta}{2}. \tag{C.163}$$

■

Lemma C.3.3. *Let:*

$$\left(\hat{\mathbb{B}}_h^{j,k} f\right)(s,a) := \frac{1}{N_h^{j,k}(s,a)} \sum_{(s,a,r,s') \in D_h^{j,k}} r + f(s'), \tag{C.164}$$

$$\left(\hat{\mathbb{B}}_h^{\mathcal{G},k} f\right)(s,a) := \frac{1}{\sum_{j\in\mathcal{G}} \tilde{N}_h^{j,k}(s,a)} \sum_{j\in\mathcal{G}} \tilde{N}_h^{j,k}(s,a) \left(\hat{\mathbb{B}}_h^{j,k} f\right)(s,a), \tag{C.165}$$

where we define $\frac{0}{0} = 0$*. For any* $f : \mathcal{S} \mapsto [H]$*, with probability at least* $1-\delta/2$*,* $\mathcal{E}_{ct}(s,a,h,k)$ *happens, where*

$$\mathcal{E}_{ct}(s,a,h,k) := \left\{ \left| \left(\hat{\mathbb{B}}_h^{\mathcal{G},k} f\right)(s,a) - \left(\mathbb{B}_h f\right)(s,a) \right| \le \frac{H-h+1}{\sqrt{\sum_{j\in\mathcal{G}} \tilde{N}_h^{j,k}(s,a)}} \sqrt{2\log\frac{4mK}{\delta}} \right\} \tag{C.166}$$

Proof of Lemma C.3.3. During the data-collecting process, the agents are allowed to collect data simultaneously. For analysis purposes, we artificially order the data in the following sequence:

$$E^{1,1}, E^{2,1}, \ldots, E^{m,1}, E^{1,2}, \ldots, E^{m,2}, \ldots, E^{1,K}, \ldots, E^{m,K} \tag{C.167}$$

where $E^{j,k} := \left\{ \left(s_h^{j,k}, a_h^{j,k}, r_h^{j,k}, s_{h+1}^{j,k}\right) \right\}_{h=1}^{H}$. Let

$$\mathcal{F}_t = \sigma\left(\bigcup_{j,k \text{ s.t. } m(k-1)+j \le t} E^{j,k} \right). \tag{C.168}$$

Then $\{\mathcal{F}_t\}_{t=0}^{mK}$ forms a valid filtration. Let $\left\{ \{\gamma_{j,k}\}_{j\in[m]} \right\}_{k\in[K]}$ be a fixed set of scalar, s.t. $0 \le \gamma_{j,k} \le 1$, for all j,k.

For each fixed $(s,a,h) \in \mathcal{S}\times\mathcal{A}\times[H]$: for all $t \in [mK]$, Let

$$X_t = \sum_{(s,a,r,s') \in \bigcup_{(j,k)\in\mathcal{G}\times[K]} \text{ s.t. } m(k-1)+j\le t} \gamma_{j,k} \left(r + f(s') - \left(\mathbb{B}_h f\right)(s,a)\right) \tag{C.169}$$

Then $\{(\mathcal{F}_t, X_t)\}_{t=1}^{mK}$ is a martingale. As we can see, if good agent j did not visit (s,a,h)

in episode k, then $X_{m(k-1)+j} = X_{m(k-1)+j-1}$ a.s. Thus we can use the stopping time idea to shorten the martingale sequence.

Define the following functions to map from sequence index to agent index and episode index:

$$\mathcal{J}(t) := t - m\left(\lceil t/m \rceil - 1\right), \quad \mathcal{K}(t) := \lceil t/m \rceil \tag{C.170}$$

For any $n_1, \ldots, n_m$, define the following sequence of t_i's: $t_0 := 0$,

$$t_i := \min\Big(\Big\{t' \in [mK] : t' > t_{i-1} \text{ and } (s_h^{\mathcal{J}(t'),\mathcal{K}(t')}, a_h^{\mathcal{J}(t'),\mathcal{K}(t')}) = (s,a) \tag{C.171}$$

$$\text{and for all } j \le \mathcal{J}(t'), N_h^{j,\mathcal{K}(t')} \le n_j; j > \mathcal{J}(t'), N_h^{j,\mathcal{K}(t')-1} \le n_j\Big\} \cup \{K+1\}\Big). \tag{C.172}$$

Intuitively, t_i is the episode when (s,a,h) is visited in sequence page 222 for the i-th time. And for all j, agent j have not collected n_j (s,a,h) tuples. If (s,a,h) is visited for less than i times or there exists agent j visiting (s,a,h) more than n_j times, then $t_i = K+1$. By definition, t_i is a stopping time w.r.t. $\{\mathcal{F}_t\}_{t=1}^{mK}$.

In particular, let n_{cut} be the $(2\alpha m + 1)$th-largest of all n_j's and $\tilde{n}_j = \min(n_{\text{cut}}, n_j)$. We choose $\gamma_{j,k} := \frac{\tilde{n}_j}{n_j} \le 1$.

By optional sampling theorem, $\{(\mathcal{F}_{t_i}, X_{t_i})\}_{i=1}^{mK}$ is a martingale.

By Azuma-Hoeffding's inequality: for any $\tau := \sum_{j\in[m]} n_j \le mK$

$$\mathbb{P}\left(|X_{t_\tau}| \ge \beta\right) \le 2\exp\left(-\frac{2\beta^2}{4(H-h+1)^2 \sum_{t=1}^{\tau} \gamma_{\mathcal{J}(t),\mathcal{K}(t)}^2}\right) \tag{C.173}$$

Let $\frac{\delta}{2mK} = 2\exp\left(-\frac{2\beta^2}{4(H-h+1)^2 \sum_{t=1}^{\tau} \gamma_{\mathcal{J}(t),\mathcal{K}(t)}^2}\right)$, we get: for any (s,a,h), for any $\tau \le mK$, with probability at least $1 - \frac{\delta}{2mK}$:

$$|X_{t_\tau}| < \sqrt{\sum_{t=1}^{\tau} \gamma_{\mathcal{J}(t),\mathcal{K}(t)}^2}(H-h+1)\sqrt{2\log\frac{4mK}{\delta}}. \tag{C.174}$$

By union bound, for any (s,a,h), with probability at least $1-\frac{\delta}{2}$, for any $\tau \le mK$:

$$|X_{t_\tau}| < \sqrt{\sum_{t=1}^{\tau} \gamma^2_{\mathcal{J}(t),\mathcal{K}(t)}}(H-h+1)\sqrt{2\log\frac{4mK}{\delta}}. \tag{C.175}$$

This means for any $(s,a,h,k) \in \mathcal{S}\times\mathcal{A}\times[H]\times[K]$ and any $\tau \le mk$

$$\mathbb{P}\left(\overline{\mathcal{E}_{ct}(s,a,h,k)}\middle| N_h^{j,k}(s,a)=n_j,\forall j\right) \tag{C.176}$$

$$\le\mathbb{P}\left(\left|\left(\hat{\mathbb{B}}_h^{\mathcal{G},k}f\right)(s,a)-\left(\mathbb{B}_h f\right)(s,a)\right| \ge \frac{(H-h+1)\sqrt{\sum_{t=1}^{\tau}\frac{\tilde{N}_h^{\mathcal{J}(t),\mathcal{K}(t)}(s,a)}{N_h^{\mathcal{J}(t),\mathcal{K}(t)}(s,a)}}}{\sum_{j\in\mathcal{G}}\tilde{N}_h^{j,k}(s,a)}\right. \tag{C.177}$$

$$\left.\cdot\sqrt{2\log\frac{4SAHmK^2}{\delta}}\middle| N_h^{j,k}(s,a)=n_j,\forall j\right) \tag{C.178}$$

$$\le\mathbb{P}\left(\left|\left(\hat{\mathbb{B}}_h^{\mathcal{G},k}f\right)(s,a)-\left(\mathbb{B}_h f\right)(s,a)\right| \ge \frac{(H-h+1)\sqrt{\sum_{t=1}^{\tau}\left(\frac{\tilde{N}_h^{\mathcal{J}(t),\mathcal{K}(t)}(s,a)}{N_h^{\mathcal{J}(t),\mathcal{K}(t)}(s,a)}\right)^2}}{\sum_{j\in\mathcal{G}}\tilde{N}_h^{j,k}(s,a)}\right. \tag{C.179}$$

$$\left.\cdot\sqrt{2\log\frac{4SAHmK^2}{\delta}}\middle| N_h^{j,k}(s,a)=n_j,\forall j\right) \tag{C.180}$$

$$\le\frac{\delta}{2} \quad \left(\text{By } \gamma_{j,k}=\frac{\tilde{N}_h^{j,k}(s,a)}{N_h^{j,k}(s,a)}\right) \tag{C.181}$$

Thus

$$\mathbb{P}\left(\overline{\mathcal{E}_{ct}(s,a,h,k)}\right) = \sum_{(n_1,\ldots,n_m)\in[K]^m}\mathbb{P}\left(\overline{\mathcal{E}_{ct}(s,a,h,k)}\middle| N_h^{j,k}(s,a)=n_j,\forall j\right) \tag{C.182}$$

$$\mathbb{P}\left(N_h^{j,k}(s,a)=n_j,\forall j\right) \tag{C.183}$$

$$\le\frac{\delta}{2} \tag{C.184}$$

■

The Regret Decomposition For UCB Style Algorithm

We follow the regret decomposition strategy in (Jin et al., 2020b) under event $\mathcal{E}$, i.e. the estimation error for the Bellman operator is bounded by the bonus term.

The estimated Bellman operator can be used to approximate the Q function:

Lemma C.3.4. *Under event $\mathcal{E}$, for any $(s,a,h,k) \in \mathcal{S} \times \mathcal{A} \times H \times K$, and any policy π'*

$$\left|\left(\hat{\mathbb{B}}_h^k \hat{V}_{h+1}^k\right)(s,a) - Q_h^{\pi'}(s,a) - \mathbb{E}_{s' \sim P_h(\cdot|s,a)}\left[\hat{V}_{h+1}^k(s') - V_{h+1}^{\pi'}(s')\right]\right| \leq \Gamma_h^k(s,a) \quad \text{(C.185)}$$

Proof of Lemma C.3.4.

$$\begin{aligned}
&\left|\left(\hat{\mathbb{B}}_h^k \hat{V}_{h+1}^k\right)(s,a) - Q_h^{\pi'}(s,a) - \mathbb{E}_{s' \sim P_h(\cdot|s,a)}\left[\hat{V}_{h+1}^k(s') - V_{h+1}^{\pi'}(s')\right]\right| && \text{(C.186)}\\
\leq &\left|\left(\hat{\mathbb{B}}_h^k \hat{V}_{h+1}^k\right)(s,a) - \left(\mathbb{B}_h \hat{V}_{h+1}^k\right)(s,a)\right| && \text{(C.187)}\\
&+ \left|\left(\mathbb{B}_h \hat{V}_{h+1}^k\right)(s,a) - \left(\mathbb{B}_h V_{h+1}^{\pi'}\right)(s,a) - \mathbb{E}_{s' \sim P_h(\cdot|s,a)}\left[\hat{V}_{h+1}^k(s') - V_{h+1}^{\pi'}(s')\right]\right| && \text{(C.188)}\\
&\left(\text{By triangular inequality and the fact that } \left(\mathbb{B}_h V_{h+1}^{\pi'}\right)(s,a) = Q_h^{\pi'}(s,a).\right) && \text{(C.189)}\\
\leq &\Gamma_h^k(s,a) && \text{(C.190)}\\
&(\text{We can bound the first term by the definition of event } \mathcal{E}, && \text{(C.191)}\\
&\text{and the second term is zero by the definition of Bellman operator.}) && \text{(C.192)}
\end{aligned}$$

■

Under event $\mathcal{E}$ we can upper bound the value function and Q function of the optimal policy by the estimated value function and Q function of policy $\hat{\pi}^k$:

Lemma C.3.5 (Optimism)**.** *Under event $\mathcal{E}$, $\forall s,a,h,k$:*

$$\hat{Q}_h^k(s,a) \geq Q_h^*(s,a), \quad \hat{V}_h^k(s) \geq V_h^*(s) \quad \text{(C.193)}$$

Proof of Lemma C.3.5. We prove this by induction on h. Before that, note that, for any h,k,s, if

$$\hat{Q}_h^k(s,a) \geq Q_h^*(s,a), \quad \forall a \quad \text{(C.194)}$$

then because $\hat{\pi}^k$ is chosen by maximizing $\hat{Q}_h^k(s,a)$, we know

$$\hat{V}_h^k(s) = \max_a \hat{Q}_h^k(s,a) \geq \hat{Q}_h^k(s,\pi_h^*(a)) \geq Q_h^*(s,\pi_h^*(a)) = V_h^*(s) \tag{C.195}$$

This means for any h, k, s:

$$\left\{\forall a, \hat{Q}_h^k(s,a) \geq Q_h^*(s,a)\right\} \implies \left\{\hat{V}_h^k(s) \geq V_h^*(s)\right\} \tag{C.196}$$

We now begin our induction:

- For the base case, our goal is to show for any s, a, k, in the last step H,

$$\hat{Q}_H^k(s,a) \geq Q_H^*(s,a), \quad \hat{V}_H^k(s) \geq V_H^*(s) \tag{C.197}$$

  First note that $\hat{V}_{H+1} = V_{H+1}^* = 0$. By Lemma C.3.4 and choose $\pi' = \pi^*$,

$$\left|\left(\hat{\mathbb{B}}_H^k \hat{V}_{H+1}^k\right)(s,a) - Q_h^*(s,a)\right| \leq \Gamma_H^k(s,a) \tag{C.198}$$

  By definition of $\hat{Q}_H^k(s,a)$, and the fact that $Q_h^*(s,a)$ only contains the reward at step H, which is bounded by 1:

$$\hat{Q}_H^k(s,a) = \min\left(\left(\hat{\mathbb{B}}_H^k \hat{V}_{H+1}^k\right)(s,a) + \Gamma_H^k(s,a), 1\right) \geq Q_H^*(s,a) \tag{C.199}$$

  By (C.196), $\hat{V}_H^k(s) \geq V_H^*(s), \forall s$.

- Suppose for any s, a, k, the statement holds for step $h+1$, i.e.

$$\hat{Q}_{h+1}^k(s,a) \geq Q_{h+1}^*(s,a), \quad \hat{V}_{h+1}^k(s) \geq V_{h+1}^*(s) \tag{C.200}$$

  our goal is to show $\forall s, a, k$:

$$\hat{Q}_h^k(s,a) \geq Q_h^*(s,a), \quad \hat{V}_h^k(s) \geq V_h^*(s) \tag{C.201}$$

$$\left(\hat{\mathbb{B}}_h^k \hat{V}_{h+1}^k\right)(s,a) + \Gamma_h^k(s,a) \tag{C.202}$$

$$\geq \left(\hat{\mathbb{B}}_h^k \hat{V}_{h+1}^k\right)(s,a) + \left|\left(\hat{\mathbb{B}}_h^k \hat{V}_{h+1}^k\right)(s,a) - Q_h^*(s,a) - \mathbb{E}_{s'\sim P_h(\cdot|s,a)}\left[\hat{V}_{h+1}^k(s') - V_{h+1}^*(s')\right]\right| \tag{C.203}$$

(By Lemma C.3.4 and let $\pi' = \pi^*$) (C.204)

$$\geq Q_h^*(s,a) + \mathbb{E}_{s'\sim P_h(\cdot|s,a)}\left[\hat{V}_{h+1}^k(s') - V_{h+1}^*(s')\right] \tag{C.205}$$

(By triangular inequality) (C.206)

$$\geq Q_h^*(s,a) \tag{C.207}$$

$\left(\forall s, \hat{V}_{h+1}^k(s') \geq V_{h+1}^*(s')\right.$ by page 226$\left.\right)$ (C.208)

By definition of Q function $Q_h^*(s,a) \leq H - h + 1$. Thus

$$\hat{Q}_h^k(s,a) = \min\left(\left(\hat{\mathbb{B}}_h^k \hat{V}_{h+1}^k\right)(s,a) + \Gamma_h^k(s,a), H - h + 1\right) \geq Q_h^*(s,a) \tag{C.209}$$

By (C.196), $\hat{V}_h^k(s) \geq V_h^*(s), \forall s$.

■

We are now ready to prove the regret decomposition lemma:

Lemma C.3.6. *Under good event $\mathcal{E}$:*

$$\sum_{k=1}^{K}\sum_{j\in\mathcal{G}}\left(V_1^*(s_1) - V_1^{\hat{\pi}^k}(s_1)\right) \tag{C.210}$$

$$\leq 2\sum_{k=1}^{K}\sum_{j\in\mathcal{G}}\sum_{h=1}^{H}\Gamma_h^k(s_h^{j,k}, a_h^{j,k}) \tag{C.211}$$

$$+ \sum_{k=1}^{K}\sum_{j\in\mathcal{G}}\sum_{h=1}^{H}\left(\mathbb{E}_{s'\sim P_h(\cdot|s_h^k,a_h^k)}\left[\hat{V}_{h+1}^k(s') - V_{h+1}^{\hat{\pi}^k}(s')\right] - \left(\hat{V}_{h+1}^k(s_{h+1}^{j,k}) - V_{h+1}^{\hat{\pi}^k}(s_{h+1}^{j,k})\right)\right) \tag{C.212}$$

Proof of Lemma C.3.6. We start by showing the decomposition of regret after step h in one episode of a single agent: by Lemma C.3.4 and Lemma C.3.5, under event $\mathcal{E}$,

for any s, k, h

$$V_h^*(s) - V_h^{\hat{\pi}^k}(s) \le \hat{V}_h^{\hat{\pi}^k}(s) - V_h^{\hat{\pi}^k}(s) \quad \text{(By Lemma C.3.5)} \tag{C.213}$$
$$= \hat{Q}_h^k(s, \hat{\pi}_h^k(s)) - Q_h^{\hat{\pi}^k}(s, \hat{\pi}_h^k(s)) \tag{C.214}$$
$$\le \left(\hat{\mathbb{B}}_h^k \hat{V}_{h+1}^k\right)(s,a) + \Gamma_h^k(s,a) - Q_h^{\hat{\pi}^k}(s, \hat{\pi}_h^k(s)) \quad \left(\text{By definition of } \hat{Q}_h^k\right) \tag{C.215}$$
$$\le \left|\left(\hat{\mathbb{B}}_h^k \hat{V}_{h+1}^k\right)(s, \hat{\pi}_h^k(s)) - Q_h^{\hat{\pi}^k}(s, \hat{\pi}_h^k(s)) - \mathbb{E}_{s' \sim P_h(\cdot|s, \hat{\pi}_h^k(s))}\left[\hat{V}_{h+1}^k(s') - V_{h+1}^{\hat{\pi}^k}(s')\right]\right| \tag{C.216}$$
$$+ \left|Q_h^{\hat{\pi}^k}(s, \hat{\pi}_h^k(s)) + \mathbb{E}_{s' \sim P_h(\cdot|s, \hat{\pi}_h^k(s))}\left[\hat{V}_{h+1}^k(s') - V_{h+1}^{\hat{\pi}^k}(s')\right]\right| \tag{C.217}$$
$$+ \Gamma_h^k(s,a) - Q_h^{\hat{\pi}^k}(s, \hat{\pi}_h^k(s)) \tag{C.218}$$
$$\text{(By using triangular inequality on the first term)} \tag{C.219}$$
$$\le \Gamma_h^k(s, \hat{\pi}_h^k(s)) + Q_h^{\hat{\pi}^k}(s, \hat{\pi}_h^k(s)) + \mathbb{E}_{s' \sim P_h(\cdot|s, \hat{\pi}_h^k(s))}\left[\hat{V}_{h+1}^k(s') - V_{h+1}^{\hat{\pi}^k}(s')\right] \tag{C.220}$$
$$+ \Gamma_h^k(s,a) - Q_h^{\hat{\pi}^k}(s, \hat{\pi}_h^k(s)) \tag{C.221}$$
$$\Big(\text{The first term is by using Lemma C.3.4 with } \pi' = \hat{\pi}^k, \tag{C.222}$$
$$\text{the term inside the absolute in the second is non-negative by Lemma C.3.5)} \tag{C.223}$$
$$= 2\Gamma_h^k(s, \hat{\pi}_h^k(s)) + \mathbb{E}_{s' \sim P_h(\cdot|s, \hat{\pi}_h^k(s))}\left[\hat{V}_{h+1}^k(s') - V_{h+1}^{\hat{\pi}^k}(s')\right] \tag{C.224}$$

This indeed gives a recursive formula: for any trajectory $\left\{(s_h^k, a_h^k, r_h^k, s_{h+1}^k)\right\}_{h \in [H]}$

$$\hat{V}_h^{\hat{\pi}^k}(s_h^k) - V_h^{\hat{\pi}^k}(s_h^k) \tag{C.225}$$
$$\le 2\Gamma_h^k(s_h^k, \hat{\pi}_h^k(s_h^k)) + \mathbb{E}_{s' \sim P_h(\cdot|s_h^k, \hat{\pi}_h^k(s_h^k))}\left[\hat{V}_{h+1}^k(s') - V_{h+1}^{\hat{\pi}^k}(s')\right] \tag{C.226}$$
$$= \hat{V}_{h+1}^k(s_h^k) - V_{h+1}^{\hat{\pi}^k}(s_h^k) + 2\Gamma_h^k(s_h^k, \hat{\pi}_h^k(s_h^k)) \tag{C.227}$$
$$+ \left(\mathbb{E}_{s' \sim P_h(\cdot|s_h^k, \hat{\pi}_h^k(s_h^k))}\left[\hat{V}_{h+1}^k(s') - V_{h+1}^{\hat{\pi}^k}(s')\right] - \left(\hat{V}_{h+1}^k(s_h^k) - V_{h+1}^{\hat{\pi}^k}(s_h^k)\right)\right) \tag{C.228}$$

Then, we can show the regret decomposition in one episode of a single agent by recursion:

for any trajectory $\left\{(s_h^k, a_h^k, r_h^k, s_{h+1}^k)\right\}_{h \in [H]}$ collected by a clean agent under policy

$\hat{\pi}^k$:

$$V_1^*(s_1^k) - V_1^{\hat{\pi}^k}(s_1^k) \le \hat{V}_1^k(s_1^k) - V_1^{\hat{\pi}^k}(s_1^k) \tag{C.229}$$
$$\le \left(\hat{V}_2^k(s_2^k) - V_2^{\hat{\pi}^k}(s_2^k)\right) + 2\Gamma_1^k(s_1^k, a_1^k) \tag{C.230}$$
$$+ \left(\mathbb{E}_{s'\sim P_1(\cdot|s_1^k,a_1^k)}\left[\hat{V}_2^k(s') - V_2^{\hat{\pi}^k}(s')\right] - \left(\hat{V}_2^k(s_2^k) - V_2^{\hat{\pi}^k}(s_2^k)\right)\right) \tag{C.231}$$
$$\le \left(\hat{V}_3^k(s_3^k) - V_3^{\hat{\pi}^k}(s_3^k)\right) + \sum_{h=1}^{2} 2\Gamma_h^k(s_h^k, a_h^k) \tag{C.232}$$
$$+ \sum_{h=1}^{2}\left(\mathbb{E}_{s'\sim P_h(\cdot|s_h^k,a_h^k)}\left[\hat{V}_{h+1}^k(s') - V_{h+1}^{\hat{\pi}^k}(s')\right] - \left(\hat{V}_{h+1}^k(s_{h+1}^k) - V_{h+1}^{\hat{\pi}^k}(s_{h+1}^k)\right)\right) \tag{C.233}$$
$$\le \cdots \tag{C.234}$$
$$\le \sum_{h=1}^{H} 2\Gamma_h^k(s_h^k, a_h^k) \tag{C.235}$$
$$+ \sum_{h=1}^{H}\left(\mathbb{E}_{s'\sim P_h(\cdot|s_h^k,a_h^k)}\left[\hat{V}_{h+1}^k(s') - V_{h+1}^{\hat{\pi}^k}(s')\right] - \left(\hat{V}_{h+1}^k(s_{h+1}^k) - V_{h+1}^{\hat{\pi}^k}(s_{h+1}^k)\right)\right) \tag{C.236}$$

Now we are ready to show the total regret decomposition. For each episode, we can make the regret decomposition w.r.t. any trajectory collected by a clean agent following policy $\hat{\pi}^k$. For convenience, we specialize the trajectories to be exactly the ones that are collected by the good agents and are used to calculate the bonus terms. The purpose is, in the future, when we bound the regret, we need to bound the cumulative bonus used in the trajectory. By decomposing the regret w.r.t. the trajectory collected in the algorithm, it is naturally guaranteed that the (s, a, h) tuples that are collected a lot by the good agents have a lower bonus. This is because, with more data collected, we can narrow down the confidence interval and design small but still valid bonus terms.

Because in our MDP definition, the MDP has a deterministic initial distribution, meaning the good agents always have the same starting state:

$$\sum_{k=1}^{K}\sum_{j\in\mathcal{G}}\left(V_1^*(s_1) - V_1^{\hat{\pi}^k}(s_1)\right) = \sum_{k=1}^{K}\sum_{j\in\mathcal{G}}\left(V_1^*(s_1^{j,k}) - V_1^{\hat{\pi}^k}(s_1^{j,k})\right) \tag{C.237}$$

$$\leq 2\sum_{k=1}^{K}\sum_{j\in\mathcal{G}}\sum_{h=1}^{H}\Gamma_h^k(s_h^{j,k},a_h^{j,k}) \tag{C.238}$$

$$+\sum_{k=1}^{K}\sum_{j\in\mathcal{G}}\sum_{h=1}^{H}\left(\mathbb{E}_{s'\sim P_h(\cdot|s_h^{j,k},a_h^{j,k})}\left[\hat{V}_{h+1}^k(s')-V_{h+1}^{\hat{\pi}^k}(s')\right]-\left(\hat{V}_{h+1}^k(s_{h+1}^{j,k})-V_{h+1}^{\hat{\pi}^k}(s_{h+1}^{j,k})\right)\right) \tag{C.239}$$

■

Evenness Of Clean Agents

We need at least $(2\alpha m+1)$-agents to cover (s,a,h) in order to learn the Bellman operator properly. In this section, we show that the agents have "even" coverage on the visited (s,a,h) tuples in each (except a relatively small number) of the episodes. In the following we use $\tilde{m}:=(1-\alpha)m=|\mathcal{G}|$ to denote the number of good agents.

Formally, we have:

Lemma C.3.7 (Even coverage of good agent)**.** *For any $(s,a,h,k)\in\mathcal{S}\times\mathcal{A}\times[H]\times[K]$, we define the following event:*

$$\mathcal{E}_{even}(s,a,h,k):=\left\{\text{if }\sum_{j\in\mathcal{G}}N_h^{j,k}(s,a)\geq 400m\log\tfrac{2mKSAH}{\delta},\text{ then }\max_{i,j\in\mathcal{G}}\tfrac{N_h^{j,k}(s,a)}{N_h^{i,k}(s,a)}\leq 2\right\} \tag{C.240}$$

then, we have: for all $0<\delta<\frac{1}{4}$

$$\mathbb{P}\left(\bigcap_{(s,a,h,k)\in\mathcal{S}\times\mathcal{A}\times[H]\times[K]}\mathcal{E}_{even}(s,a,h,k)\right)\geq 1-2\delta \tag{C.241}$$

Remark C.3.1 (Intuition of the good event)**.** The event $\mathcal{E}_{\text{even}}(s,a,h,k)$ characterizes that: if in any episode k, a (s,a,h) tuple gets enough coverage from the clean agents, then the coverage of each agent are very close.

See proof of Lemma C.3.7 in Section C.3.

Proof of Lemma C.3.7

Proof of Lemma C.3.7 depends on the concentration of $N_h^{j,k}(s,a)$:

Lemma C.3.8 (Concentration of counts around empirical mean)**.** *For all* $0 < \delta < \frac{1}{4}$

$$\mathbb{P}\left(\bigcup_{s,a,h,k,j} \left\{ \left| N_h^{j,k}(s,a) - \frac{1}{|\mathcal{G}|} \sum_{j \in \mathcal{G}} N_h^{j,k}(s,a) \right| \right. \right. \tag{C.242}$$

$$\left. \left. > 18 \log \frac{2SAHmK}{\delta} + 4\sqrt{\log \frac{2SAHmK}{\delta}} \sqrt{\frac{1}{|\mathcal{G}|} \sum_{j \in \mathcal{G}} N_h^{j,k}(s,a)} \right\} \right) < 2\delta \tag{C.243}$$

Proof of Lemma C.3.8. See page 233. ■

Proof of Lemma C.3.7. Let

$$N_0 := 400m \log \frac{2mKSAH}{\delta} \tag{C.244}$$

For any $(s,a,h,k) \in \mathcal{S} \times \mathcal{A} \times [H] \times [K]$, define events:

$$\mathcal{E}_1(s,a,h,k) := \left\{ \sum_{j \in \mathcal{G}} N_h^{j,k}(s,a) \geq N_0 \right\} \tag{C.245}$$

$$\mathcal{E}_2(s,a,h,k) := \left\{ \max_{i,j \in \mathcal{G}} \frac{N_h^{j,k}(s,a)}{N_h^{i,k}(s,a)} \leq 2 \right\} \tag{C.246}$$

Recall:

$$\mathcal{E}_{\text{even}}(s,a,h,k) \tag{C.247}$$

$$:= \left\{ \text{if } \textstyle\sum_{j \in \mathcal{G}} N_h^{j,k}(s,a) \geq 400m \log \frac{2mKSAH}{\delta}, \text{ then } \max_{i,j \in \mathcal{G}} \frac{N_h^{j,k}(s,a)}{N_h^{i,k}(s,a)} \leq 2 \right\} \tag{C.248}$$

Then we can rewrite even $\mathcal{E}_{\text{even}}(s,a,h,k)$ as:

$$\mathcal{E}_{\text{even}}(s,a,h,k) = \overline{\mathcal{E}_1(s,a,h,k)} \cup \mathcal{E}_2(s,a,h,k) \tag{C.249}$$

We first show that if there are two $N_h^{j,k}$'s, whose ratio exceeds 2, then there must be some $N_h^{j,k}$ that deviates a lot from the empirical mean of $N_h^{j,k}$'s:

$$\overline{\mathcal{E}_2(s,a,h,k)} = \left\{ \max_{i,j\in\mathcal{G}} \frac{N_h^{j,k}(s,a)}{N_h^{i,k}(s,a)} > 2 \right\} \tag{C.250}$$

$$\subseteq \bigcup_{i\in\mathcal{G}} \left\{ N_h^{i,k}(s,a) > \frac{498}{400}\frac{1}{|\mathcal{G}|}\sum_{j\in\mathcal{G}} N_h^{j,k}(s,a) \right\} \cup \bigcup_{i\in\mathcal{G}} \left\{ N_h^{i,k}(s,a) < \frac{302}{400}\frac{1}{|\mathcal{G}|}\sum_{j\in\mathcal{G}} N_h^{j,k}(s,a) \right\} \tag{C.251}$$

$$= \bigcup_{i\in\mathcal{G}} \left\{ N_h^{i,k}(s,a) - \frac{1}{|\mathcal{G}|}\sum_{j\in\mathcal{G}} N_h^{j,k}(s,a) > \frac{98}{400}\frac{1}{|\mathcal{G}|}\sum_{j\in\mathcal{G}} N_h^{j,k}(s,a) \right\} \tag{C.252}$$

$$\cup \bigcup_{i\in\mathcal{G}} \left\{ N_h^{i,k}(s,a) - \frac{1}{|\mathcal{G}|}\sum_{j\in\mathcal{G}} N_h^{j,k}(s,a) < -\frac{98}{400}\frac{1}{|\mathcal{G}|}\sum_{j\in\mathcal{G}} N_h^{j,k}(s,a) \right\} \tag{C.253}$$

$$= \bigcup_{i\in\mathcal{G}} \left\{ \left| N_h^{i,k}(s,a) - \frac{1}{|\mathcal{G}|}\sum_{j\in\mathcal{G}} N_h^{j,k}(s,a) \right| > \frac{98}{400}\frac{1}{|\mathcal{G}|}\sum_{j\in\mathcal{G}} N_h^{j,k}(s,a) \right\} \tag{C.254}$$

To show that $\mathcal{E}_{\text{even}}(s,a,h,k)$ happens w.h.p.:

$$\mathbb{P}\left(\bigcup_{s,a,h,k} \overline{\mathcal{E}_{\text{even}}(s,a,h,k)} \right) = \mathbb{P}\left(\bigcup_{s,a,h,k} \overline{\overline{\mathcal{E}_1(s,a,h,k)} \cup \mathcal{E}_2(s,a,h,k)} \right) \tag{C.255}$$

$$= \mathbb{P}\left(\bigcup_{s,a,h,k} \mathcal{E}_1(s,a,h,k) \cap \overline{\mathcal{E}_2(s,a,h,k)} \right) \tag{C.256}$$

$$\leq \mathbb{P}\Bigg(\exists s,a,h,k, \sum_{j\in\mathcal{G}} N_h^{j,k}(s,a) \geq N_0, \tag{C.257}$$

$$\exists i \in \mathcal{G}, \left| N_h^{i,k}(s,a) - \frac{1}{|\mathcal{G}|}\sum_{j\in\mathcal{G}} N_h^{j,k}(s,a) \right| > \frac{98}{400}\frac{1}{|\mathcal{G}|}\sum_{j\in\mathcal{G}} N_h^{j,k}(s,a) \Bigg) \tag{C.258}$$

(By (C.254)) $\tag{C.259}$

$$\leq \mathbb{P}\left(\exists s,a,h,k,i \left| N_h^{i,k}(s,a) - \frac{1}{|\mathcal{G}|}\sum_{j\in\mathcal{G}} N_h^{j,k}(s,a) \right| \right. \tag{C.260}$$

$$> \frac{18}{400}\frac{1}{|\mathcal{G}|}N_0 + 4\sqrt{\frac{1}{400}\frac{1}{|\mathcal{G}|}N_0}\sqrt{\frac{1}{|\mathcal{G}|}\sum_{j\in\mathcal{G}} N_h^{j,k}(s,a)}\Bigg) \tag{C.261}$$

$$= \mathbb{P}\left(\exists s,a,h,k,i \left| N_h^{i,k}(s,a) - \frac{1}{|\mathcal{G}|}\sum_{j\in\mathcal{G}} N_h^{j,k}(s,a)\right| \right. \tag{C.262}$$

$$\left. > 18\log\frac{2mKSAH}{\delta} + 4\sqrt{\log\frac{2mKSAH}{\delta}}\sqrt{\frac{1}{|\mathcal{G}|}\sum_{j\in\mathcal{G}} N_h^{j,k}(s,a)}\right) \tag{C.263}$$

$$< 2\delta \quad \text{(By Lemma C.3.8)} \tag{C.264}$$

■

Proof of Lemma C.3.8

The high-level ideas are:

1. For each s, a, h,
   - for each $j \in \mathcal{G}$, define centered $N_h^{j,k}(s,a)$ as a martingale;
   - define centered $\sum_{j\in\mathcal{G}} N_h^{j,k}(s,a)$ as a martingale;
2. apply a modified Bernstein type of martingale concentration bound for both centered $N_h^{j,k}(s,a)$'s and centered $\sum_{j\in\mathcal{G}} N_h^{j,k}(s,a)$ (see page 233 and page 234);
3. because $N_h^{j,k}(s,a)$ and $\frac{1}{\tilde{m}}\sum_{j\in\mathcal{G}} N_h^{j,k}(s,a)$ have the same mean, we can use triangular inequality to show these two terms are close, and the distance is bounded by the variance term in Bernstein inequality.
4. Bernstein on $\frac{1}{\tilde{m}}\sum_{j\in\mathcal{G}} N_h^{j,k}(s,a)$ also allow us to bound its variance in terms of itself.
5. We can get our result by combining Step 3 and Step 4.

Lemma C.3.9 (Concentration of each $N_h^{j,k}(s,a)$)**.** *For all $0 < \delta \le 1/4$, with probability at least $1-\delta$, for all $(s,a,h,j,k) \in \mathcal{S}\times\mathcal{A}\times[H]\times\mathcal{G}\times[K]$:*

$$\left|N_h^{j,k}(s,a) - \sum_{t=1}^{k} d_h^{\hat{\pi}^t}(s,a)\right| < 3\log\frac{2SAHmK}{\delta} + \sqrt{2\sum_{t=1}^{k} d_h^{\hat{\pi}^t}(s,a)\log\frac{2SAHmK}{\delta}} \tag{C.265}$$

Proof of Lemma C.3.9. See page 235 ■

Lemma C.3.10 (Concentration of each $\frac{1}{\tilde{m}}\sum_{j\in\mathcal{G}} N_h^{j,k}(s,a)$)**.** *For all $0 < \delta \le 1/4$, with probability at least $1-\delta$, for all $(s,a,h,k) \in \mathcal{S}\times\mathcal{A}\times[H]\times[K]$:*

$$\left|\sum_{j\in\mathcal{G}} N_h^{j,k}(s,a) - |\mathcal{G}|\sum_{t=1}^{k} d_h^{\hat{\pi}^t}(s,a)\right| < 3\log\frac{2SAHmK}{\delta} + \sqrt{2|\mathcal{G}|\sum_{t=1}^{k} d_h^{\hat{\pi}^t}(s,a)\log\frac{2SAHmK}{\delta}} \tag{C.266}$$

Proof of Lemma C.3.10. See page 236 ■

Proof of Lemma C.3.8. Let $\mathcal{E}_N$ the intersection of the events in page 233 and page 234. Then by page 233 and page 234, $\mathcal{E}_N$ happens with probability at least $1-2\delta$. By page 234,

$$\sqrt{\sum_{t=1}^{k} d_h^{\hat{\pi}^t}(s,a)} \le 4\sqrt{\log\frac{2SAHmK}{\delta}} + \sqrt{\frac{1}{|\mathcal{G}|}\sum_{j\in\mathcal{G}} N_h^{j,k}(s,a)} \tag{C.267}$$

By page 234 and page 234, for all s,a,h,j,k

$$\left|\frac{1}{|\mathcal{G}|}\sum_{j'\in\mathcal{G}} N_h^{j',k}(s,a) - N_h^{j,k}(s,a)\right| \tag{C.268}$$

$$\le \left|N_h^{j,k}(s,a) - \sum_{t=1}^{k} d_h^{\hat{\pi}^t}(s,a)\right| + \left|\frac{1}{|\mathcal{G}|}\sum_{j'\in\mathcal{G}} N_h^{j',k}(s,a) - \sum_{t=1}^{k} d_h^{\hat{\pi}^t}(s,a)\right| \tag{C.269}$$

$$\le 6\log\frac{2SAHmK}{\delta} + 2\sqrt{2\sum_{t=1}^{k} d_h^{\hat{\pi}^t}(s,a)\log\frac{2SAHmK}{\delta}} \tag{C.270}$$

$$\leq 6\log\frac{2SAHmK}{\delta} + 2\sqrt{2\log\frac{2SAHmK}{\delta}}\left(4\sqrt{\log\frac{2SAHmK}{\delta}} + \sqrt{\frac{1}{|\mathcal{G}|}\sum_{j\in\mathcal{G}} N_h^{j,k}(s,a)}\right) \tag{C.271}$$

$$\leq 18\log\frac{2SAHmK}{\delta} + 4\sqrt{\log\frac{2SAHmK}{\delta}}\sqrt{\frac{1}{|\mathcal{G}|}\sum_{j\in\mathcal{G}} N_h^{j,k}(s,a)} \tag{C.272}$$

■

Proof of Lemma C.3.9

Proof of Lemma C.3.9. For each fixed $(s,a,h,j) \in \mathcal{S}\times\mathcal{A}\times[H]\times\mathcal{G}$: for all $t \in [K]$, define

$$\mathcal{F}_k := \sigma\left(\bigcup_{t\leq k}\bigcup_{j\in[m]}\left\{\left(s_h^{j,t}, a_h^{j,t}, r_h^{j,t}, s_{h+1}^{j,t}\right)\right\}_{h=1}^{H}\right). \tag{C.273}$$

Let

$$S_h^{j,k}(s,a) = N_h^{j,k}(s,a) - \sum_{t=1}^{k} d_h^{\hat{\pi}^t}(s,a) \tag{C.274}$$

$$T_h^{j,k}(s,a) = \sum_{t=1}^{k} d_h^{\hat{\pi}^t}(s,a)\left(1 - d_h^{\hat{\pi}^t}(s,a)\right) \tag{C.275}$$

Then $\left\{\left(\mathcal{F}_k, S_h^{j,k}(s,a)\right)\right\}_{t=k}^{K}$ is a martingale. Since $d_h^{\hat{\pi}^k}(s,a)$ depends on $\hat{\pi}^k$, which is calculated use data in the first $k-1$ episodes, then $d_h^{\hat{\pi}^k}(s,a) \in \mathcal{F}_{k-1}$. By page 243,

$$\mathbb{P}\left(\bigcup_{k=1}^{K}\left\{|S_h^{j,k}(s,a)| \geq 3\log\frac{2SAHmK}{\delta} + \sqrt{2\sum_{t=1}^{k} d_h^{\hat{\pi}^t}(s,a)\log\frac{2SAHmK}{\delta}}\right\}\right) \tag{C.276}$$

$$\leq \mathbb{P}\left(\bigcup_{k=1}^{K}\left\{|S_h^{j,k}(s,a)| \geq 3\log\frac{2SAHmK}{\delta} + \sqrt{2T_h^{j,k}(s,a)\log\frac{2SAHmK}{\delta}}\right\}\right) \leq \frac{\delta}{SAHm} \tag{C.277}$$

By union bound, with probability at least $1-\delta$, $\forall(s,a,h,j,k)\in\mathcal{S}\times\mathcal{A}\times[H]\times\mathcal{G}\times[K]$:

$$|S_h^{j,k}(s,a)|< 3\log\frac{2SAHmK}{\delta}+\sqrt{2\sum_{t=1}^{k}d_h^{\hat{\pi}^t}(s,a)\log\frac{2SAHmK}{\delta}} \tag{C.278}$$

■

Proof of Lemma C.3.10

Proof of Lemma C.3.10. During the data-collecting process, the agents are allowed to collect data simultaneously. For analysis purposes, we artificially order the data in the following sequence:

$$E^{1,1},E^{2,1},\ldots,E^{m,1},E^{1,2},\ldots,E^{m,2},\ldots,E^{1,K},\ldots,E^{m,K} \tag{C.279}$$

where $E^{j,k}:=\left\{\left(s_h^{j,k},a_h^{j,k},r_h^{j,k},s_{h+1}^{j,k}\right)\right\}_{h=1}^{H}$. Let

$$\mathcal{F}_t=\sigma\left(\bigcup_{j,k\text{ s.t. }m(k-1)+j\le t}E^{j,k}\right). \tag{C.280}$$

Then $\{\mathcal{F}_t\}_{t=0}^{mK}$ forms a valid filtration. Define the following functions to map from sequence index to agent index and episode index:

$$\mathcal{J}(t):=t-m\left(\lceil t/m\rceil-1\right),\quad \mathcal{K}(t):=\lceil t/m\rceil \tag{C.281}$$

For each fixed $(s,a,h)\in\mathcal{S}\times\mathcal{A}\times[H]$, for all $t\in[mK]$, we define $S_h^{\mathcal{G},t}(s,a)$ as the (centered) total counts of (s,a,h) collected by all good agents up to time t. The t-th term in page 236 could be in the center of an episode, meaning some agents have not collected their trajectories yet. So we need to treat the agents differently: Let

$$S_h^{\mathcal{G},t}(s,a)=\sum_{j\in\mathcal{G},j\le\mathcal{J}(t)}\left(N_h^{j,\mathcal{K}(t)}(s,a)-\sum_{t=1}^{\mathcal{K}(t)}d_h^{\hat{\pi}^t}(s,a)\right) \tag{C.282}$$

$$+\sum_{j\in\mathcal{G},j>\mathcal{J}(t)}\left(N_h^{j,\mathcal{K}(t)-1}(s,a)-\sum_{t=1}^{\mathcal{K}(t)-1}d_h^{\hat{\pi}^t}(s,a)\right) \tag{C.283}$$

Then $\left\{\left(\mathcal{F}_t, S_h^{\mathcal{G},t}(s,a)\right)\right\}_{t=1}^{mK}$ is a martingale. Similar to page 233, define

$$T_h^{\mathcal{G},t}(s,a)=\sum_{j\in\mathcal{G},j\leq\mathcal{J}(t)}\sum_{t=1}^{\mathcal{K}(t)}d_h^{\hat{\pi}^t}(s,a)\left(1-d_h^{\hat{\pi}^t}(s,a)\right) \tag{C.284}$$

$$+\sum_{j\in\mathcal{G},j>\mathcal{J}(t)}\sum_{t=1}^{\mathcal{K}(t)-1}d_h^{\hat{\pi}^t}(s,a)\left(1-d_h^{\hat{\pi}^t}(s,a)\right) \tag{C.285}$$

Then by page 243,

$$\mathbb{P}\left(\bigcup_{k\in[K]}\left\{\left|\sum_{j\in\mathcal{G}}N_h^{j,k}(s,a)-|\mathcal{G}|\sum_{t=1}^{k}d_h^{\hat{\pi}^t}(s,a)\right|\geq 3\log\frac{2SAHmK}{\delta}\right.\right. \tag{C.286}$$

$$\left.\left.+\sqrt{2|\mathcal{G}|\sum_{t=1}^{k}d_h^{\hat{\pi}^t}(s,a)\log\frac{2SAHmK}{\delta}}\right\}\right) \tag{C.287}$$

$$\leq\mathbb{P}\left(\bigcup_{k\in[K]}\left\{\left|\sum_{j\in\mathcal{G}}N_h^{j,k}(s,a)-|\mathcal{G}|\sum_{t=1}^{k}d_h^{\hat{\pi}^t}(s,a)\right|\geq 3\log\frac{2SAHmK}{\delta}\right.\right. \tag{C.288}$$

$$\left.\left.+\sqrt{2|\mathcal{G}|\sum_{t=1}^{k}d_h^{\hat{\pi}^t}(s,a)\left(1-d_h^{\hat{\pi}^t}(s,a)\right)\log\frac{2SAHmK}{\delta}}\right\}\right) \tag{C.289}$$

$$\leq\mathbb{P}\left(\bigcup_{k=1}^{mK}\left\{|S_h^{\mathcal{G},k}(s,a)|\geq 3\log\frac{2SAHmK}{\delta}+\sqrt{2T_h^{\mathcal{G},k}(s,a)\log\frac{2SAHmK}{\delta}}\right\}\right) \tag{C.290}$$

$$\leq\frac{\delta}{SAH} \tag{C.291}$$

By union bound, with probability at least $1-\delta$, for all $(s,a,h,k)\in\mathcal{S}\times\mathcal{A}\times[H]\times[K]$:

$$\left|\sum_{j\in\mathcal{G}}N_h^{j,k}(s,a)-|\mathcal{G}|\sum_{t=1}^{k}d_h^{\hat{\pi}^t}(s,a)\right|<3\log\frac{2SAHmK}{\delta}+\sqrt{2|\mathcal{G}|\sum_{t=1}^{k}d_h^{\hat{\pi}^t}(s,a)\log\frac{2SAHmK}{\delta}} \tag{C.292}$$

■

C.4 Proof of Theorem 5.6.1

By the following lemma, we can upper bound the suboptimality by the cumulative bonuses:

Lemma C.4.1. [*Suboptimality for Pessimistic Value Iteration, Lemma 3.2 in (Zhang et al., 2021a) and Theorem 4.2 in (Jin et al., 2021)*] *Under the event $\mathcal{E}$ that the $\Gamma_h(s,a)$ satisfies the required property of bounding the Bellman error, i.e. $|\hat{Q}_h(s,a) - (\mathbb{B}\hat{V}_{h+1})(s,a)| \leq \Gamma_h(s,a), \forall h \in [H], (s,a) \in \mathcal{S} \times \mathcal{A}$ then against any comparator policy $\tilde{\pi}$, it achieves*

$$\mathrm{SubOpt}(\hat{\pi}, \tilde{\pi}) \leq 2 \sum_{h=1}^{H} \mathbb{E}_{d^{\tilde{\pi}}}[\Gamma_h(s_h, a_h)] \tag{C.293}$$

Recall that for all $(s,a,h) \in \mathcal{S}\times\mathcal{A}\times[H]$,

$$N_h^j(s,a) := \sum_{k \in [K_j]} \mathbf{1}\left\{(s_h^{j,k}, a_h^{j,k}) = (s,a)\right\}, \quad \forall j \in [m]. \tag{C.294}$$

and $N_h^{\mathrm{cut}}(s,a)$ is the $(2\alpha m + 1)$-largest among $\left\{N_h^j(s,a)\right\}_{j\in[m]}$. $N_h^{\mathcal{G},\mathrm{cut}_1}(s,a)$ is the $(\alpha m + 1)$-th largest of $\left\{N_h^j(s,a)\right\}_{j\in\mathcal{G}}$ and $N_h^{\mathcal{G},\mathrm{cut}_2}(s,a)$ is the $(2\alpha m + 1)$-th largest of $\left\{N_h^j(s,a)\right\}_{j\in\mathcal{G}}$. The bonuses are given by:

- If $N_h^{\mathrm{cut}}(s,a) = 0$

$$\Gamma_h(s,a) = H - h + 1; \tag{C.295}$$

- If $N_h^{\mathrm{cut}}(s,a) > 0$

$$\Gamma_h(s,a) := \frac{2(H-h+1)}{\sqrt{\sum_{j\in[m]} \tilde{N}_h^j(s,a)}} \sqrt{2\log\frac{2SAH}{\delta}} \tag{C.296}$$

$$+ \frac{8\alpha m \sqrt{N_h^{\mathrm{cut}}(s,a)}}{\sum_{j\in[m]} \tilde{N}_h^j(s,a)} (H-h+1) \sqrt{2\log\frac{2mSAH}{\delta}} \tag{C.297}$$

Where

$$\tilde{N}_h^j(s,a) = \max\left(N_h^{\mathrm{cut}}(s,a), N_h^j(s,a)\right). \tag{C.298}$$

Proof of Theorem 5.6.1. We first show that with probability at least $1-\delta$,

$$|(\hat{\mathbb{B}}_h\hat{V}_{h+1})(s,a)-(\mathbb{B}_h\hat{V}_{h+1})(s,a)|\leq \Gamma_h(s,a), \quad \forall(s,a)\in\mathcal{S}\times\mathcal{A},\forall h\in[H] \tag{C.299}$$

where $\Gamma_h(s,a)$ is defined in (C.293).

- if $N_h^{\text{cut}}(s,a)=0$, by definition, $(\hat{\mathbb{B}}_h\hat{V}_{h+1})(s,a)=0$. By definition of $\hat{V}_h$ and $\mathbb{B}_h$, $(\mathbb{B}_h\hat{V}_{h+1})(s,a)\in[0,H-h+1]$, thus (C.299) holds;
- if $N_h^{\text{cut}}(s,a)>0$, for any fixed $h\in[H]$, $(s,a)\in\mathcal{S}\times\mathcal{A}$, $f:\mathcal{S}\to[0,H]$. Because $(\hat{\mathbb{B}}_hf)(s,a)$ is bounded and thus sub-Gaussian, we can use Theorem 5.3.1 to upper bound $|(\hat{\mathbb{B}}_hf)(s,a)-(\mathbb{B}_hf)(s,a)|$:

$$\mathbb{P}\left(\left|(\hat{\mathbb{B}}_hf)(s,a)-(\mathbb{B}_hf)(s,a)\right|\geq\Gamma_h(s,a)\right)\leq\frac{\delta}{HSA} \tag{C.300}$$

Thus

$$\begin{aligned}
&\mathbb{P}\left(|(\hat{\mathbb{B}}_h\hat{V}_{h+1})(s,a)-(\mathbb{B}_h\hat{V}_{h+1})(s,a)|\geq\Gamma_h(s,a)\right) && \text{(C.301)}\\
=&\int_{[0,H]^{\mathcal{S}}}\mathbb{P}\left(|(\hat{\mathbb{B}}_h\hat{V}_{h+1})(s,a)-(\mathbb{B}_h\hat{V}_{h+1})(s,a)|\geq\Gamma_h(s,a)\middle|\hat{V}_{h+1}(\cdot)\right)d\mathbb{P}(\hat{V}_{h+1}(\cdot)) && \text{(C.302)}\\
\leq&\frac{\delta}{HSA} && \text{(C.303)}
\end{aligned}$$

By union bound, with probability at least $1-\delta$,

$$|(\hat{\mathbb{B}}_h\hat{V}_{h+1})(s,a)-(\mathbb{B}_h\hat{V}_{h+1})(s,a)|\leq \Gamma_h(s,a), \quad \forall(s,a)\in\mathcal{S}\times\mathcal{A},\forall h\in[H] \tag{C.304}$$

Then, by Lemma C.4.1, with probability at least $1-\delta$,

$$\begin{aligned}
\text{SubOpt}(\hat{\pi},\tilde{\pi})\leq&\ 2\sum_{h=1}^{H}\mathbb{E}_{d^{\tilde{\pi}}}[\Gamma_h(s_h,a_h)] && \text{(C.305)}\\
=&2\sum_{h=1}^{H}\mathbb{E}_{d^{\tilde{\pi}}}\left[\Gamma_h(s_h,a_h)\mathbf{1}\left\{N_h^{\mathcal{G},\text{cut}_2}(s_h,a_h)=0\right\}\right] && \text{(C.306)}
\end{aligned}$$

$$+ 2\sum_{h=1}^{H} \mathbb{E}_{d^{\tilde{\pi}}}\left[\Gamma_h(s_h,a_h)\mathbf{1}\left\{N_h^{\mathcal{G},\mathrm{cut}_2}(s_h,a_h)>0\right\}\right] \tag{C.307}$$

$$=: \mathscr{A}_1 + \mathscr{A}_2. \tag{C.308}$$

By definition of $p^{\mathcal{G},0}$ in page 64,

$$\mathscr{A}_1 \le 2Hp^{\mathcal{G},0} \tag{C.309}$$

$$\mathscr{A}_2 = 2\sum_{h=1}^{H} \mathbb{E}_{d^{\tilde{\pi}}}\left[\Gamma_h(s_h,a_h)\mathbf{1}\left\{N_h^{\mathcal{G},\mathrm{cut}_2}(s_h,a_h)>0\right\}\right] \tag{C.310}$$

$$\le 2\sum_{h=1}^{H} \mathbb{E}_{d^{\tilde{\pi}}}\left[\left(\frac{2(H-h+1)}{\sqrt{\sum_{j\in\mathcal{G}}\tilde{N}_h^j(s,a)}}\sqrt{2\log\frac{2SAH}{\delta}}\right.\right. \tag{C.311}$$

$$\left.\left.+\frac{8\alpha m\sqrt{N_h^{\mathrm{cut}}(s,a)}}{\sum_{j\in\mathcal{G}}\tilde{N}_h^j(s,a)}(H-h+1)\sqrt{2\log\frac{2mSAH}{\delta}}\right)\mathbf{1}\left\{N_h^{\mathcal{G},\mathrm{cut}_2}(s_h,a_h)>0\right\}\right]. \tag{C.312}$$

By the definition of κ_{even} in page 65: for $a=\tilde{\pi}(s)$,

$$\frac{1}{\sqrt{\sum_{j\in\mathcal{G}}\tilde{N}_h^j(s,a)}} = \frac{\sqrt{\sum_{j\in\mathcal{G}}N_h^j(s,a)}}{\sqrt{\sum_{j\in\mathcal{G}}\tilde{N}_h^j(s,a)}}\frac{1}{\sqrt{\sum_{j\in\mathcal{G}}N_h^j(s,a)}} \tag{C.313}$$

$$\le \frac{\sqrt{\sum_{j\in\mathcal{G}}N_h^j(s,a)}}{\sqrt{\sum_{j\in\mathcal{G}}\tilde{N}_h^{j,\mathrm{cut}_2}(s,a)}}\frac{1}{\sqrt{\sum_{j\in\mathcal{G}}N_h^j(s,a)}} \tag{C.314}$$

$$\le \frac{\sqrt{\kappa_{\mathsf{even}}}}{\sqrt{\sum_{j\in\mathcal{G}}N_h^j(s,a)}} \tag{C.315}$$

and

$$\frac{m\sqrt{N_h^{\mathrm{cut}}(s,a)}}{\sum_{j\in\mathcal{G}}\tilde{N}_h^j(s,a)} \le \frac{1}{\sqrt{1-\alpha}}\sqrt{\frac{\sum_{j\in\mathcal{G}}N_h^j(s,a)}{\sum_{j\in\mathcal{G}}\tilde{N}_h^j(s,a)}\frac{m(1-\alpha)N_h^{\mathrm{cut}}(s,a)}{\sum_{j\in\mathcal{G}}\tilde{N}_h^j(s,a)}}\frac{\sqrt{m}}{\sqrt{\sum_{j\in\mathcal{G}}N_h^j(s,a)}} \tag{C.316}$$

$$\leq \sqrt{\frac{\sum_{j\in\mathcal{G}} N_h^j(s,a)}{\sum_{j\in\mathcal{G}} \tilde{N}_h^{j,\mathrm{cut}_2}(s,a)} \frac{m(1-\alpha)N_h^{\mathcal{G},\mathrm{cut}_1}(s,a)}{\sum_{j\in\mathcal{G}} \tilde{N}_h^{j,\mathrm{cut}_2}(s,a)}} \frac{\sqrt{2m}}{\sqrt{\sum_{j\in\mathcal{G}} N_h^j(s,a)}} \tag{C.317}$$

$$\leq \frac{\sqrt{2\kappa_{\text{even}} m}}{\sqrt{\sum_{j\in\mathcal{G}} N_h^j(s,a)}} \tag{C.318}$$

Thus

$$\mathscr{A}_2 \leq 2\sum_{h=1}^{H} \mathbb{E}_{d^{\tilde{\pi}}}\left[\left(\frac{2}{\sqrt{\sum_{j\in\mathcal{G}} \tilde{N}_h^j(s,a)}} + \frac{8\alpha m\sqrt{N_h^{\mathrm{cut}}(s,a)}}{\sum_{j\in\mathcal{G}} \tilde{N}_h^j(s,a)}\right) H\sqrt{2\log\frac{2mSAH}{\delta}}\right. \tag{C.319}$$

$$\left.\mathbf{1}\left\{N_h^{\mathcal{G},\mathrm{cut}_2}(s_h,a_h)>0\right\}\right] \tag{C.320}$$

$$\leq 2\sum_{h=1}^{H} \mathbb{E}_{d^{\tilde{\pi}}}\left[\frac{\left(2+8\alpha\sqrt{2m}\right)\sqrt{\kappa_{\text{even}}}}{\sqrt{\sum_{j\in\mathcal{G}} N_h^j(s,a)}} H\sqrt{2\log\frac{2mSAH}{\delta}}\mathbf{1}\left\{N_h^{\mathcal{G},\mathrm{cut}_2}(s_h,a_h)>0\right\}\right] \tag{C.321}$$

$$\leq 2\left(2+8\alpha\sqrt{2m}\right)\sqrt{\kappa_{\text{even}}} H\sqrt{2\log\frac{2mSAH}{\delta}} \sum_{h=1}^{H} \mathbb{E}_{d^{\tilde{\pi}}}\left[\frac{\mathbf{1}\left\{N_h^{\mathcal{G},\mathrm{cut}_2}(s_h,a_h)>0\right\}}{\sqrt{\sum_{j\in\mathcal{G}} N_h^j(s,a)}}\right] \tag{C.322}$$

Recall that $\mathcal{C}_h = \left\{s | N_h^{\mathcal{G},\mathrm{cut}_2}(s,\tilde{\pi}(s))>0\right\}$. By Cauchy–Schwarz inequality and the definition of κ in page 64,

$$\mathbb{E}_{d^{\tilde{\pi}}}\left[\frac{\mathbf{1}\left\{N_h^{\mathcal{G},\mathrm{cut}_2}(s_h,a_h)>0\right\}}{\sqrt{\sum_{j\in\mathcal{G}} N_h^j(s,a)}}\right] \leq \sqrt{\mathbb{E}_{d^{\tilde{\pi}}}\left[\frac{\mathbf{1}\left\{N_h^{\mathcal{G},\mathrm{cut}_2}(s_h,a_h)>0\right\}}{\sum_{j\in\mathcal{G}} N_h^j(s,a)}\right]} \tag{C.323}$$

$$= \sqrt{\sum_{s\in\mathcal{C}_h} \frac{d_h^{\tilde{\pi}}(s)}{\sum_{j\in\mathcal{G}} N_h^j(s,a)}} \tag{C.324}$$

$$= \sqrt{\sum_{s\in\mathcal{C}_h} \frac{d_h^{\tilde{\pi}}(s)}{\sum_{j\in\mathcal{G}} N_h^j(s,a)/\sum_{j\in\mathcal{G}} K_j} \frac{1}{\sum_{j\in\mathcal{G}} K_j}} \tag{C.325}$$

$$\leq \sqrt{\sum_{s \in \mathcal{C}_h} \frac{\kappa}{\sum_{j \in \mathcal{G}} K_j}} \leq \sqrt{\frac{\kappa S}{\sum_{j \in \mathcal{G}} K_j}} \tag{C.326}$$

In conclusion,

$$\begin{aligned}
\mathrm{SubOpt}(\hat{\pi}, \tilde{\pi}) \leq & \mathscr{A}_1 + \mathscr{A}_2 && \text{(C.327)} \\
\leq & 2Hp^{\mathcal{G},0} + 2\left(2 + 8\alpha\sqrt{2m}\right) \frac{\sqrt{\kappa\kappa_{\text{even}} S}}{\sqrt{\sum_{j \in \mathcal{G}} K_j}} H^2 \sqrt{2 \log \frac{2mSAH}{\delta}} && \text{(C.328)} \\
= & 2Hp^{\mathcal{G},0} + O\left(\sqrt{\kappa\kappa_{\text{even}}} H^2 \sqrt{S} \frac{1 + \alpha\sqrt{m}}{\sqrt{\sum_{j \in \mathcal{G}} K_j}} \sqrt{\log \frac{mSAH}{\delta}}\right) && \text{(C.329)}
\end{aligned}$$

■

C.5 Useful Inequalities

Theorem C.5.1 (Bernstein type of bound for martingale, Theorem 1.6 of (Freedman, 1975))**.** *Let $(\Omega, \mathcal{F}, P)$ be a probability triple. Let $\mathcal{F}_0 \subset \mathcal{F}_1 \subset \cdots$ be an increasing sequence of sub-σ-fields of $\mathcal{F}$. Let $X_1, X_2, \ldots$ be random variables on $(\Omega, \mathcal{F}, P)$, such that X_n is $\mathcal{F}_n$ measurable. Let $V_n = \mathbb{V}[X_n | \mathcal{F}_{n-1}]$. Assume $|X_n| \leq 1$ and $\mathbb{E}[X_n | \mathcal{F}_{n-1}] = 0$. Let*

$$S_n = X_1 + \cdots + X_n \tag{C.330}$$

$$T_n = V_1 + \cdots + V_n, \tag{C.331}$$

where $S_0 = T_0 = 0$. Then, for any $a > 0$, $b > 0$,

$$\mathbb{P}\left(|S_n| \geq a \text{ and } T_n \leq b \text{ for some } n\right) \leq 2 \exp\left(-\frac{a^2}{2(a+b)}\right). \tag{C.332}$$

By union bound and partition, we can get a more useful version of page 242. We first present a result, which shows: given,

$$\mathbb{P}\left(X \geq t, Y \leq t\right) \leq \delta(t) \tag{C.333}$$

We can bound $\mathbb{P}(X \geq Y)$ up to some error.

Lemma C.5.1. *Let $\{A_n\}_{n=1}^N$ and $\{B_n\}_{n=1}^N$ be two sequences of random variables. We don't make any assumptions about independence. Assume*

- *$\forall n$, $0 \leq B_n \leq nM$ almost surely;*
- *$\forall \delta > 0$, $f_\delta : \mathbb{R}_+ \mapsto \mathbb{R}_+$, $f_\delta(\cdot)$ monotonic increasing,*

If for all $t > 0$,

$$\mathrm{P}\left(\bigcup_{n=1}^{N} \{|A_n| \geq f_\delta(t), B_n \leq t\}\right) \leq \delta \tag{C.334}$$

Then for any $\epsilon > 0$,

$$\mathrm{P}\left(\bigcup_{n=1}^{N} \{|A_n| \geq f_\delta(B_n + \epsilon)\}\right) \leq NM\lceil 1/\epsilon \rceil \delta \tag{C.335}$$

Proof. See proof in page 244. ∎

Corollary C.5.1. *Under the assumption of Theorem C.5.1, suppose X_n terminate at $n = N$. Then, for all $0 < \delta < 2\exp(-2)$,*

$$\mathbb{P}\left(\bigcup_{n=1}^{N} \left\{|S_n| \geq 3\log\frac{2N}{\delta} + \sqrt{2T_n \log\frac{2N}{\delta}}\right\}\right) \leq \delta \tag{C.336}$$

Proof of Corollary C.5.1. Let $\frac{\delta}{N} = 2\exp\left(-\frac{a^2}{2(a+b)}\right)$ then

$$a = \log\frac{2N}{\delta} + \sqrt{\log^2\frac{2N}{\delta} + 2b\log\frac{2N}{\delta}} \tag{C.337}$$

by Theorem C.5.1, For all $b > 0$,

$$\mathbb{P}\left(|S_n| \geq \log\frac{2N}{\delta} + \sqrt{\log^2\frac{2N}{\delta} + 2b\log\frac{2N}{\delta}}, \text{ and } T_n \leq b \text{ for some } n\right) \leq \delta/N \tag{C.338}$$

In Lemma C.5.1, let:

- $A_n = S_n$, $B_n = T_n$, $M = 1$
- $\epsilon = \frac{1}{2}\log\frac{2N}{\delta}$
- $f_\delta(x) = \log\frac{2N}{\delta} + \sqrt{\log^2\frac{2N}{\delta} + 2x\log\frac{2N}{\delta}}$

Because $0 < \delta < 2\exp(-2)$, $\epsilon \geq 1$. then, we get:

$$\mathbb{P}\left(\bigcup_{n=1}^{N}\left\{|S_n| \geq 3\log\frac{2N}{\delta} + \sqrt{2T_n\log\frac{2N}{\delta}}\right\}\right) \tag{C.339}$$

$$\leq \mathbb{P}\left(\bigcup_{n=1}^{N}\left\{|S_n| \geq \log\frac{2N}{\delta} + \sqrt{2\log^2\frac{2N}{\delta} + 2T_n\log\frac{2N}{\delta}}\right\}\right) \tag{C.340}$$

$$\leq N\lceil 1/\epsilon\rceil\frac{\delta}{N} \leq \delta \tag{C.341}$$

■

Proof For Lemma C.5.1

Proof of Lemma C.5.1. For discrete random variables, we can just condition on each possible value of B_n and use a union bound. Here, because B_n can be a continuous random variable, we divide the range of B_n into intervals and upper bound the target by the law of total probability.

For all n, let:

$$0 < \frac{1}{\lceil 1/\epsilon\rceil} < \frac{2}{\lceil 1/\epsilon\rceil} < \cdots < \frac{nM\lceil 1/\epsilon\rceil}{\lceil 1/\epsilon\rceil} = nM \tag{C.342}$$

Be a partition of interval $[0, nM]$. Let $I_i := \left[\frac{i-1}{\lceil 1/\epsilon\rceil}, \frac{i}{\lceil 1/\epsilon\rceil}\right], i = 1, \ldots, nM\lceil 1/\epsilon\rceil$ be a set of intervals. Note that, $\bigcup_{i=1}^{nM\lceil 1/\epsilon\rceil} I_i = [0, nM]$.

Then

$$\bigcup_{n=1}^{N}\{|A_n| \geq f_\delta(B_n + \epsilon)\} = \bigcup_{n=1}^{N}\bigcup_{i=1}^{nM\lceil 1/\epsilon\rceil}\{|A_n| \geq f_\delta(B_n + \epsilon), B_n \in I_i\} \tag{C.343}$$

$$= \bigcup_{n=1}^{N} \bigcup_{i=1}^{nM\lceil 1/\epsilon \rceil} \left\{ |A_n| \geq f_\delta(B_n + \epsilon), \frac{i-1}{\lceil 1/\epsilon \rceil} \leq B_n \leq \frac{i}{\lceil 1/\epsilon \rceil} \right\} \tag{C.344}$$

$$\subseteq \bigcup_{n=1}^{N} \bigcup_{i=1}^{nM\lceil 1/\epsilon \rceil} \left\{ |A_n| \geq f_\delta(\frac{i}{\lceil 1/\epsilon \rceil}), B_n \leq \frac{i}{\lceil 1/\epsilon \rceil} \right\} \tag{C.345}$$

$$\subseteq \bigcup_{n=1}^{N} \bigcup_{i=1}^{NM\lceil 1/\epsilon \rceil} \left\{ |A_n| \geq f_\delta(\frac{i}{\lceil 1/\epsilon \rceil}), B_n \leq \frac{i}{\lceil 1/\epsilon \rceil} \right\} \tag{C.346}$$

$$= \bigcup_{i=1}^{NM\lceil 1/\epsilon \rceil} \bigcup_{n=1}^{N} \left\{ |A_n| \geq f_\delta(\frac{i}{\lceil 1/\epsilon \rceil}), B_n \leq \frac{i}{\lceil 1/\epsilon \rceil} \right\} \tag{C.347}$$

Thus

$$\mathrm{P}\left(\bigcup_{n=1}^{N} \{ |A_n| \geq f_\delta(B_n + \epsilon) \} \right) \leq \sum_{i=1}^{NM\lceil 1/\epsilon \rceil} \mathbb{P}\left(\bigcup_{n=1}^{N} \left\{ |A_n| \geq f_\delta(\frac{i}{\lceil 1/\epsilon \rceil}), B_n \leq \frac{i}{\lceil 1/\epsilon \rceil} \right\} \right) \tag{C.348}$$

$$\leq NM\lceil 1/\epsilon \rceil \delta \quad \text{(By page 243)} \tag{C.349}$$

■

D APPENDIX FOR CHAPTER 6

D.1 Deferred Algorithms

See Algorithm 17.

Algorithm 17 TRIMMED-MEAN (Univariate mean estimator in Lugosi and Mendelson (2021))

Input: Corrupted dataset $X_1, \dots, X_N$, corruption level ϵ, confidence level δ
$\tilde{\epsilon} \leftarrow 8\epsilon + \frac{24}{N} \log \frac{8}{\delta}$
Let $\tilde{X}_1 \leq \cdots \leq \tilde{X}_{N/2}$ be a rearrangement of $X_{\frac{N}{2}+1}, \dots X_N$.
Let $\alpha \leftarrow \tilde{X}_{\tilde{\epsilon}N/2}$, $\beta \leftarrow \tilde{X}_{(1-\tilde{\epsilon})N/2}$
Let $\phi_{\alpha,\beta}(\cdot)$ be a clipping function, s.t. $\phi_{\alpha,\beta}(x) = \beta$ if $x > \beta$; $\phi_{\alpha,\beta}(x) = x$ if $\alpha \leq x \leq \beta$; $\phi_{\alpha,\beta}(x) = \alpha$ if $x < \alpha$;
Return: $\hat{\mu} \leftarrow \frac{2}{N} \sum_{i=1}^{N/2} \phi_{\alpha,\beta}(X_i)$.

D.2 Proof of Proposition 6.3.1

Proof of Proposition 6.3.1. By definition of $\Delta_{\min}^{\mathcal{A}}$, we can find (h', s', a') and an optimal policy π^* s.t. $\Delta_{h'}(s', a') = V_{h'}^*(s') - Q_{h'}^*(s', a') = \Delta_{\min}^{\mathcal{A}}$ and $d_{h'}^{\pi^*}(s', a') > 0$. We choose such (h', s', a') and π^* with the smallest $d_{h'}^{\pi^*}(s', a')$.

Let $\tilde{\pi}^*$ be a policy that only differ with π^* at (h', s'): for all h,

$$\tilde{\pi}_h^*(s) = \begin{cases} a' & \text{if } (h, s) = (h', s') \\ \pi_h^*(s) & \text{o.w.} \end{cases}$$

By definition, π^* and $\tilde{\pi}^*$ have the same state occupancy distribution up to step h' and the same state-action occupancy distribution up to step $h' - 1$:

$$d_h^{\pi^*}(s) = d_h^{\tilde{\pi}^*}(s) \quad \forall s \in \mathcal{S} \text{ and } h \leq h'. \tag{D.1}$$

Then the suboptimality of $\tilde{\pi}_h^*$ comes from the suboptimal action at (h', s'):

$$
\begin{aligned}
&\mathrm{SubOpt}(\tilde{\pi}^*) = V_{p_0}^* - V_{p_0}^{\tilde{\pi}^*} = V_{p_0}^{\pi^*} - V_{p_0}^{\tilde{\pi}^*} \\
=&\left(\mathbb{E}_{(s_h,a_h)\sim d_h^{\pi^*}}\left[\sum_{h=1}^{h'-1} r_h(s_h,a_h)\right] + \mathbb{E}_{s_{h'}\sim d_{h'}^{\pi^*}}\left[V_{h'}^{\pi^*}(s_{h'})\right]\right) \\
&-\left(\mathbb{E}_{(s_h,a_h)\sim d_h^{\tilde{\pi}^*}}\left[\sum_{h=1}^{h'-1} r_h(s_h,a_h)\right] + \mathbb{E}_{s_{h'}\sim d_{h'}^{\tilde{\pi}^*}}\left[V_{h'}^{\tilde{\pi}^*}(s_{h'})\right]\right) \\
=&\mathbb{E}_{s_{h'}\sim d_{h'}^{\pi^*}}\left[V_{h'}^{\pi^*}(s_{h'})\right] - \mathbb{E}_{s_{h'}\sim d_{h'}^{\tilde{\pi}^*}}\left[V_{h'}^{\tilde{\pi}^*}(s_{h'})\right] \\
&(\text{the rewards before step } h' \text{ cancel out because } \tilde{\pi}^* \text{ and } \pi^* \text{ takes the} \\
&\text{same actions before } h') \\
=&\sum_{s\in\mathcal{S}} d_{h'}^{\pi^*}(s)(V_{h'}^*(s) - Q_{h'}^*(s,\tilde{\pi}^*(s))) \\
&\left(\text{ by 1. (D.1); 2. rewrite } V_{h'}^{\tilde{\pi}^*} \text{ using } Q_{h'}^*\ \right) \\
=&d_{h'}^{\pi^*}(s')(V_{h'}^*(s') - Q_{h'}^*(s',a'))) \\
&(\text{ by definition of } \tilde{\pi}^*, V_{h'}^*(s) = Q_{h'}^*(s,\tilde{\pi}^*(s)) \text{ when } s \neq s' \ ) \\
=&d_{h'}^{\pi^*}(s')\Delta_{\min}^{\mathcal{A}} > 0 \\
&(\text{ by definition of } (s', a', h') \ )
\end{aligned}
$$

By definition of $\Delta_{\min}^{\Pi}$, we know $\Delta_{\min}^{\Pi} \leq \mathrm{SubOpt}(\tilde{\pi}^*)$, thus

$$\Delta_{\min}^{\Pi} \leq \mathrm{SubOpt}(\tilde{\pi}^*) = d_{h'}^{\pi^*}(s')\Delta_{\min}^{\mathcal{A}} \leq \Delta_{\min}^{\mathcal{A}}.$$

■

D.3 Proof of Theorem 6.4.1

We first define the following good events:

$$\mathcal{E}_{\mathrm{cb}} := \{\forall (s,a,h) \in \mathcal{S}\times\mathcal{A}\times[H] : |\hat{r}_h(s,a) - r_h(s,a)| \leq b_h^1(s,a),$$

$$\left|\widehat{\mathrm{PV}}_{h,s,a} - P_{h,s,a}^{\top}\underline{V}_{h+1}\right| \le b_h^2(s,a)\}$$
$$\mathcal{E}_{\mathrm{b}} := \Big\{\forall (s,a,h) \in \mathcal{C} : b_h^1(s,a) + b_h^2(s,a) \le b\Big\},$$

where $\mathcal{C} := \Big\{(s,a,h) \in \mathcal{S}\times\mathcal{A}\times[H] : \exists \pi^* \in \Pi^*, \text{ s.t. } d_h^{\pi^*}(s,a) > 0\Big\}$. By union and the condition of Theorem 6.4.1, $\Pr[\mathcal{E}_{\mathrm{cb}}] \ge 1-\delta$ and $\Pr[\mathcal{E}_{\mathrm{b}}] = 1$. Thus $\Pr[\mathcal{E}_{\mathrm{cb}} \cap \mathcal{E}_{\mathrm{b}}] \ge 1-\delta$ We now show that under event $\mathcal{E}_{\mathrm{cb}} \cap \mathcal{E}_{\mathrm{b}}$, $\mathrm{SubOpt}(\hat{\pi}) \le 2Hb$.

We first note that when the confidence bound is proper, then the $\underline{Q}$ is a pessimistic estimation for $Q^{\hat{\pi}}$:

Lemma D.3.1 (pessimistic estimation)**.** *Under event $\mathcal{E}_{\mathrm{cb}}$, for all $(s,a,h) \in \mathcal{S}\times\mathcal{A}\times[H]$,*

$$Q_h^{\hat{\pi}}(s,a) \ge \underline{Q}_h(s,a).$$

We defer Proof of Lemma D.3.1 to Section D.3.

We can use backward induction to upper bound the difference between V^* and $\underline{V}$:

Lemma D.3.2. *Under event $\mathcal{E}_{\mathrm{cb}} \cap \mathcal{E}_{\mathrm{b}}$, for all $(s,h) \in \mathcal{S}_h\times[H]$, we have:*

$$V_h^*(s) - \underline{V}_h(s) \le 2(H-h+1)b.$$

Recall that $\mathcal{S}_h := \Big\{s \in \mathcal{S} : \exists \pi^* \in \Pi^*, \text{ s.t. } d_h^{\pi^*}(s) > 0\Big\}$. We defer Proof of Lemma D.3.2 to Section D.3.

By Lemma D.3.1 and Lemma D.3.2, for all h and $s \in \mathcal{S}_h$, we have:

$$\begin{aligned} V_h^*(s) - Q_h^{\hat{\pi}}(s,\hat{\pi}_h(s)) \le& V_h^*(s) - \underline{Q}_h(s,\hat{\pi}_h(s)) = V_h^*(s) - \underline{V}_h(s) \\ \le& 2(H-h+1)b \le 2Hb \end{aligned} \tag{D.2}$$

By definition of $\mathcal{S}_1$, we have $\{s \in \mathcal{S} : p_0(s) > 0\} = \mathcal{S}_1$. Thus

$$\begin{aligned} \mathrm{SubOpt}(\hat{\pi}) =& \sum_{s\in\mathcal{S}} p_0(s)\Big(V_1^*(s) - V_1^{\hat{\pi}}(s)\Big) = \sum_{s:p_0(s)>0} p_0(s)\Big(V_1^*(s) - V_1^{\hat{\pi}}(s)\Big) \\ =& \sum_{s\in\mathcal{S}_1} p_0(s)\Big(V_1^*(s) - Q_1^{\hat{\pi}}(s,\hat{\pi}_1(s))\Big) \le 2Hb \quad \text{(by (D.2))} \end{aligned}$$

Proof of Lemma D.3.1

Proof of Lemma D.3.1. We prove by backward induction:

For $h = H$, $Q_H^{\hat{\pi}}(s,a) = r_H(s,a)$,

$$\underline{Q}_H(s,a) = \max\Big\{0, \hat{r}_H(s,a) - b_H^1(s,a)\Big\} \le \max\{0, r_H(s,a)\} = Q_H^{\hat{\pi}}(s,a),$$

where the inequality is because of the definition of $\mathcal{E}_{\text{cb}}$,

Suppose the statement holds for $h = k+1$, i.e.:

$$Q_{k+1}^{\hat{\pi}}(s,a) \ge \underline{Q}_{k+1}(s,a), \quad \forall s, a.$$

Then we have

$$V_{k+1}^{\hat{\pi}}(s) = Q_{k+1}^{\hat{\pi}}(s, \hat{\pi}(s)) \ge \underline{Q}_{k+1}(s, \hat{\pi}(s)) = \underline{V}_{k+1}(s), \quad \forall s. \tag{D.3}$$

For all s, a

$$\begin{aligned}
\underline{Q}_k(s,a) =& \max\Big\{0, \hat{r}_k(s,a) + \hat{P}_{k,s,a}^\top \underline{V}_{k+1} - b_k^1(s,a) - b_k^2(s,a)\Big\} \\
\le& \max\Big\{0, \hat{r}_k(s,a) - b_k^1(s,a)\Big\} + \max\Big\{0, \hat{P}_{k,s,a}^\top \underline{V}_{k+1} - b_k^2(s,a)\Big\} \\
\le& \max\{0, r_k(s,a)\} + \max\Big\{0, P_{k,s,a}^\top \underline{V}_{k+1}\Big\} \quad \text{(By definition of } \mathcal{E}_{\text{cb}}) \\
\le& r_k(s,a) + P_{k,s,a}^\top V_{k+1}^{\hat{\pi}} \quad \text{(By (D.3))} \\
=& Q_k^{\hat{\pi}}(s,a).
\end{aligned}$$

■

Proof of Lemma D.3.2

Proof of D.3.2. We prove by backward induction: Let π^* be an optimal policy.

For $h = H$, $\forall s \in \mathcal{S}_H$,

$$V_H^*(s) - \underline{V}_H(s) \le V_H^*(s) - \underline{Q}_H(s, \pi_H^*(s))$$

$$
\begin{aligned}
=&r_H(s,\pi_H^*(s)) - \max\Big\{0, \hat{r}_H(s,\pi_H^*(s)) - b_H^1(s,\pi_H^*(s))\Big\}\\
\leq&r_H(s,\pi_H^*(s)) - \hat{r}_H(s,\pi_H^*(s)) + b_H^1(s,\pi_H^*(s))\\
\leq&2b_H^1(s,\pi_H^*(s)) \quad \text{(By definition of } \mathcal{E}_{\text{cb}})\\
\leq&2b \quad \text{(By definition of } \mathcal{E}_{\text{b}}).
\end{aligned}
$$

Suppose the statement holds for $h = k+1$, i.e.

$$
V_{k+1}^*(s) - \underline{V}_{k+1}(s) \leq 2(H-k)b, \quad \forall s \in \mathcal{S}_{k+1}, \tag{D.4}
$$

then, $\forall s \in \mathcal{S}_k$, we have,

$$
\begin{aligned}
&V_k^*(s) - \underline{V}_k(s) = V_k^*(s) - \underline{Q}_k(s,\hat{\pi}_k(s)) \leq V_k^*(s) - \underline{Q}_k(s,\pi_k^*(s))\\
=&r_k(s,\pi_k^*(s)) + P_{k,s,\pi_k^*(s)}^\top V_{k+1}^*\\
&- \max\Big\{0, \hat{r}_k(s,\pi_k^*(s)) + \hat{P}_{k,s,\pi_k^*(s)}^\top \underline{V}_{k+1} - b_k^1(s,\pi_k^*(s)) - b_k^2(s,\pi_k^*(s))\Big\}\\
\leq&r_k(s,\pi_k^*(s)) - \hat{r}_k(s,\pi_k^*(s)) + P_{k,s,\pi_k^*(s)}^\top V_{k+1}^* - \hat{P}_{k,s,\pi_k^*(s)}^\top \underline{V}_{k+1}\\
&+ b_k^1(s,\pi_k^*(s)) + b_k^2(s,\pi_k^*(s))\\
=&r_k(s,\pi_k^*(s)) - \hat{r}_k(s,\pi_k^*(s)) + P_{k,s,\pi_k^*(s)}^\top \Big(V_{k+1}^* - \underline{V}_{k+1}\Big) + \Big(P_{k,s,\pi_k^*(s)} - \hat{P}_{k,s,\pi_k^*(s)}\Big)^\top \underline{V}_{k+1}\\
&+ b_k^1(s,\pi_k^*(s)) + b_k^2(s,\pi_k^*(s))\\
\leq&2b_k^1(s,\pi_k^*(s)) + 2b_k^2(s,\pi_k^*(s)) + P_{k,s,\pi_k^*(s)}^\top \Big(V_{k+1}^* - \underline{V}_{k+1}\Big) \quad \text{(By definition of } \mathcal{E}_{\text{cb}})\\
=&2b_k^1(s,\pi_k^*(s)) + 2b_k^2(s,\pi_k^*(s)) + \sum_{s'\in\mathcal{S}} P_k(s' \mid s,\pi_k^*(s))\Big(V_{k+1}^*(s') - \underline{V}_{k+1}(s')\Big)\\
=&2b_k^1(s,\pi_k^*(s)) + 2b_k^2(s,\pi_k^*(s)) + \sum_{s'\in\mathcal{S}_{k+1}} P_k(s' \mid s,\pi_k^*(s))\Big(V_{k+1}^*(s') - \underline{V}_{k+1}(s')\Big)\\
&\text{(By definition of } \mathcal{S}_{k+1})\\
\leq&2b + \sum_{s'\in\mathcal{S}_{k+1}} 2(H-k)bP_k(s' \mid s,\pi_k^*(s)) \quad \text{(by Definition of } \mathcal{E}_{\text{b}} \text{ and (D.4))}\\
=&2b + 2(H-k)b = 2(H-k+1)b.
\end{aligned}
$$

■

D.4 Proof of Theorem 6.4.2

We will show that $\mathrm{SubOpt}(\hat{\pi}) = 0$ under event $\mathcal{E}_{\mathrm{cb}} \cap \mathcal{E}_{\mathrm{b}}$ and condition $2Hb < \Delta^{\mathcal{A}}_{\min}$. By the proof of Theorem 6.4.1, $\mathcal{E}_{\mathrm{cb}} \cap \mathcal{E}_{\mathrm{b}}$ happens with probability at least $1 - \delta$ and we finish the proof.

By (D.2), for all h and $s \in \mathcal{S}_h$,

$$V^*_h(s) - Q^*_h(s, \hat{\pi}_h(s)) \le V^*_h(s) - Q^{\hat{\pi}}_h(s, \hat{\pi}_h(s)) \le 2Hb.$$

Because $2Hb < \Delta^{\mathcal{A}}_{\min}$, we have $V^*_h(s) - Q^*_h(s, \hat{\pi}_h(s)) < \Delta^{\mathcal{A}}_{\min}$. By the definition of $\Delta^{\mathcal{A}}_{\min}$, $\hat{\pi}_h(s)$ is an optimal action. This means, for all state s covered by some optimal policy, $\hat{\pi}$ chooses the optimal action. More specifically,

$$\forall h, \forall s \in \mathcal{S}_h, \exists \pi^* \in \Pi^*, \text{ s.t. } \pi^*_h(s) = \hat{\pi}_h(s). \tag{D.5}$$

and for all h,

$$V^*_h(s) - Q^*_h(s, \hat{\pi}_h(s)) = 0, \quad \forall s \in \mathcal{S}_h. \tag{D.6}$$

We now show that $\hat{\pi}$ only covers state visited by some optimal policy, i.e. for all $h \in [H]$,

$$\left\{ s \in \mathcal{S} : d^{\hat{\pi}}_h(s) > 0 \right\} \subseteq \mathcal{S}_h. \tag{D.7}$$

We prove by contradiction, suppose the following set is nonempty:

$$\left\{ (s, h) : d^{\hat{\pi}}_h(s) > 0, s \notin \mathcal{S}_h \right\} \neq \varnothing. \tag{D.8}$$

W.l.o.g., suppose $(s_{h'}, h')$ is the element with smallest h in RHS of (D.8),

We have $d^{\hat{\pi}}_{h'}(s_{h'}) > 0$ and

$$d^{\pi^*}_{h'}(s_{h'}) = 0 \quad \forall \pi^* \in \Pi^*. \tag{D.9}$$

Because $(s_{h'}, h')$ is such set with smallest h, we have for all $h < h'$ and $s \in \mathcal{S}$, if s is covered by $\hat{\pi}$, then s is also covered by some optimal policy, i.e. for all $h < h'$, if $d^{\hat{\pi}}_h(s) > 0$, then $s \in \mathcal{S}_h$.

Because $d^{\hat{\pi}}_{h'}(s_{h'}) > 0$, these exists a state-action sequence

$$s_1, a_1, s_2, a_2, \ldots, s_{h'-1}, a_{h'-1},$$

s.t.

$$d^{\hat{\pi}}_{h'}(s_{h'}) \geq p_0(s_1)P_1(s_2 \mid s_1, a_2) \cdots P_{h'-1}(s_{h'} \mid s_{h'-1}, a_{h'-1}) > 0$$

and

$$s_h \in \mathcal{S}_h, \hat{\pi}_h(s_h) = a_h \quad \forall h = 1, \ldots, h' - 1.$$

By (D.5), we can find an optimal policy $\pi^{**} \in \Pi^*$, s.t.

$$\pi^{**}_h(s_h) = \hat{\pi}_h(s_h) = a_h \quad \forall h = 1, \ldots, h' - 1.$$

Thus

$$d^{\pi^{**}}_{h'}(s_{h'}) \geq p_0(s_1)P_1(s_2 \mid s_1, a_2) \cdots P_{h'-1}(s_{h'} \mid s_{h'-1}, a_{h'-1}) > 0,$$

which contradicts with (D.9). Thus we have proved (D.7).

By performance difference lemma:

$$\begin{aligned}
\mathrm{SubOpt}(\hat{\pi}) =& V^*_{p_0} - V^{\hat{\pi}}_{p_0} = \sum_{h=1}^{H} \mathbb{E}_{s \sim d^{\hat{\pi}}_h(s)}[V^*_h(s) - Q^*_h(s, \hat{\pi}_h(s))] \\
=& \sum_{h=1}^{H} \sum_{s \in \mathcal{S}_h} d^{\hat{\pi}}_h(s)(V^*_h(s) - Q^*_h(s, \hat{\pi}_h(s))) \quad \text{(by (D.7))} \\
=& 0 \quad \text{(by (D.6))}.
\end{aligned}$$

D.5 Proof of Proposition 6.5.1

By the property of subGaussian random variable and Hoeffding's inequality: for all (s, a, h), with probability at least $1 - \frac{\delta}{2SAH}$

$$\left|\hat{r}^{\mathrm{emp}}_{h,s,a} - r_h(s, a)\right| \leq \sigma \sqrt{\frac{2 \log \frac{8SAH}{\delta}}{N_h(s, a)}} = b^{1,\mathrm{emp}}_{h,s,a}, \tag{D.10}$$

$$\left|\widehat{\mathrm{PV}}^{\mathrm{emp}}_{h,s,a} - P_{h,s,a}^{\top}\underline{V}_{h+1}\right| \le \|\underline{V}_{h+1}\|_\infty \sqrt{\frac{\log\frac{8SAH}{\delta}}{2N_h(s,a)}}, \tag{D.11}$$

where (D.11) is obtained by conditioning on $\underline{V}_{h+1}$.

Then we only need to show that $\|\underline{V}_{h+1}\|_\infty \le H-h \le H$. We show the statement by backward induction.

When $h=H$, $\underline{V}_{H+1}$ is set to be 0 by definition.

Suppose $\|\underline{V}_{h+1}\|_\infty \le H-h$ for $h=k$, i.e. $\|\underline{V}_{k+1}\|_\infty \le H-k$, then

$$\left|\left(\hat{P}_{k,s,a} - P_{k,s,a}\right)^{\top}\underline{V}_{k+1}\right| \le \|\underline{V}_{h+1}\|_\infty \sqrt{\frac{\log\frac{8SAH}{\delta}}{2N_h(s,a)}} \le H\sqrt{\frac{\log\frac{8SAH}{\delta}}{2N_h(s,a)}} = b_k^2(s,a). \tag{D.12}$$

Thus for all s, we have

$$\begin{aligned}
\underline{V}_k(s) =& \underline{Q}_k(s,\hat{\pi}_k(s)) \\
=& \max\left\{0, \hat{r}_k(s,a) + \hat{P}_{k,s,a}^{\top}\underline{V}_{k+1} - b_k^1(s,a) - b_k^2(s,a)\right\} \\
\le& \max\left\{0, \hat{r}_k(s,a) - b_k^1(s,a)\right\} + \max\left\{0, \hat{P}_{k,s,a}^{\top}\underline{V}_{k+1} - b_k^2(s,a)\right\} \\
\le& \max\{0, r_k(s,a)\} + \max\left\{0, P_{k,s,a}^{\top}\underline{V}_{k+1}\right\} \\
& \text{(the first term is because of (D.10), the second term is by (D.12))} \\
\le& 1 + \|\underline{V}_{k+1}\|_\infty \le H-k+1
\end{aligned}$$

I.e.

$$\left\|\underline{V}_{(k-1)+1}\right\|_\infty \le H-(k-1).$$

We have finished the induction.

D.6 Proof of Proposition 6.5.2

We define the following events:

$$\mathcal{E}_{\mathrm{cb}}^{\mathrm{emp}} := \{\forall(s,a,h) \in \mathcal{S}\times\mathcal{A}\times[H] : \left|\hat{r}^{\mathrm{emp}}_{h,s,a} - r_h(s,a)\right| \le b^{1,\mathrm{emp}}_{h,s,a},$$

$$\left|\widehat{\mathrm{PV}}^{\mathrm{emp}}_{h,s,a} - P^{\top}_{h,s,a}\underline{V}_{h+1}\right| \le b^{2,\mathrm{emp}}_{h,s,a}\}$$

$$\mathcal{E}^{\mathrm{emp}}_{\mathrm{b}} := \left\{\forall (s,a,h) \in \mathcal{C} : b^{1,\mathrm{emp}}_{h,s,a} + b^{2,\mathrm{emp}}_{h,s,a} \le 2(2\sigma + H)\frac{\log\frac{8SAH}{\delta}}{\sqrt{NP}}\right\},$$

recall that $\mathcal{C} := \left\{(s,a,h) \in \mathcal{S}\times\mathcal{A}\times[H] : \exists \pi^* \in \Pi^*, \text{ s.t. } d^{\pi^*}_h(s,a) > 0\right\}$. By Proposition 6.5.1 and union bound, $\Pr[\mathcal{E}^{\mathrm{emp}}_{\mathrm{cb}}] \ge 1 - \delta/2$. For $\mathcal{E}^{\mathrm{emp}}_{\mathrm{b}}$:

$$\begin{aligned}\Pr[\mathcal{E}^{\mathrm{emp}}_{\mathrm{b}}] &= \Pr\left[\forall (s,a,h) \in \mathcal{C} : \sigma\sqrt{\frac{2\log\frac{8SAH}{\delta}}{N_h(s,a)}} + H\sqrt{\frac{\log\frac{8SAH}{\delta}}{2N_h(s,a)}} \le 2(2\sigma+H)\frac{\log\frac{8SAH}{\delta}}{\sqrt{NP}}\right] \\ &= \Pr\left[\forall (s,a,h) \in \mathcal{C} : \frac{1}{N_h(s,a)} \le \frac{8\log\frac{8SAH}{\delta}}{NP}\right] \ge 1 - \frac{\delta}{2},\end{aligned}$$

where the inequality is by Assumption 6.5.2, Lemma D.8.1 and union bound. Thus $\Pr[\mathcal{E}^{\mathrm{emp}}_{\mathrm{cb}} \cap \mathcal{E}^{\mathrm{emp}}_{\mathrm{b}}] \ge 1 - \delta$. Then we finish the proof by using the exact same argument in the Proof of Theorem 6.4.2, but with events $\mathcal{E}^{\mathrm{emp}}_{\mathrm{cb}} \cap \mathcal{E}^{\mathrm{emp}}_{\mathrm{b}}$.

D.7 Theorem 6.5.4

We follow the main steps in Proof of Theorem 1 in Lugosi and Mendelson (2021) but apply a novel variant of Bernstein's inequality for heavy-tailed distribution

We use the following notations:

- Trimming level: $\tilde{\epsilon} = 8\epsilon + \frac{24}{N}\log\frac{8}{\delta}$;
- Centered versions of X: $\overline{X} := X - \mu$;
- The quantile: $Q_p(\overline{X}) := \sup\left\{M \in \mathbb{R} : \mathbb{P}\left[\overline{X} \ge M\right] \ge 1 - p\right\}$, for $0 < p < 1$;
- The clean sample: $Y_1, \ldots, Y_N$;
- The corrupted sample: $X_1, \ldots, X_N$;
- error term:

$$\overline{\mathcal{E}}(\eta, X) := \max\left\{\mathbb{E}\left[\left|\overline{X} - Q_{\eta/2}(\overline{X})\right| \mathbb{I}\left\{\overline{X} \le Q_{\eta/2}(\overline{X})\right\}\right], \tag{D.13}$$

$$\mathbb{E}\Big[\Big|\overline{X} - Q_{1-\eta/2}(\overline{X})\Big| \mathbb{I}\Big\{\overline{X} \geq Q_{1-\eta/2}(\overline{X})\Big\}\Big]\Big\} \tag{D.14}$$

We define the following concentration events:

$$\mathcal{E}_1 := \mathcal{E}_1^1 \cap \mathcal{E}_1^2 \cap \mathcal{E}_1^3 \cap \mathcal{E}_1^4 \tag{D.15}$$

$$\begin{aligned}\mathcal{E}_2 :=& \Bigg\{ \left| \frac{2}{N} \sum_{i=N/2+1}^{N} \phi_{\alpha,\beta}(Y_i) - \mu \right| \\ \leq& 2\sigma\Big(\frac{2A_\gamma}{N}\log\frac{4}{\delta}\Big)^{\frac{\gamma}{1+\gamma}} + 2\Big(\Big|Q_{1-\tilde{\epsilon}/2}(\overline{X})\Big| + \Big|Q_{\tilde{\epsilon}/2}(\overline{X})\Big|\Big)\frac{2A_\gamma}{N}\log\frac{4}{\delta} + \overline{\mathcal{E}}(4\tilde{\epsilon}, X)\Bigg\}\end{aligned} \tag{D.16}$$

where α, β are the clipping points chosen in the algorithm (depending on the second half of the corrupted samples: $X_{N/2+1}, \ldots, X_N$) and

$$\mathcal{E}_1^1 := \Big\{ \Big| i : Y_i \geq \mu + Q_{1-2\tilde{\epsilon}}(\overline{X}), i = N/2+1, \ldots, N \Big| \geq \frac{3}{4}\tilde{\epsilon}N \Big\} \tag{D.17}$$

$$\mathcal{E}_1^2 := \Big\{ \Big| i : Y_i \leq \mu + Q_{1-\tilde{\epsilon}/2}(\overline{X}), i = N/2+1, \ldots, N \Big| \geq \Big(1 - \frac{3}{4}\tilde{\epsilon}\Big)N/2 \Big\} \tag{D.18}$$

$$\mathcal{E}_1^3 := \Big\{ \Big| i : Y_i \leq \mu + Q_{2\tilde{\epsilon}}(\overline{X}), i = N/2+1, \ldots, N \Big| \geq \frac{3}{4}\tilde{\epsilon}N \Big\} \tag{D.19}$$

$$\mathcal{E}_1^4 := \Big\{ \Big| i : Y_i \geq \mu + Q_{\tilde{\epsilon}/2}(\overline{X}), i = N/2+1, \ldots, N \Big| \geq \Big(1 - \frac{3}{4}\tilde{\epsilon}\Big)N/2 \Big\} \tag{D.20}$$

We first show that $\mathbb{P}[\mathcal{E}_1 \cap \mathcal{E}_2] \geq 1 - \delta$. Let $U := \mathbb{I}\Big\{\overline{X} \geq Q_{1-2\tilde{\epsilon}}(\overline{X})\Big\}$. By definition of quantile, we have: $\mathbb{P}\Big[\overline{X} \geq Q_{1-2\tilde{\epsilon}}\Big] = 2\tilde{\epsilon}$ and

$$\mathbb{V}[U] \leq \mathbb{E}\Big[U^2\Big] = \mathbb{P}\Big[\overline{X} \geq Q_{1-2\tilde{\epsilon}}\Big] = 2\tilde{\epsilon}. \tag{D.21}$$

By using Bernstein's inequality on U, we can show that $\mathbb{P}[\mathcal{E}_1^1] \geq 1 - \exp(-\tilde{\epsilon}N/24)$. Similarly, we can show that $\mathbb{P}[\mathcal{E}_1^2], \mathbb{P}[\mathcal{E}_1^3], \mathbb{P}[\mathcal{E}_1^4] \geq 1 - \exp(-\tilde{\epsilon}N/24)$. By union bound,

$$\mathbb{P}[\mathcal{E}_1] \geq 1 - 4\exp(-\tilde{\epsilon}N/24) \geq 1 - \frac{\delta}{2}. \tag{D.22}$$

We can bound the clipping point α, β under event $\mathcal{E}_1$: First note that $\epsilon \leq \tilde{\epsilon}/8$.

Because $X_{N/2+1}, \ldots, X_N$ and $Y_{N/2+1}, \ldots, Y_N$ differ by at most ϵN points, we have:

$$\Big|i : X_i \geq \mu + Q_{1-2\tilde{\epsilon}}(\overline{X}), i = N/2+1, \ldots, N\Big| \geq \frac{3}{4}\tilde{\epsilon}N - \epsilon N \geq \frac{3}{4}\tilde{\epsilon}N - \frac{\tilde{\epsilon}}{8}N \geq \frac{\tilde{\epsilon}N}{2} \tag{D.23}$$

$$\Big|i : X_i \leq \mu + Q_{1-\tilde{\epsilon}/2}(\overline{X}), i = N/2+1, \ldots, N\Big| \geq \Big(1 - \frac{3}{4}\tilde{\epsilon}\Big)N/2 - \epsilon N \tag{D.24}$$

$$\geq \Big(1 - \frac{3}{4}\tilde{\epsilon}\Big)N/2 - \frac{\tilde{\epsilon}}{8}N = \frac{(1-\tilde{\epsilon})N}{2} \tag{D.25}$$

Because β is chosen to be the $(1-\tilde{\epsilon})N/2$ largest of $X_{N/2+1}, \ldots, X_N$, we have:

$$\mu + Q_{1-2\tilde{\epsilon}}(\overline{X}) \leq \beta \leq \mu + Q_{1-\tilde{\epsilon}/2}(\overline{X}). \tag{D.26}$$

Similarly, we can show that

$$\mu + Q_{\tilde{\epsilon}/2}(\overline{X}) \leq \alpha \leq \mu + Q_{2\tilde{\epsilon}}(\overline{X}). \tag{D.27}$$

We now show that $\mathcal{E}_2|\mathcal{E}_1$ happens with high probability: We first show that the expectation of the clipped sample is close to μ:

$$\mathbb{E}[\phi_{\alpha,\beta}(Y_i)] \leq \mathbb{E}\Big[\phi_{\mu+Q_{2\tilde{\epsilon}}(\overline{X}),\infty}(Y_i)\Big] \tag{D.28}$$

$$= \mathbb{E}\Big[Y_i \mathbb{I}\Big\{Y_i \geq \mu + Q_{2\tilde{\epsilon}}(\overline{X})\Big\}\Big] + \mathbb{E}\Big[\Big(\mu + Q_{2\tilde{\epsilon}}(\overline{X})\Big)\mathbb{I}\Big\{Y_i < \mu + Q_{2\tilde{\epsilon}}(\overline{X})\Big\}\Big] \tag{D.29}$$

$$= \mathbb{E}\Big[Y_i \mathbb{I}\Big\{Y_i \geq \mu + Q_{2\tilde{\epsilon}}(\overline{X})\Big\}\Big] + \mathbb{E}\Big[Y_i \mathbb{I}\Big\{Y_i < \mu + Q_{2\tilde{\epsilon}}(\overline{X})\Big\}\Big] \tag{D.30}$$

$$+ \mathbb{E}\Big[\Big(\mu + Q_{2\tilde{\epsilon}}(\overline{X}) - Y_i\Big)\mathbb{I}\Big\{Y_i < \mu + Q_{2\tilde{\epsilon}}(\overline{X})\Big\}\Big] \tag{D.31}$$

$$\leq \mu + \mathbb{E}\Big[\Big|\overline{X} - Q_{2\tilde{\epsilon}}(\overline{X})\Big| \mathbb{I}\Big\{\overline{X} < Q_{2\tilde{\epsilon}}(\overline{X})\Big\}\Big] \tag{D.32}$$

Similarly,

$$\mathbb{E}[\phi_{\alpha,\beta}(Y_i)] \geq \mathbb{E}\Big[\phi_{-\infty,\mu+Q_{1-2\tilde{\epsilon}}(\overline{X})}(Y_i)\Big] \tag{D.33}$$

$$=\mathbb{E}\Big[Y_i\mathbb{I}\Big\{Y_i \le \mu + Q_{1-2\tilde{\epsilon}}(\overline{X})\Big\}\Big] + \mathbb{E}\Big[\Big(\mu + Q_{1-2\tilde{\epsilon}}(\overline{X})\Big)\mathbb{I}\Big\{Y_i > \mu + Q_{1-2\tilde{\epsilon}}(\overline{X})\Big\}\Big] \tag{D.34}$$

$$=\mathbb{E}\Big[Y_i\mathbb{I}\Big\{Y_i \le \mu + Q_{1-2\tilde{\epsilon}}(\overline{X})\Big\}\Big] + \mathbb{E}\Big[Y_i\mathbb{I}\Big\{Y_i > \mu + Q_{1-2\tilde{\epsilon}}(\overline{X})\Big\}\Big] \tag{D.35}$$

$$+ \mathbb{E}\Big[\Big(\mu + Q_{1-2\tilde{\epsilon}}(\overline{X}) - Y_i\Big)\mathbb{I}\Big\{Y_i > \mu + Q_{2\tilde{\epsilon}}(\overline{X})\Big\}\Big] \tag{D.36}$$

$$\ge \mu - \mathbb{E}\Big[\Big|\overline{X} - Q_{1-2\tilde{\epsilon}}(\overline{X})\Big|\,\mathbb{I}\Big\{\overline{X} > Q_{1-2\tilde{\epsilon}}(\overline{X})\Big\}\Big] \tag{D.37}$$

Thus

$$|\mathbb{E}[\phi_{\alpha,\beta}(Y_i)] - \mu| \le \overline{\mathcal{E}}(4\tilde{\epsilon}, X) \tag{D.38}$$

The $(1+\gamma)$-th moment is bounded by:

$$\mathbb{E}\Big[|\phi_{\alpha,\beta}(Y_i) - \mathbb{E}[\phi_{\alpha,\beta}(Y_i)]|^{1+\gamma}\Big] \tag{D.39}$$

$$\le \mathbb{E}\Big[|Y_i - \mathbb{E}[\phi_{\alpha,\beta}(Y_i)]|^{1+\gamma}\Big] = \mathbb{E}\Big[|Y_i - \mu + \mu - \mathbb{E}[\phi_{\alpha,\beta}(Y_i)]|^{1+\gamma}\Big] \tag{D.40}$$

$$\le 2^{1+\gamma}\mathbb{E}\Big[|Y_i - \mu|^{1+\gamma}\Big] + 2^{1+\gamma}\,|\mu - \mathbb{E}[\phi_{\alpha,\beta}(Y_i)]|^{1+\gamma} \tag{D.41}$$

$$\le 2^{1+\gamma}\sigma^{1+\gamma} + 2^{1+\gamma}\overline{\mathcal{E}}(4\tilde{\epsilon}, X)^{1+\gamma} \le 2\cdot 2^{1+\gamma}\Big(\sigma + \overline{\mathcal{E}}(4\tilde{\epsilon}, X)\Big)^{1+\gamma} \tag{D.42}$$

$$\le \Big(4\Big(\sigma + \overline{\mathcal{E}}(4\tilde{\epsilon}, X)\Big)\Big)^{1+\gamma} \tag{D.43}$$

Conditioning on $\mathcal{E}_1$, $\phi_{\alpha,\beta}(Y_i)$ is a random variable bounded by

$$\mu + Q_{\tilde{\epsilon}/2}(\overline{X}) \le \alpha \le \phi_{\alpha,\beta}(Y_i) \le \beta \le \mu + Q_{1-\tilde{\epsilon}/2}(\overline{X}). \tag{D.44}$$

Thus, with probability 1, we have

$$|\phi_{\alpha,\beta}(Y_i) - \mathbb{E}[\phi_{\alpha,\beta}(Y_i)]| \le Q_{1-\tilde{\epsilon}/2}(\overline{X}) - Q_{\tilde{\epsilon}/2}(\overline{X}) \le \Big|Q_{1-\tilde{\epsilon}/2}(\overline{X})\Big| + \Big|Q_{\tilde{\epsilon}/2}(\overline{X})\Big|. \tag{D.45}$$

By Corollary D.7.1, there exists A_γ, s.t. with probability at least $1 - \frac{\delta}{2}$ we have

$$\left|\frac{2}{N}\sum_{i=1}^{N/2}(\phi_{\alpha,\beta}(Y_i) - \mathbb{E}[\phi_{\alpha,\beta}(Y_i)])\right|$$

$$\le 2\sigma\left(\frac{2A_\gamma}{N}\log\frac{4}{\delta}\right)^{\frac{\gamma}{1+\gamma}} + 2\Big(\Big|Q_{1-\tilde{\epsilon}/2}(\overline{X})\Big| + \Big|Q_{\tilde{\epsilon}/2}(\overline{X})\Big|\Big)\frac{2A_\gamma}{N}\log\frac{4}{\delta} \tag{D.46}$$

By (D.38) and (D.46),

$$\left|\frac{2}{N}\sum_{i=1}^{N/2}\phi_{\alpha,\beta}(Y_i)-\mu\right| \le \left|\frac{2}{N}\sum_{i=1}^{N/2}(\phi_{\alpha,\beta}(Y_i)-\mathbb{E}[\phi_{\alpha,\beta}(Y_i)])\right| + |\mathbb{E}[\phi_{\alpha,\beta}(Y_N)]-\mu| \tag{D.47}$$

$$\le 2\sigma\left(\frac{2A_\gamma}{N}\log\frac{4}{\delta}\right)^{\frac{\gamma}{1+\gamma}} + 2\left(\left|Q_{1-\tilde{\epsilon}/2}(\overline{X})\right| + \left|Q_{\tilde{\epsilon}/2}(\overline{X})\right|\right)\frac{2A_\gamma}{N}\log\frac{4}{\delta} + \overline{\mathcal{E}}(4\tilde{\epsilon}, X) \tag{D.48}$$

Now we've shown that $\mathcal{E}_1 \cap \mathcal{E}_2$ happens with probability at least $1-\delta$. In the following, we will upper bound the estimation error of the trimmed mean estimation under $\mathcal{E}_1 \cap \mathcal{E}_2$. Because clipped clean samples $\phi_{\alpha,\beta}(Y_1), \ldots, \phi_{\alpha,\beta}(Y_{N/2})$ and clipped corrupted samples $\phi_{\alpha,\beta}(X_1), \ldots, \phi_{\alpha,\beta}(X_{N/2})$ are all bounded and differ by at most ϵN entries, we have

$$\left|\frac{2}{N}\sum_{i=1}^{N/2}\phi_{\alpha,\beta}(Y_i) - \frac{2}{N}\sum_{i=1}^{N/2}\phi_{\alpha,\beta}(X_i)\right| \le 2\epsilon\left(\left|Q_{1-\tilde{\epsilon}/2}(\overline{X})\right| + \left|Q_{\tilde{\epsilon}/2}(\overline{X})\right|\right). \tag{D.49}$$

Thus the estimation error of trimmed mean can be bounded by

$$\left|\frac{2}{N}\sum_{i=1}^{N/2}\phi_{\alpha,\beta}(X_i)-\mu\right| \le \left|\frac{2}{N}\sum_{i=1}^{N/2}\phi_{\alpha,\beta}(Y_i) - \frac{2}{N}\sum_{i=1}^{N/2}\phi_{\alpha,\beta}(X_i)\right| + \left|\frac{2}{N}\sum_{i=1}^{N/2}\phi_{\alpha,\beta}(Y_i)-\mu\right| \tag{D.50}$$

$$\le 2\sigma\left(\frac{2A_\gamma}{N}\log\frac{4}{\delta}\right)^{\frac{\gamma}{1+\gamma}} + 2\left(\left|Q_{1-\tilde{\epsilon}/2}(\overline{X})\right| + \left|Q_{\tilde{\epsilon}/2}(\overline{X})\right|\right)\left(\frac{2A_\gamma}{N}\log\frac{4}{\delta}+\epsilon\right) + \overline{\mathcal{E}}(4\tilde{\epsilon}, X) \tag{D.51}$$

$$\le 2\sigma\left(\frac{2A_\gamma}{N}\log\frac{4}{\delta}\right)^{\frac{\gamma}{1+\gamma}} + A_\gamma\left(\left|Q_{1-\tilde{\epsilon}/2}(\overline{X})\right| + \left|Q_{\tilde{\epsilon}/2}(\overline{X})\right|\right)\frac{\tilde{\epsilon}}{2} + \overline{\mathcal{E}}(4\tilde{\epsilon}, X) \tag{D.52}$$

By Lemma D.7.2,

$$A_\gamma\left(\left|Q_{1-\tilde{\epsilon}/2}(\overline{X})\right| + \left|Q_{\tilde{\epsilon}/2}(\overline{X})\right|\right)\frac{\tilde{\epsilon}}{2} \le 2A_\gamma\sigma\left(\frac{\tilde{\epsilon}}{2}\right)^{\frac{\gamma}{1+\gamma}} \tag{D.53}$$

and

$$\begin{aligned}
&\mathbb{E}\Big[\Big|\overline{X} - Q_{2\tilde{\epsilon}}(\overline{X})\Big| \mathbb{I}\Big\{\overline{X} \leq Q_{2\tilde{\epsilon}}(\overline{X})\Big\}\Big] && \text{(D.54)}\\
\leq &\mathbb{E}\Big[\Big|\overline{X}\Big| \mathbb{I}\Big\{\overline{X} \leq Q_{2\tilde{\epsilon}}(\overline{X})\Big\}\Big] + \mathbb{E}\Big[\Big|Q_{2\tilde{\epsilon}}(\overline{X})\Big| \mathbb{I}\Big\{\overline{X} \leq Q_{2\tilde{\epsilon}}(\overline{X})\Big\}\Big] && \text{(D.55)}\\
\leq &\Big(\mathbb{E}\Big[\Big|\overline{X}\Big|^{1+\gamma}\Big]\Big)^{\frac{1}{1+\gamma}} \left(\mathbb{E}\left[\mathbb{I}\Big\{\overline{X} \leq Q_{2\tilde{\epsilon}}(\overline{X})\Big\}^{\frac{1+\gamma}{\gamma}}\right]\right)^{\frac{\gamma}{1+\gamma}} + \Big|Q_{2\tilde{\epsilon}}(\overline{X})\Big| 2\tilde{\epsilon} && \text{(D.56)}\\
\leq &2\sigma(2\tilde{\epsilon})^{\frac{\gamma}{1+\gamma}} < 4\sigma\tilde{\epsilon}^{\frac{\gamma}{1+\gamma}} && \text{(D.57)}
\end{aligned}$$

Similarly,

$$\mathbb{E}\Big[\Big|\overline{X} - Q_{1-2\tilde{\epsilon}}(\overline{X})\Big| \mathbb{I}\Big\{\overline{X} \geq Q_{1-2\tilde{\epsilon}}(\overline{X})\Big\}\Big] \leq 4\sigma\tilde{\epsilon}^{\frac{\gamma}{1+\gamma}} \tag{D.58}$$

Thus

$$\overline{\mathcal{E}}(4\tilde{\epsilon}, X) \leq 4\sigma\tilde{\epsilon}^{\frac{\gamma}{1+\gamma}} \tag{D.59}$$

We can further upper bound the estimation error by:

$$\begin{aligned}
\left|\frac{2}{N}\sum_{i=1}^{N/2} \phi_{\alpha,\beta}(X_i) - \mu\right| \leq &2\sigma\Big(\frac{2A_\gamma}{N}\log\frac{4}{\delta}\Big)^{\frac{\gamma}{1+\gamma}} + A_\gamma\Big(\Big|Q_{1-\tilde{\epsilon}/2}(\overline{X})\Big| + \Big|Q_{\tilde{\epsilon}/2}(\overline{X})\Big|\Big)\frac{\tilde{\epsilon}}{2} && \text{(D.60)}\\
&+ \overline{\mathcal{E}}(4\tilde{\epsilon}, X) && \text{(D.61)}\\
\leq &4A_\gamma\sigma\Big(\frac{1}{N}\log\frac{4}{\delta}\Big)^{\frac{\gamma}{1+\gamma}} + 2A_\gamma\sigma\Big(\frac{\tilde{\epsilon}}{2}\Big)^{\frac{\gamma}{1+\gamma}} + 4\sigma\tilde{\epsilon}^{\frac{\gamma}{1+\gamma}} && \text{(D.62)}\\
\leq &4A_\gamma\sigma\Big(\frac{1}{N}\log\frac{4}{\delta}\Big)^{\frac{\gamma}{1+\gamma}} + 8A_\gamma\sigma\tilde{\epsilon}^{\frac{\gamma}{1+\gamma}} && \text{(D.63)}\\
= &4A_\gamma\sigma\Big(\frac{1}{N}\log\frac{4}{\delta}\Big)^{\frac{\gamma}{1+\gamma}} + 8A_\gamma\sigma\Big(8\epsilon + \frac{24}{N}\log\frac{8}{\delta}\Big)^{\frac{\gamma}{1+\gamma}} && \text{(D.64)}\\
\leq &4A_\gamma\sigma\Big(\frac{1}{N}\log\frac{4}{\delta}\Big)^{\frac{\gamma}{1+\gamma}} + 8A_\gamma\sigma(16\epsilon)^{\frac{\gamma}{1+\gamma}} + 8A_\gamma\sigma\Big(\frac{48}{N}\log\frac{8}{\delta}\Big)^{\frac{\gamma}{1+\gamma}} && \text{(D.65)}\\
\leq &8A_\gamma\sigma(16\epsilon)^{\frac{\gamma}{1+\gamma}} + 16A_\gamma\sigma\Big(\frac{48}{N}\log\frac{8}{\delta}\Big)^{\frac{\gamma}{1+\gamma}} && \text{(D.66)}\\
\leq &128A_\gamma\sigma\epsilon^{\frac{\gamma}{1+\gamma}} + 768A_\gamma\sigma\Big(\frac{1}{N}\log\frac{8}{\delta}\Big)^{\frac{\gamma}{1+\gamma}} && \text{(D.67)}
\end{aligned}$$

Auxiliary Lemmas

Lemma D.7.1 (Bernstein's inequality under weak moment assumption)**.** *Suppose X_j, $j = 1, \ldots, n$ is a sequence of independent zero-mean random variable bounded by $|X_j| \leq M$ and there exists $\alpha \in (0, 1]$, s.t.*

$$\mathbb{E}\,|X_j|^{1+\gamma} \leq \sigma^{1+\gamma}, \text{ for all } j = 1, \ldots, n. \tag{D.68}$$

then there exists $A_\gamma \geq 1$ (depending only on α) s.t.:

$$\mathbb{P}\left(\frac{1}{n}\sum_{j=1}^{n} X_j > t\right) \leq \exp\left\{-\frac{n}{A_\gamma}\frac{t^{\frac{\gamma+1}{\gamma}}}{\sigma^{\frac{1+\gamma}{\gamma}} + Mt^{\frac{1}{\gamma}}}\right\}. \tag{D.69}$$

This proof is based on the proof of standard Bernstein's inequality in the notes at https://www.stat.cmu.edu/ larry/=sml/Concentration.pdf.

Proof. For any $s > 0$ and any j, we have

$$\begin{aligned}
\mathbb{E}\left[e^{sX_j}\right] =& \mathbb{E}\left[1 + sX_j + \sum_{i=2}^{\infty}\frac{s^i X_j^i}{i!}\right] = 1 + \sum_{i=2}^{\infty}\frac{s^i \mathbb{E}\left[X_j^{1+\gamma} X_j^{i-1-\gamma}\right]}{i!} && \text{(D.70)}\\
\leq& 1 + \sum_{i=2}^{\infty}\frac{s^i M^{i-1-\gamma}\mathbb{E}\left[|X_j|^{1+\gamma}\right]}{i!} \leq 1 + \frac{\sigma^{1+\gamma}}{M^{1+\gamma}}\sum_{i=2}^{\infty}\frac{s^i M^i}{i!} && \text{(D.71)}\\
=& 1 + \frac{\sigma^{1+\gamma}}{M^{1+\gamma}}\left(e^{sM} - 1 - sM\right) && \text{(D.72)}\\
\leq& \exp\left\{\frac{\sigma^{1+\gamma}}{M^{1+\gamma}}\left(e^{sM} - 1 - sM\right)\right\}. && \text{(D.73)}
\end{aligned}$$

For any $t > 0$, we have

$$\begin{aligned}
\mathbb{P}\left(\frac{1}{n}\sum_{j=1}^{n} X_j > t\right) =& \mathbb{P}\left(\sum_{j=1}^{n} X_j > nt\right) = \mathbb{P}\left(\exp\left(s\sum_{j=1}^{n} X_j\right) > \exp(snt)\right) && \text{(D.74)}\\
\leq& \exp(-snt)\mathbb{E}\left[\exp\left(s\sum_{j=1}^{n} X_j\right)\right] = \exp(-snt)\prod_{j=1}^{n}\mathbb{E}[\exp(sX_j)] && \text{(D.75)}
\end{aligned}$$

$$\leq \exp(-snt)\exp\left\{\frac{n\sigma^{1+\gamma}}{M^{1+\gamma}}\left(e^{sM}-1-sM\right)\right\} \tag{D.76}$$

$$\leq \exp\left\{-\frac{n\sigma^{1+\gamma}}{M^{1+\gamma}}\left(\frac{M^{1+\gamma}}{\sigma^{1+\gamma}}st - e^{sM}+1+sM\right)\right\} \tag{D.77}$$

Let

$$s = \frac{1}{M}\log\left(1+\frac{M^{\gamma}t}{\sigma^{1+\gamma}}\right) \tag{D.78}$$

we have

$$\exp\left\{-\frac{n\sigma^{1+\gamma}}{M^{1+\gamma}}\left(\frac{M^{1+\gamma}}{\sigma^{1+\gamma}}st - e^{sM}+1+sM\right)\right\} \tag{D.79}$$

$$= \exp\left\{-\frac{n\sigma^{1+\gamma}}{M^{1+\gamma}}\left(\frac{M^{1+\gamma}}{\sigma^{1+\gamma}}t\frac{1}{M}\log\left(1+\frac{M^{\gamma}t}{\sigma^{1+\gamma}}\right)-1-\frac{M^{\gamma}t}{\sigma^{1+\gamma}}+1+\log\left(1+\frac{M^{\gamma}t}{\sigma^{1+\gamma}}\right)\right)\right\} \tag{D.80}$$

$$= \exp\left\{-\frac{n\sigma^{1+\gamma}}{M^{1+\gamma}}\left(\left(1+\frac{M^{\gamma}t}{\sigma^{1+\gamma}}\right)\log\left(1+\frac{M^{\gamma}t}{\sigma^{1+\gamma}}\right)-\frac{M^{\gamma}t}{\sigma^{1+\gamma}}\right)\right\} \tag{D.81}$$

$$=: \exp\left\{-\frac{n\sigma^{1+\gamma}}{M^{1+\gamma}}h\left(\frac{M^{\gamma}t}{\sigma^{1+\gamma}}\right)\right\}, \tag{D.82}$$

where

$$h(x) = (1+x)\log(1+x) - x \tag{D.83}$$

There exists a $A_\gamma \geq 1$, s.t.

$$(1+x)\log(1+x) - x \geq \frac{1}{A_\gamma}\frac{x^{\frac{\gamma+1}{\gamma}}}{1+x^{\frac{1}{\gamma}}} \tag{D.84}$$

Then

$$\mathbb{P}\left(\frac{1}{n}\sum_{j=1}^{n}X_j > t\right) \leq \exp\left\{-\frac{n\sigma^{1+\gamma}}{M^{1+\gamma}}h\left(\frac{M^{\gamma}t}{\sigma^{1+\gamma}}\right)\right\} \tag{D.85}$$

$$\leq \exp\left\{-\frac{n\sigma^{1+\gamma}}{M^{1+\gamma}}\frac{1}{A_\gamma}\frac{\left(\frac{M^{\gamma}t}{\sigma^{1+\gamma}}\right)^{\frac{\gamma+1}{\gamma}}}{1+\left(\frac{M^{\gamma}t}{\sigma^{1+\gamma}}\right)^{\frac{1}{\gamma}}}\right\} \tag{D.86}$$

$$= \exp\left\{ -\frac{n}{A_\gamma} \frac{t^{\frac{\gamma+1}{\gamma}} \left(\frac{1}{\sigma^{1+\gamma}}\right)^{\frac{1}{\gamma}}}{1 + \left(\frac{M^\gamma t}{\sigma^{1+\gamma}}\right)^{\frac{1}{\gamma}}} \right\} \tag{D.87}$$

$$= \exp\left\{ -\frac{n}{A_\gamma} \frac{t^{\frac{\gamma+1}{\gamma}}}{\sigma^{\frac{1+\gamma}{\gamma}} + M t^{\frac{1}{\gamma}}} \right\} \tag{D.88}$$

■

Corollary D.7.1. *Suppose X_j, $j = 1, \ldots, n$ is a sequence of independent zero-mean random variable bounded by $|X_j| \leq M$ and there exists $\alpha \in (0, 1]$, s.t.*

$$\mathbb{E}\,|X_j|^{1+\gamma} \leq \sigma^{1+\gamma}, \text{ for all } j = 1, \ldots, n. \tag{D.89}$$

then there exists $A_\gamma \geq 1$ (depending only on α) s.t.: W.p. $\geq 1 - \delta$,

$$\left| \frac{1}{n} \sum_{j=1}^{n} X_j \right| \leq 2\sigma \left(\frac{A_\gamma}{n} \log \frac{2}{\delta} \right)^{\frac{\gamma}{1+\gamma}} + 2M \frac{A_\gamma}{n} \log \frac{2}{\delta} \tag{D.90}$$

Proof. Let

$$t = 2\sigma \left(\frac{A_\gamma}{n} \log \frac{2}{\delta} \right)^{\frac{\gamma}{1+\gamma}} + 2M \frac{A_\gamma}{n} \log \frac{2}{\delta} \tag{D.91}$$

If $\sigma^{\frac{1+\gamma}{\gamma}} \leq M t^{\frac{1}{\gamma}}$,

$$\exp\left\{ -\frac{n}{A_\gamma} \frac{t^{\frac{\gamma+1}{\gamma}}}{\sigma^{\frac{1+\gamma}{\gamma}} + M t^{\frac{1}{\gamma}}} \right\} \leq \exp\left\{ -\frac{n}{A_\gamma} \frac{t^{\frac{\gamma+1}{\gamma}}}{2M t^{\frac{1}{\gamma}}} \right\} = \exp\left\{ -\frac{n}{A_\gamma} \frac{t}{2M} \right\} \tag{D.92}$$

$$\leq \exp\left\{ -\frac{n}{A_\gamma} \frac{1}{2M} 2M \frac{A_\gamma}{n} \log \frac{2}{\delta} \right\} = \frac{\delta}{2} \tag{D.93}$$

If $\sigma^{\frac{1+\gamma}{\gamma}} \geq M t^{\frac{1}{\gamma}}$,

$$\exp\left\{ -\frac{n}{A_\gamma} \frac{t^{\frac{\gamma+1}{\gamma}}}{\sigma^{\frac{1+\gamma}{\gamma}} + M t^{\frac{1}{\gamma}}} \right\} \leq \exp\left\{ -\frac{n}{A_\gamma} \frac{t^{\frac{\gamma+1}{\gamma}}}{2\sigma^{\frac{1+\gamma}{\gamma}}} \right\} \tag{D.94}$$

$$\leq \exp\left\{-\frac{n}{A_\gamma}\frac{1}{2\sigma^{\frac{1+\gamma}{\gamma}}}2^{\frac{\gamma+1}{\gamma}}\sigma^{\frac{\gamma+1}{\gamma}}\left(\frac{A_\gamma}{n}\log\frac{2}{\delta}\right)^{\frac{\gamma}{1+\gamma}\frac{\gamma+1}{\gamma}}\right\} \tag{D.95}$$

$$\leq \exp\left\{-\frac{n}{A_\gamma}\left(\frac{A_\gamma}{n}\log\frac{2}{\delta}\right)\right\} = \frac{\delta}{2} \tag{D.96}$$

By Lemma D.7.1,

$$\mathbb{P}\left(\frac{1}{n}\sum_{j=1}^{n}X_j \geq 2\sigma\left(\frac{A_\gamma}{n}\log\frac{2}{\delta}\right)^{\frac{\gamma}{1+\gamma}} + 2M\frac{A_\gamma}{n}\log\frac{2}{\delta}\right) \leq \frac{\delta}{2} \tag{D.97}$$

Similarly,

$$\mathbb{P}\left(\frac{1}{n}\sum_{j=1}^{n}X_j \leq -2\sigma\left(\frac{A_\gamma}{n}\log\frac{2}{\delta}\right)^{\frac{\gamma}{1+\gamma}} + 2M\frac{A_\gamma}{n}\log\frac{2}{\delta}\right) \leq \frac{\delta}{2} \tag{D.98}$$

We finish the proof by using union bound. ■

Properties of Quantiles

Lemma D.7.2. *Let $\overline{X}$ be a zero-mean r.v. with $\mathbb{E}_{X\sim\mathcal{D}}\left[|X-\mu|^{1+\gamma}\right] \leq \sigma^{1+\gamma}$, for some $\gamma \in (0,1]$. Let $Q_p(\overline{X}) := \sup\left\{M \in \mathbb{R} : \mathbb{P}\left[\overline{X} \geq M\right] \geq 1-p\right\}$, for $0 < p \leq 1/2$. Then*

$$\max\left\{\left|Q_p(\overline{X})\right|, \left|Q_{1-p}(\overline{X})\right|\right\} \leq \sigma p^{-\frac{1}{1+\gamma}} \tag{D.99}$$

Proof. • If $0 < Q_p(\overline{X}) \leq Q_{1-p}(\overline{X})$, we have

$$\left|Q_p(\overline{X})\right| p \tag{D.100}$$

$$\leq \left|Q_{1-p}(\overline{X})\right| p = \mathbb{E}\left[Q_{1-p}(\overline{X})\mathbb{I}\left\{\overline{X} \geq Q_{1-p}(\overline{X})\right\}\right] \leq \mathbb{E}\left[\overline{X}\mathbb{I}\left\{\overline{X} \geq Q_{1-p}(\overline{X})\right\}\right] \tag{D.101}$$

$$= \mathbb{E}\left[\left|\overline{X}\right|\mathbb{I}\left\{\overline{X} \geq Q_{1-p}(\overline{X})\right\}\right] \leq \left(\mathbb{E}\left[\left|\overline{X}\right|^{1+\gamma}\right]\right)^{\frac{1}{1+\gamma}}\mathbb{E}\left[\mathbb{I}\left\{\overline{X} \geq Q_{1-p}(\overline{X})\right\}^{\frac{1+\gamma}{\gamma}}\right]^{\frac{\gamma}{1+\gamma}} \tag{D.102}$$

$$\text{(By Holder's inequality)} \tag{D.103}$$

$$\leq \sigma p^{\frac{\gamma}{1+\gamma}} \tag{D.104}$$

- If $Q_p(\overline{X}) \leq Q_{1-p}(\overline{X}) < 0$, we have

$$\left|Q_{1-p}(\overline{X})\right| p \tag{D.105}$$

$$\leq \left|Q_p(\overline{X})\right| p = \mathbb{E}\left[-Q_p(\overline{X})\mathbb{I}\left\{\overline{X} \leq Q_p(\overline{X})\right\}\right] \leq \mathbb{E}\left[-\overline{X}\mathbb{I}\left\{\overline{X} \leq Q_p(\overline{X})\right\}\right] \tag{D.106}$$

$$= \mathbb{E}\left[\left|\overline{X}\right| \mathbb{I}\left\{\overline{X} \leq Q_p(\overline{X})\right\}\right] \leq \left(\mathbb{E}\left[\left|\overline{X}\right|^{1+\gamma}\right]\right)^{\frac{1}{1+\gamma}} \mathbb{E}\left[\mathbb{I}\left\{\overline{X} \leq Q_p(\overline{X})\right\}^{\frac{1+\gamma}{\gamma}}\right]^{\frac{\gamma}{1+\gamma}} \tag{D.107}$$

$$\text{(By Holder's inequality)} \tag{D.108}$$

$$\leq \sigma p^{\frac{\gamma}{1+\gamma}} \tag{D.109}$$

- if $Q_p(\overline{X}) \leq 0 \leq Q_{1-p}(\overline{X})$, we can similarly show that $\left|Q_p(\overline{X})\right| p \leq \sigma p^{\frac{\gamma}{1+\gamma}}$ and $\left|Q_{1-p}(\overline{X})\right| p \leq \sigma p^{\frac{\gamma}{1+\gamma}}$.

■

D.8 Useful results

Lemma D.8.1 (Binomial concentration, Lemma A.1 of Xie et al. (2021)). *Suppose* $N \sim \mathrm{Bin}(n, p)$ *where* $n \geq 1$ *and* $p \in [0, 1]$. *Then with probability at least* $1 - \delta$, *we have*

$$\frac{p}{\max(N, 1)} \leq \frac{8 \log \frac{1}{\delta}}{n}. \tag{D.110}$$

E APPENDIX FOR CHAPTER 7

E.1 General Guarantee in the Value Space

By using standard analysis, we can demonstrate that learning on $A+\Gamma$ leads to a strategy pair $(\mathbf{p},\mathbf{q})$ with small *duality gap* defined as

A small duality gap implies that $(\mathbf{p},\mathbf{q})$ is an approximate Nash equilibrium of A:

Proposition E.1.1. Suppose $\|\Gamma\|_{\max} := \max_{i,j}\left|\mathbf{e}_i^\top\Gamma\mathbf{e}_j\right| \le \gamma$. If $(\mathbf{p},\mathbf{q})$ is NE of $A+\Gamma$, then the duality gap of $(\mathbf{p},\mathbf{q})$ is bounded by: $\mathrm{br}(\mathbf{q})^\top A\mathbf{q} - \mathbf{p}^\top A\mathrm{br}(\mathbf{p}) \le 2\gamma$ and $(\mathbf{p},\mathbf{q})$ is 2γ-approximate Nash equilibrium of A.

A similar result also appears as an intermediate step in Cui and Du (2022).

Proof of Proposition E.1.1. Suppose $(\mathbf{p}^*,\mathbf{q}^*)$ is NE of A. Because $(\mathbf{p},\mathbf{q})$ is NE of $A+\Gamma$, we have

$$\begin{aligned}\mathrm{br}(\mathbf{q})^\top A\mathbf{q} - \mathbf{p}^\top A\mathrm{br}(\mathbf{p}) =& \mathrm{br}(\mathbf{q})^\top(A+\Gamma)\mathbf{q} - \mathbf{p}^\top(A+\Gamma)\mathrm{br}(\mathbf{p}) - \mathrm{br}(\mathbf{q})^\top\Gamma\mathbf{q} + \mathbf{p}^\top\Gamma\mathrm{br}(\mathbf{p})\\ \le& \mathbf{p}^\top(A+\Gamma)\mathbf{q} - \mathbf{p}^\top(A+\Gamma)\mathbf{q} + 2\gamma\\ &(\text{ The last two term is because } \|\Gamma\|_{\max}\le\gamma.\ )\\ =& 2\gamma. \end{aligned} \tag{E.1}$$

We now show that $(\mathbf{p},\mathbf{q})$ is 2γ-approximate NE: for all $\mathbf{p}_0\in\Delta([m])$,

$$\mathbf{p}_0^\top A\mathbf{q} - \mathbf{p}^\top A\mathbf{q} \le \mathrm{br}(\mathbf{q})^\top A\mathbf{q} - \mathbf{p}^\top A\mathrm{br}(\mathbf{p}) \le 2\gamma,$$

where we use the definition of $\mathrm{br}(\cdot)$ in the first step and (E.1) in the second step. ∎

E.2 Proof of Lemma 7.4.1

To show $\Delta_{\mathcal{I}_A} = \min_{j\in\mathcal{J}_A} \min_{i\notin\mathcal{I}_A}\Big(v^* - \mathbf{e}_i^\top A\mathbf{e}_j\Big)$, we only need to show:

$$\min_{\mathbf{q}\in\mathcal{Q}} \min_{i\notin\mathcal{I}_A}\Big(-\mathbf{e}_i^\top A\mathbf{q}\Big) = \min_{j\in\mathcal{J}_A} \min_{i\notin\mathcal{I}_A}\Big(-\mathbf{e}_i^\top A\mathbf{e}_j\Big) \tag{E.2}$$

Because under Assumption 7.4.1 $\{\mathbf{e}_j : j \in \mathcal{J}_A\} \subseteq \mathcal{Q}$, we have:

$$\min_{\mathbf{q}\in\mathcal{Q}} \min_{i\notin\mathcal{I}_A}\Big(-\mathbf{e}_i^\top A\mathbf{q}\Big) \le \min_{j\in\mathcal{J}_A} \min_{i\notin\mathcal{I}_A}\Big(-\mathbf{e}_i^\top A\mathbf{e}_j\Big) \tag{E.3}$$

Let $\mathbf{q}_0 \in \mathcal{Q}$, $i_0 \notin \mathcal{I}_A$, s.t.

$$-\mathbf{e}_{i_0}^\top A\mathbf{q}_0 = \min_{\mathbf{q}\in\mathcal{Q}} \min_{i\notin\mathcal{I}_A}\Big(-\mathbf{e}_i^\top A\mathbf{q}\Big)$$

Let $j_0 \in \mathbf{supp}\,(\mathcal{Q})$, s.t.

$$-\mathbf{e}_{i_0}^\top A\mathbf{e}_{j_0} = \min_{j\in\mathbf{supp}\,(\mathcal{Q})}\Big(-\mathbf{e}_{i_0}^\top A\mathbf{e}_j\Big). \tag{E.4}$$

Then we have:

$$\begin{aligned}
\min_{\mathbf{q}\in\mathcal{Q}} \min_{i\notin\mathcal{I}_A}\Big(-\mathbf{e}_i^\top A\mathbf{q}\Big) &= -\,\mathbf{e}_{i_0}^\top A\mathbf{q}_0 = -\sum_{j\in\mathbf{supp}\,(\mathcal{Q})} q_{0,j}\mathbf{e}_{i_0}^\top A\mathbf{e}_j \\
&\ge -\,\mathbf{e}_{i_0}^\top A\mathbf{e}_{j_0} \quad (\text{ By (E.4) }) \\
&\ge \min_{j\in\mathcal{J}_A} \min_{i\notin\mathcal{I}_A}\Big(-\mathbf{e}_i^\top A\mathbf{e}_j\Big) \quad (\text{ because } j_0 \in \mathbf{supp}\,(\mathcal{Q}) \subseteq \mathcal{J}_A,\, i_0 \notin \mathcal{I}_A\ )
\end{aligned} \tag{E.5}$$

By (E.3) and (E.5), we prove (E.2) and Lemma 7.4.1. We can similarly show $\Delta_{\mathcal{J}_A} = \min_{i\in\mathcal{I}_A} \min_{j\notin\mathcal{J}_A}\Big(\mathbf{e}_i^\top A\mathbf{e}_j - v^*\Big)$.

E.3 Proof of Theorem 7.4.2

We now prove the "⇒" direction of Theorem 7.4.2.

Pure Base NE

We first show that if A is subset-NE-robust within radius γ, then A satisfies Assumption 7.4.1.

At a high level, we first show that if Assumption 7.4.1 does not hold, then game A has at least a mixed "base NE". A small perturbation will perturb the mixed NE and lead to a strategy that is no longer NE of A.

We first formally restate NE as the solution to linear programming and define the "base NEs".

Consider the linear programming:

$$\min_{(\mathbf{q},v)\in Q} v \qquad \max_{(\mathbf{p},v)\in P} v, \tag{E.6}$$

where

$$P := \Big\{(\mathbf{p}, v) : \mathbf{p} \in \Delta([m]), A^\top \mathbf{p} \geq v\Big\} \tag{E.7}$$

$$Q := \{(\mathbf{q}, v) : \mathbf{q} \in \Delta([n]), A\mathbf{q} \leq v\}. \tag{E.8}$$

Consider the set of vertex of the Q. By the theory of LP, the maximum value of LP is achieved on some vertices. This means there is a subset of vertices, s.t. $v = v^*$. The union of support of such "vertex" $\mathbf{q}$ is $\mathcal{J}_A$. Similar holds for the row player. Formally, Let:

$$\mathcal{P}^* := \{\mathbf{p} \in \mathcal{P} : (\mathbf{p}, v^*) \text{ is vertex of } P\} \tag{E.9}$$

$$\mathcal{Q}^* := \{\mathbf{q} \in \mathcal{Q} : (\mathbf{q}, v^*) \text{ is vertex of } Q\} \tag{E.10}$$

Then we have $\mathcal{P} = \text{conv}(\mathcal{P}^*)$. $\mathcal{Q} = \text{conv}(\mathcal{Q}^*)$, $\mathcal{I}_A = \bigcup_{\mathbf{p}\in\mathcal{P}^*} \textbf{supp}\,(\mathbf{p})$ and $\mathcal{J}_A = \bigcup_{\mathbf{q}\in\mathcal{Q}^*} \textbf{supp}\,(\mathbf{q})$. Recall that $\mathcal{P}\times\mathcal{Q}$ is the set of NEs of A.

We now show that if Assumption 7.4.1 does not hold, then there exists a mixed "base NE", i.e. there exists $\mathbf{p} \in \mathcal{P}^*$, $\|\mathbf{p}\|_0 > 1$ or $\mathbf{q} \in \mathcal{Q}^*$, $\|\mathbf{q}\|_0 > 1$. We prove by contradiction, suppose $\forall \mathbf{p} \in \mathcal{P}$ and $\forall \mathbf{q} \in \mathcal{Q}$, $\|\mathbf{p}\|_0 = \|\mathbf{q}\|_0 = 1$. Then for all $(i, j) \in \mathcal{I}_A\times\mathcal{J}_A$, $(\mathbf{e}_i, \mathbf{e}_j)$ is NE, meaning $A_{i,j} = v^*$, thus $A_{\mathcal{I}_A,\mathcal{J}_A}$ is constant. Now

we've proved the statement.

As the next step, we show that the mixed "base NE" can be perturbed by some perturbation matrix with magnitude upper bounded by γ_0 for any $\gamma_0 > 0$. We first show that there exists $\mathbf{p}^* \in \mathcal{P}^*$, s.t. $\|\mathbf{p}^*\| = \min_{\mathbf{p}\in\mathcal{P}} \|\mathbf{p}\|_0$. We prove this by contradiction. Suppose there exists $\mathbf{p} \notin \mathcal{P}^*$, s.t. $\|\mathbf{p}\|_0 < \min_{\mathbf{p}'\in\mathcal{P}^*} \|\mathbf{p}'\|_0$. By definition of vertex, there exists $k > 1$ and $\lambda_1, \ldots, \lambda_k \in (0, 1)$, $\mathbf{p}_1, \ldots, \mathbf{p}_k \in \mathcal{P}^*$ s.t. $\mathbf{p} = \sum_{i=1}^k \lambda_k \mathbf{p}_k$. Then we have $\|\mathbf{p}\|_0 \geq \|\mathbf{p}_1\|_0 \geq \min_{\mathbf{p}'\in\mathcal{P}^*} \|\mathbf{p}'\|_0$, which contradicts the fact that $\|\mathbf{p}\|_0 < \min_{\mathbf{p}'\in\mathcal{P}^*} \|\mathbf{p}'\|_0$. We can prove a similar result for $\mathcal{Q}^*$.

Let $(\mathbf{p}^*, \mathbf{q}^*) \in \mathcal{P}^* \times \mathcal{Q}^*$, s.t. $\|\mathbf{p}^*\|_0 = \min_{\mathbf{p}\in\mathcal{P}} \|\mathbf{p}\|_0$ and $\|\mathbf{q}^*\|_0 = \min_{\mathbf{q}\in\mathcal{Q}} \|\mathbf{q}\|_0$.

If $\|\mathbf{p}^*\|_0 = \|\mathbf{q}^*\|_0 = 1$, by the results above, there exists $\mathbf{p} \in \mathcal{P}^*$, s.t. $\|\mathbf{p}\|_0 > 1$ or there exists $\mathbf{q} \in \mathcal{Q}^*$, s.t. $\|\mathbf{q}\|_0 > 1$. W.l.o.g., assume there exists $\mathbf{q} \in \mathcal{Q}^*$, s.t. $\|\mathbf{q}\|_0 > 1$, $\mathcal{I}_0 := \mathbf{supp}(\mathbf{p}^*) = \{1\}$, $\mathcal{J}_0 := \mathbf{supp}(\mathbf{q}) = [l]$. Then we have $A_{\mathcal{I}_0,\mathcal{J}_0} = v^*\mathbf{1}^\top$. Because $(\mathbf{q}, v^*)$ is a vertex of (E.6), we can find $\mathcal{I}_1$, s.t.:

1. $\mathcal{I}_0 \subsetneq \mathcal{I}_1$;
2. $|\mathcal{I}_1| = |\mathcal{J}_0|$
3. $\begin{pmatrix} A_{\mathcal{I}_1,\mathcal{J}_0} & -1 \\ 1 & 0 \end{pmatrix}$ is invertible;
4. $\begin{pmatrix} A_{\mathcal{I}_1,\mathcal{J}_0} & -1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} \mathbf{q}_{\mathcal{J}_0} \\ v^* \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$.

Consider the following two perturbation matrices:

$$\Gamma^1 = \begin{pmatrix} \Gamma^1_{\mathcal{I}_1,\mathcal{J}_0} & \gamma_0\mathbf{1}\mathbf{1}^\top \\ -\gamma_0\mathbf{1}\mathbf{1}^\top & \mathbf{0} \end{pmatrix}, \ \Gamma^2 = \begin{pmatrix} \Gamma^2_{\mathcal{I}_1,\mathcal{J}_0} & \gamma_0\mathbf{1}\mathbf{1}^\top \\ -\gamma_0\mathbf{1}\mathbf{1}^\top & \mathbf{0} \end{pmatrix}, \tag{E.11}$$

where

$$\Gamma^1_{\mathcal{I}_1,\mathcal{J}_0} = \begin{pmatrix} -\frac{(A_{\mathcal{I}_1,\mathcal{J}_0} - v^*)\mathbf{e}_1\gamma}{\mathbf{e}_1^\top \mathbf{q}_{\mathcal{J}_0} + \gamma} & \frac{(A_{\mathcal{I}_1,\mathcal{J}_0} - v^*)\mathbf{e}_2\gamma}{\mathbf{e}_2^\top \mathbf{q}_{\mathcal{J}_0} - \gamma} & \mathbf{0}\cdots\mathbf{0} \end{pmatrix} \tag{E.12}$$

$$\Gamma^2_{\mathcal{I}_1,\mathcal{J}_0} = \begin{pmatrix} \frac{(A_{\mathcal{I}_1,\mathcal{J}_0} - v^*)\mathbf{e}_1\gamma}{\mathbf{e}_1^\top \mathbf{q}_{\mathcal{J}_0} - \gamma} & -\frac{(A_{\mathcal{I}_1,\mathcal{J}_0} - v^*)\mathbf{e}_2\gamma}{\mathbf{e}_2^\top \mathbf{q}_{\mathcal{J}_0} + \gamma} & \mathbf{0}\cdots\mathbf{0} \end{pmatrix} \tag{E.13}$$

where $\gamma > 0$ is small enough, s.t.:

1. $\gamma \leq \gamma_0$;
2. $\begin{pmatrix} A_{\mathcal{I}_1,\mathcal{J}_0} + \Gamma^1_{\mathcal{I}_1,\mathcal{J}_0} & -1 \\ 1 & 0 \end{pmatrix}$ and $\begin{pmatrix} A_{\mathcal{I}_1,\mathcal{J}_0} + \Gamma^2_{\mathcal{I}_1,\mathcal{J}_0} & -1 \\ 1 & 0 \end{pmatrix}$ are invertible;
3. $\left\|\Gamma^1_{\mathcal{I}_1,\mathcal{J}_0}\right\|_{\max} \leq \gamma_0$, $\left\|\Gamma^2_{\mathcal{I}_1,\mathcal{J}_0}\right\|_{\max} \leq \gamma_0$;
4. $\gamma \left\|A_{\mathcal{I}_1^c,1} - A_{\mathcal{I}_1^c,2}\right\|_\infty \leq \gamma_0$.

We now show that $(\mathbf{p}^*, \mathbf{q}_0^1)$ is an NE in $A+\Gamma^1$ and $(\mathbf{p}^*, \mathbf{q}_0^2)$ is an NE in $A+\Gamma^2$, where

$$\mathbf{e}_j^\top \mathbf{q}_0^1 = \begin{cases} \mathbf{e}_j^\top \mathbf{q}_0 + \gamma & \text{if } j = 1 \\ \mathbf{e}_j^\top \mathbf{q}_0 - \gamma & \text{if } j = 2 \\ \mathbf{e}_j^\top \mathbf{q}_0 & \text{o.w.} \end{cases}, \ \mathbf{e}_j^\top \mathbf{q}_0^2 = \begin{cases} \mathbf{e}_j^\top \mathbf{q}_0 - \gamma & \text{if } j = 1 \\ \mathbf{e}_j^\top \mathbf{q}_0 + \gamma & \text{if } j = 2 \\ \mathbf{e}_j^\top \mathbf{q}_0 & \text{o.w.} \end{cases}. \tag{E.14}$$

Then

$$\begin{aligned} (A+\Gamma^1)\mathbf{q}_0^1 =& A\mathbf{q}_0^1 + \Gamma^1 \mathbf{q}_0^1 = A\mathbf{q}_0 + \gamma(A_{\cdot,1} - A_{\cdot,2}) && \text{(E.15)} \\ &+ \begin{pmatrix} -(A_{\mathcal{I}_1,\mathcal{J}_0} - v^*)\mathbf{e}_1\gamma + (A_{\mathcal{I}_1,\mathcal{J}_0} - v^*)\mathbf{e}_2\gamma \\ -\gamma_0 \mathbf{1} \end{pmatrix} && \text{(E.16)} \\ =& A\mathbf{q}_0 + \begin{pmatrix} \mathbf{0} \\ \gamma(A_{\mathcal{I}_1^c,1} - A_{\mathcal{I}_1^c,2}) - \gamma_0 \mathbf{1} \end{pmatrix} \leq v^* \mathbf{1} && \text{(E.17)} \end{aligned}$$

$$\mathbf{p}^{*\top}(A+\Gamma^1) = A_{1,\cdot} \geq v^* \mathbf{1} \tag{E.18}$$

$$\mathbf{p}^{*\top}(A+\Gamma^1)\mathbf{q}_0^1 = v^* \tag{E.19}$$

thus $(\mathbf{p}^*, \mathbf{q}_0^1)$ is an NE in $A+\Gamma^1$, similarly, we can show that $(\mathbf{p}^*, \mathbf{q}_0^2)$ is an NE in $A+\Gamma^2$. Because $\frac{1}{2}(\mathbf{q}_0^1 + \mathbf{q}_0^2) = \mathbf{q}$ and the fact that $(\mathbf{q}, v^*)$ is a vertex of (E.6), this

means at least one of $(\mathbf{q}_0^1, v^*)$ and $(\mathbf{q}_0^2, v^*)$ is not feasible in (E.6). Thus at least one of Γ^1 and Γ^2 creates a new NE.

If $\|\mathbf{p}^*\|_0 > \|\mathbf{q}^*\|_0 = 1$ or $1 = \|\mathbf{p}^*\|_0 < \|\mathbf{q}^*\|_0$ we can add perturbation similarly.

In the following, we deal with the case that $\|\mathbf{p}^*\|_0 > 1$ and $\|\mathbf{q}^*\|_0 > 1$. W.l.o.g., assume $\|\mathbf{p}^*\|_0 \leq \|\mathbf{q}^*\|_0$. We will add perturbation step-by-step until the perturbation creates a new NE. Let $\mathcal{I}_0 := \mathbf{supp}\,(\mathbf{p}^*)$, $\mathcal{J}_0 := \mathbf{supp}\,(\mathbf{q}^*)$.

Consider the first step perturbation:

$$A + \Gamma_1 = \begin{pmatrix} A_{\cdot,\mathcal{J}_0} & A_{\cdot,\mathcal{J}_0^c} + \gamma_0 \mathbf{1}\mathbf{1}^\top \end{pmatrix}. \tag{E.20}$$

Note that

$$\mathbf{p}^{*\top} \begin{pmatrix} A_{\cdot,\mathcal{J}_0} & A_{\cdot,\mathcal{J}_0^c} + \gamma_0 \mathbf{1}^\top \mathbf{1} \end{pmatrix} \geq \begin{pmatrix} v^* \mathbf{1} & (v^* + \gamma_0)\mathbf{1} \end{pmatrix} \geq v^* \mathbf{1}, \tag{E.21}$$

so the column player should only choose actions in $\mathcal{J}_0$ and $(\mathbf{p}^*, \mathbf{q}^*)$ is still NE of the perturbed game.

Consider the LP for the row player in game $A + \Gamma_1$:

$$\max_{(\mathbf{p},v):\mathbf{p}\in\Delta([m]),(A+\Gamma_1)^\top \mathbf{p}\geq v} v. \tag{E.22}$$

Because $(\mathbf{p}^*, \mathbf{q}^*)$ is an NE of $A + \Gamma_1$, $(\mathbf{p}^*, v^*)$ is an optimal solution of (E.22). If $(\mathbf{p}^*, v^*)$ is not a vertex, then there exists optimal solution $(\mathbf{p}_1, v^*)$ and $(\mathbf{p}_2, v^*)$ s.t. $\mathbf{p}^* = \frac{1}{2}\mathbf{p}_1 + \frac{1}{2}\mathbf{p}_2$. Because $(\mathbf{p}^*, v^*)$ is an optimal vertex of (E.6), at least one of $(\mathbf{p}_1, v^*)$ and $(\mathbf{p}_2, v^*)$ is not optimal solution of (E.6), which means at least one of $(\mathbf{p}_1, \mathbf{q}^*)$ and $\mathbf{p}_2, \mathbf{q}^*$ is NE of $A + \Gamma_1$ but is not NE of A.

By (E.21), the set of active constraints at $(\mathbf{p}^*, v^*)$ are:

$$\mathbf{p}^{*\top} A_{\cdot,\mathcal{J}_0} = v^* \mathbf{1},\ \mathbf{1}^\top \mathbf{p}^* = 1,\ \mathbf{e}_i^\top \mathbf{p}^* = 0, \forall i \in \mathcal{I}_0^c. \tag{E.23}$$

If $(\mathbf{p}^*, v^*)$ is a vertex, $\begin{pmatrix} A_{\mathcal{I}_0,\mathcal{J}_0} & \mathbf{1} \\ -\mathbf{1} & 0 \end{pmatrix}$ has linearly independent rows.

Then there exists $\mathcal{I}_1$, s.t.:

1. $\mathcal{I}_0 \subseteq \mathcal{I}_1$;
2. $\begin{pmatrix} A_{\mathcal{I}_1,\mathcal{J}_0} & -\mathbf{1} \\ \mathbf{1} & 0 \end{pmatrix}$ is invertible;
3. $A_{\mathcal{I}_1,\mathcal{J}_0}\mathbf{q}^* = v^*\mathbf{1}$.

Consider the second step perturbation:

$$A + \Gamma_1 + \Gamma_2 = \begin{pmatrix} A_{\mathcal{I}_1,\mathcal{J}_0} & A_{\mathcal{I}_1,\mathcal{J}_0^c} + \gamma_0 \mathbf{1}\mathbf{1}^\top \\ A_{\mathcal{I}_1^c,\mathcal{J}_0} - \gamma_0 \mathbf{1}\mathbf{1}^\top & A_{\mathcal{I}_1^c,\mathcal{J}_0^c} + \gamma_0 \mathbf{1}\mathbf{1}^\top \end{pmatrix}. \tag{E.24}$$

If $\mathcal{I}_1 = \mathcal{I}_0$, consider the third step pertrubation:

$$\Gamma_3^1 = \begin{pmatrix} \Gamma^1_{\mathcal{I}_1,\mathcal{J}_0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix}, \ \Gamma_3^2 = \begin{pmatrix} \Gamma^2_{\mathcal{I}_1,\mathcal{J}_0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix}, \tag{E.25}$$

where $\Gamma^1_{\mathcal{I}_1,\mathcal{J}_0}$ and $\Gamma^2_{\mathcal{I}_1,\mathcal{J}_0}$ has the same format as in (E.12) and (E.13), but γ subjects to the following additional constraints: γ is small enough, s.t.

1.
$$\mathbf{p}^1_{\mathcal{I}_1} := \begin{pmatrix} (A + \Gamma_1 + \Gamma_2 + \Gamma_3^1)^\top_{\mathcal{I}_1,\mathcal{J}_0} & -\mathbf{1} \\ \mathbf{1} & 0 \end{pmatrix}^{-1} \begin{pmatrix} \mathbf{0} \\ 1 \end{pmatrix} > 0,$$
$$\mathbf{p}^2_{\mathcal{I}_1} := \begin{pmatrix} (A + \Gamma_1 + \Gamma_2 + \Gamma_3^2)^\top_{\mathcal{I}_1,\mathcal{J}_0} & -\mathbf{1} \\ \mathbf{1} & 0 \end{pmatrix}^{-1} \begin{pmatrix} \mathbf{0} \\ 1 \end{pmatrix} > 0$$
2. $\mathbf{p}^{1\top}_{\mathcal{I}_1}(A + \Gamma_1 + \Gamma_2 + \Gamma_3^1)_{\mathcal{I}_1,\mathcal{J}_0^c} > v^*$, $\mathbf{p}^{2\top}_{\mathcal{I}_1}(A + \Gamma_1 + \Gamma_2 + \Gamma_3^1)_{\mathcal{I}_1,\mathcal{J}_0^c} > v^*$

We now show that $(\mathbf{p}^1, \mathbf{q}_0^1)$ and $(\mathbf{p}^2, \mathbf{q}_0^2)$ are NE of $A+\Gamma_1+\Gamma_2+\Gamma_3^1$ and $A+\Gamma_1+\Gamma_2+\Gamma_3^2$, respectively, where $\mathbf{q}_0^1, \mathbf{q}_0^2$ are as defined in (E.14) and

$$\mathbf{e}_i^\top \mathbf{p}^1 = \begin{cases} \mathbf{e}_i^\top \mathbf{p}^1_{\mathcal{I}_1} & \text{if } i \in \mathcal{I}_1 \\ 0 & \text{o.w.} \end{cases}, \ \mathbf{e}_i^\top \mathbf{p}^2 = \begin{cases} \mathbf{e}_i^\top \mathbf{p}^2_{\mathcal{I}_1} & \text{if } i \in \mathcal{I}_1 \\ 0 & \text{o.w.} \end{cases}. \tag{E.26}$$

By definition and the fact that γ is small, there exists $\tilde{v}$,

$$\mathbf{p}_{\mathcal{I}_1}^{1\top}(A+\Gamma_1+\Gamma_2+\Gamma_3^1)_{\mathcal{I}_1,\mathcal{J}_0} = \mathbf{1}^\top \tilde{v} \tag{E.27}$$

$$\mathbf{p}_{\mathcal{I}_1}^{1\top}(A+\Gamma_1+\Gamma_2+\Gamma_3^1)_{\mathcal{I}_1,\mathcal{J}_0^c} > \mathbf{1}^\top v^* \tag{E.28}$$

$$(A+\Gamma_1+\Gamma_2+\Gamma_3^1)_{\mathcal{I}_1,\mathcal{J}_0}\mathbf{q}_{\mathcal{J}_0}^1 = \mathbf{1}v^* \tag{E.29}$$

$$(A+\Gamma_1+\Gamma_2+\Gamma_3^1)_{\mathcal{I}_1^c,\mathcal{J}_0}\mathbf{q}_{\mathcal{J}_0}^1 < \mathbf{1}v^* \tag{E.30}$$

This means $\tilde{v} = v^*$. Then $(\mathbf{p}^1, \mathbf{q}_0^1)$ is NE of $A+\Gamma_1+\Gamma_2+\Gamma_3^1$. We can show the result for $A+\Gamma_1+\Gamma_2+\Gamma_3^2$ similarly.

Similarly, because $\frac{1}{2}(\mathbf{q}_0^1+\mathbf{q}_0^2) = \mathbf{q}$ and the fact that $(\mathbf{q}, v^*)$ is a vertex of (E.6), this means at least one of $(\mathbf{q}_0^1, v^*)$ and $(\mathbf{q}_0^2, v^*)$ is not feasible in (E.6). Thus at least one of Γ^1 and Γ^2 creates a new NE.

If $\mathcal{I}_0 \subsetneq \mathcal{I}_1$: We first show that $\mathbf{q}^*$ is not the unique optimal strategy for the column player in game $A+\Gamma_1+\Gamma_2$. We prove by contradiction, suppose $\mathbf{q}^*$ is the unique optimal strategy for the column player in game $A+\Gamma_1+\Gamma_2$. By Corollary 3A of Goldman and Tucker (2016), there exists $\mathbf{p}'$, s.t. $(\mathbf{p}', \mathbf{q}^*)$ is an NE and $\mathbf{supp}(\mathbf{p}') = \mathcal{I}_1$. By Theorem E.7.1, $(\mathbf{p}', \mathbf{q}^*)$ is the unique NE. This contradicts the fact that $(\mathbf{p}^*, \mathbf{q}^*)$ is also an NE. Consider the following LP:

$$\min_{(\mathbf{q},v):\mathbf{q}\in\Delta([n]),(A+\Gamma_1+\Gamma_2)\mathbf{q}\le v} v. \tag{E.31}$$

Because $\mathbf{q}^*$ is not the unique optimal strategy for the column player in game $A+\Gamma_1+\Gamma_2$, there exists $\mathbf{q}' \neq \mathbf{q}^*$, s.t. $\mathbf{q}'$ is an optimal vertex of (E.31). By (E.21), $\mathbf{supp}(\mathbf{q}') \subseteq \mathcal{J}_0$. If $\mathbf{supp}(\mathbf{q}') \subsetneq \mathcal{J}_0$, by definition of $\mathcal{J}_0$, $(\mathbf{p}^*, \mathbf{q}')$ is NE of $A+\Gamma_1+\Gamma_2$ but is not NE of A. If $\mathbf{supp}(\mathbf{q}') = \mathcal{J}_0$, because $\mathbf{q}' \neq \mathbf{q}^*$ and both $\mathbf{q}'$ and $\mathbf{q}^*$ are vertices, $\mathbf{q}'$ and $\mathbf{q}^*$ must have different sets of active constraints. Suppose the active constraints at $\mathbf{q}'$ are:

$$(A+\Gamma_1+\Gamma_2)_{\mathcal{I}_2,\cdot}\mathbf{q}' = v^*,\ \mathbf{1}^\top\mathbf{q}' = 1,\ \mathbf{e}_j^\top\mathbf{q}' = 0, \forall j \in \mathcal{J}_0^c, \tag{E.32}$$

where $\mathcal{I}_2 \neq \mathcal{I}_1$ and $\mathcal{I}_0 \subsetneq \mathcal{I}_2$. Let $i_0 \in \mathcal{I}_2 \cap \mathcal{I}_0^c$, then we have:

$$(A + \Gamma_1 + \Gamma_2)_{i_0,\cdot}\mathbf{q}' = A_{i_0,\cdot}\mathbf{q}' - \gamma_0 = v^*, \tag{E.33}$$

thus $A_{i_0,\cdot}\mathbf{q}' = v^* + \gamma_0$. Thus $(\mathbf{q}', v^*)$ is not feasible in (E.6). Thus $(\mathbf{p}^*, \mathbf{q}')$ is an NE of $A + \Gamma_1 + \Gamma_2$ but is not NE of A.

Perturbation Magnitude

We now show that if A is subset-NE-robust within radius γ, then $\gamma < \frac{1}{2}\min\{\Delta_{\mathcal{I}_A}, \Delta_{\mathcal{J}_A}\}$. We prove this by contradiction. We show that for any game A, we can design a perturbation matrix Γ, s.t. $\|\Gamma\|_{\mathcal{I}_A \cup \mathcal{J}_A} = \frac{1}{2}\min\{\Delta_{\mathcal{I}_A}, \Delta_{\mathcal{J}_A}\}$ and $A + \Gamma$ has a NE outside $\mathcal{I}_A \times \mathcal{J}_A$. In Section E.3, we've already shown that A has pure base NE.

Without loss of generality, assume:

- $\Delta_{\mathcal{J}_A} \leq \Delta_{\mathcal{I}_A}$
- $\mathcal{I}_A = [k]$, $\mathcal{J}_A = [l]$
- $\Delta_{\mathcal{J}_A} = \mathbf{e}_k^\top A \mathbf{e}_{l+1} - v^*$
- $A_{k-1,l+1} \geq A_{i,l+1}$ for all $i \in [k]$

Consider the following perturbation on A:

$$A + \Gamma := \begin{pmatrix} A_{[k-2]\times[l]} & A_{[k-2],l+1} & A_{[k-2],[l+2:n]} \\ A_{k-1\times[l]} & A_{k-1,l+1} + \frac{1}{2}\Delta_{\mathcal{J}_A} & A_{k-1,[l+2:n]} \\ A_{k\times[l]} + \frac{1}{2}\Delta_{\mathcal{J}_A} & A_{k,l+1} - \frac{1}{2}\Delta_{\mathcal{J}_A} & A_{k,[l+2:n]} \\ A_{[k+1:m]\times[l]} & -\frac{1-\alpha}{\alpha}A_{[k+1:m],l} + \frac{1}{\alpha}v^* & A_{[k+1:m],[l+2:n]} \end{pmatrix},$$

where $\alpha = \frac{\frac{1}{2}\Delta_{\mathcal{J}_A}}{A_{k-1,l+1} - v^* + \frac{1}{2}\Delta_{\mathcal{J}_A}}$. Because $\|\cdot\|_{\mathcal{I}_A \cup \mathcal{J}_A}$ does not restrict the perturbation magnitude of the submatrix on ${\mathcal{I}_A}^c \times {\mathcal{J}_A}^c$, the perturbation above is valid. One can verify that $((1-\alpha)\mathbf{e}_l + \alpha\mathbf{e}_{l+1}, \mathbf{e}_k)$ is a NE. This NE has support on $\mathbf{e}_{l+1}$, which is outside the NE support. This This means $\mathrm{NE}(A + \Gamma) \not\subseteq \mathrm{NE}(A)$.

E.4 Proof of Theorem 7.4.3

If A and γ satisfies: $|\mathcal{I}_A| = |\mathcal{J}_A| = 1$; and $\gamma < \frac{1}{2}\min\{\Delta_{\mathcal{I}_A}, \Delta_{\mathcal{J}_A}\}$, then A and Γ also satisfies the conditions in Theorem 7.4.2. By Theorem 7.4.2, we get $\forall \Gamma : \|\Gamma\|_{\mathcal{I}_A \cup \mathcal{J}_A} \leq \gamma$, $\mathrm{NE}(A+\Gamma) \subseteq \mathrm{NE}(A)$. Because $\gamma < \frac{1}{2}\min\{\Delta_{\mathcal{I}_A}, \Delta_{\mathcal{J}_A}\}$ also implies A has a unique NE, we have $|\mathrm{NE}(A+\Gamma)| = |\mathrm{NE}(A)| = 1$ and thus $\mathrm{NE}(A+\Gamma) = \mathrm{NE}(A)$. This means A is exact-NE-robust within radius γ.

If A is exact-NE-robust within radius γ, then A is subset-NE-robust within radius γ. By Theorem 7.4.2, A satisfies Assumption 7.4.1 and $\gamma < \frac{1}{2}\min\{\Delta_{\mathcal{I}_A}, \Delta_{\mathcal{J}_A}\}$. We now show that $|\mathcal{I}_A| = |\mathcal{J}_A| = 1$. We prove this by contradiction. Without loss of generality, assume $\mathcal{I}_A = [k]$, $\mathcal{J}_A = [l]$ and $l > 1$. Let $\gamma_0 := \frac{1}{4}\min\{\Delta_{\mathcal{I}_A}, \Delta_{\mathcal{J}_A}\}$. Consider the following perturbation on A:

$$A + \Gamma := \begin{pmatrix} A_{\cdot,1} & A_{\cdot,2} + \gamma_0 & A_{\cdot,3} & A_{\cdot,n} \end{pmatrix}.$$

Then $(\mathbf{e}_1, \mathbf{e}_2)$ is NE of A but is not NE of $A+\Gamma$, which contradicts with $\mathrm{NE}(A+\Gamma) = \mathrm{NE}(A)$.

E.5 Proof of Theorem 7.4.4

We first formally restate Theorem 7.4.4:

Theorem E.5.1. *Let $A \in \mathbb{R}^{m\times n}$ be a game matrix, with a unique NE $(\mathbf{p}^*, \mathbf{q}^*)$. Let $\mathcal{I}_A = \mathbf{supp}(\mathbf{p}^*)$, $\mathcal{J}_A = \mathbf{supp}(\mathbf{q}^*)$. Then we have $|\mathcal{I}_A| = |\mathcal{J}_A|$ and there exists $\Delta_{\mathcal{I}_A}, \Delta_{\mathcal{J}_A} > 0$, s.t.:*

$$\mathbf{e}_i^\top A \mathbf{q}^* \leq \mathbf{p}^{*\top} A \mathbf{q}^* - \Delta_{\mathcal{I}_A}, \forall i \notin \mathcal{I}_A \tag{E.34}$$

$$\mathbf{p}^{*\top} A \mathbf{e}_j \geq \mathbf{p}^{*\top} A \mathbf{q}^* + \Delta_{\mathcal{J}_A}, \forall j \notin \mathcal{J}_A \tag{E.35}$$

Let $k := |\mathcal{I}_A| = |\mathcal{J}_A|$, $\gamma := \|\Gamma\|_{\mathcal{I}_A \cup \mathcal{J}_A}$, $\widetilde{A_{\mathcal{I}_A,\mathcal{J}_A}} := A^\top_{\mathcal{I}_A,\mathcal{J}_A} A_{\mathcal{I}_A,\mathcal{J}_A} + \mathbf{1}\mathbf{1}^\top - \frac{1}{k} A^\top_{\mathcal{I}_A,\mathcal{J}_A} \mathbf{1}\mathbf{1}^\top A_{\mathcal{I}_A,\mathcal{J}_A}$ and $\widetilde{A^\top_{\mathcal{I}_A,\mathcal{J}_A}} := A_{\mathcal{I}_A,\mathcal{J}_A} A^\top_{\mathcal{I}_A,\mathcal{J}_A} + \mathbf{1}\mathbf{1}^\top - \frac{1}{k} A_{\mathcal{I}_A,\mathcal{J}_A} \mathbf{1}\mathbf{1}^\top A^\top_{\mathcal{I}_A,\mathcal{J}_A}$. If:

- $\gamma < \frac{1}{k}\sigma_{\min}$, *where* $\sigma_{\min}$ *is the eigenvalue of* $\begin{pmatrix} A_{\mathcal{I}_A,\mathcal{J}_A} & -\mathbf{1} \\ \mathbf{1} & 0 \end{pmatrix}$ *with the smallest absolute value;*
- $4\gamma \left\|A_{\mathcal{I}_A,\mathcal{J}_A}\right\|_1 + 2k\gamma^2 < \frac{1}{2k^{3/2}\left\|\widetilde{A_{\mathcal{I}_A,\mathcal{J}_A}}^{-1}\right\|_2} \frac{\min(\widetilde{A_{\mathcal{I}_A,\mathcal{J}_A}}^{-1}\mathbf{1})}{\left\|\widetilde{A_{\mathcal{I}_A,\mathcal{J}_A}}^{-1}\mathbf{1}\right\|_\infty}$
- $4\gamma \left\|A^\top_{\mathcal{I}_A,\mathcal{J}_A}\right\|_1 + 2k\gamma^2 < \frac{1}{2k^{3/2}\left\|\widetilde{A^\top_{\mathcal{I}_A,\mathcal{J}_A}}^{-1}\right\|_2} \frac{\min(\widetilde{A^\top_{\mathcal{I}_A,\mathcal{J}_A}}^{-1}\mathbf{1})}{\left\|\widetilde{A^\top_{\mathcal{I}_A,\mathcal{J}_A}}^{-1}\mathbf{1}\right\|_\infty}$
- $$\Delta_{\mathcal{I}_A} - \left(\left\|A_{\mathcal{I}_A,\mathcal{J}_A}\right\|_\infty + \left\|A_{\mathcal{I}_A{}^c,\mathcal{J}_A}\right\|_\infty\right)2k^{3/2}\left(4\gamma\left\|A_{\mathcal{I}_A,\mathcal{J}_A}\right\|_1 + 2k\gamma^2\right) \cdot \left\|\widetilde{A_{\mathcal{I}_A,\mathcal{J}_A}}^{-1}\right\|_2 \left\|\widetilde{A_{\mathcal{I}_A,\mathcal{J}_A}}^{-1}\mathbf{1}\right\|_\infty - 2\gamma > 0$$
- $$\Delta_{\mathcal{J}_A} - \left(\left\|A^\top_{\mathcal{I}_A,\mathcal{J}_A}\right\|_\infty + \left\|A^\top_{\mathcal{I}_A,\mathcal{J}_A{}^c}\right\|_\infty\right)2k^{3/2}\left(4\gamma\left\|A^\top_{\mathcal{I}_A,\mathcal{J}_A}\right\|_1 + 2k\gamma^2\right) \cdot \left\|\widetilde{A^\top_{\mathcal{I}_A,\mathcal{J}_A}}^{-1}\right\|_2 \left\|\widetilde{A^\top_{\mathcal{I}_A,\mathcal{J}_A}}^{-1}\mathbf{1}\right\|_\infty - 2\gamma > 0$$

then the perturbed game $A + \Gamma$ *has a unique NE* $(\tilde{\mathbf{p}}, \tilde{\mathbf{q}})$*, s.t.:*

- $\mathbf{supp}\,(\tilde{\mathbf{p}}) = \mathcal{I}_A$, $\mathbf{supp}\,(\tilde{\mathbf{q}}) = \mathcal{J}_A$;
- $d_{\mathrm{TV}}(\mathbf{p}^*, \tilde{\mathbf{p}}) \le k^{3/2}\left(4\gamma\left\|A^\top_{\mathcal{I}_A,\mathcal{J}_A}\right\|_1 + 2k\gamma^2\right)\left\|\widetilde{A^\top_{\mathcal{I}_A,\mathcal{J}_A}}^{-1}\right\|_2 \left\|\widetilde{A^\top_{\mathcal{I}_A,\mathcal{J}_A}}^{-1}\mathbf{1}\right\|_1$
- $d_{\mathrm{TV}}(\mathbf{q}^*, \tilde{\mathbf{q}}) \le k^{3/2}\left(4\gamma\left\|A_{\mathcal{I}_A,\mathcal{J}_A}\right\|_1 + 2k\gamma^2\right)\left\|\widetilde{A_{\mathcal{I}_A,\mathcal{J}_A}}^{-1}\right\|_2 \left\|\widetilde{A_{\mathcal{I}_A,\mathcal{J}_A}}^{-1}\mathbf{1}\right\|_1$

Proof. We first present a proof sketch:

1. Because $(\mathbf{p}^*, \mathbf{q}^*)$ is the unique NE of A. According to the proof of Theorem E.7.1, we can write $(\mathbf{q}^*_{\mathcal{J}_A}, v^*)$ as the unique solution to some linear system;

2. With small perturbation, the perturbed linear system is also invertible and thus has a unique solution $(\tilde{\mathbf{q}}, \tilde{v})$, furthermore, the solution is positive: $\tilde{\mathbf{q}} > 0$ and is close to the original solution in TV distance;

3. When the perturbation is small, the perturbed game also has a positive switch-out gap.

4. By Theorem E.7.1, $\tilde{\mathbf{q}}$ and the corresponding $\tilde{\mathbf{p}}$ form the unique NE of the perturbed game, with the following properties:

   - $(\tilde{\mathbf{p}}, \tilde{\mathbf{q}})$ and $(\mathbf{p}^*, \mathbf{q}^*)$ have the same support;
   - $(\tilde{\mathbf{p}}, \tilde{\mathbf{q}})$ and $(\mathbf{p}^*, \mathbf{q}^*)$ are close in TV distance.

We introduce the following notations:

- $\overline{A_{\mathcal{I}_A,\mathcal{J}_A}} := \begin{pmatrix} A_{\mathcal{I}_A,\mathcal{J}_A} & -\mathbf{1} \\ \mathbf{1} & 0 \end{pmatrix}$
- $\overline{A^\top_{\mathcal{I}_A,\mathcal{J}_A}} := \begin{pmatrix} A^\top_{\mathcal{I}_A,\mathcal{J}_A} & -\mathbf{1} \\ \mathbf{1} & 0 \end{pmatrix}$
- $\overline{(A+\Gamma)_{\mathcal{I}_A,\mathcal{J}_A}} := \begin{pmatrix} A_{\mathcal{I}_A,\mathcal{J}_A} + \Gamma_{\mathcal{I}_A,\mathcal{J}_A} & -\mathbf{1} \\ \mathbf{1} & 0 \end{pmatrix}$
- $\overline{(A+\Gamma)^\top_{\mathcal{I}_A,\mathcal{J}_A}} := \begin{pmatrix} A^\top_{\mathcal{I}_A,\mathcal{J}_A} + \Gamma^\top_{\mathcal{I}_A,\mathcal{J}_A} & -\mathbf{1} \\ \mathbf{1} & 0 \end{pmatrix}$
- $\widetilde{(A+\Gamma)}_{\mathcal{I}_A,\mathcal{J}_A} := (A_{\mathcal{I}_A,\mathcal{J}_A} + \Gamma_{\mathcal{I}_A,\mathcal{J}_A})^\top (A_{\mathcal{I}_A,\mathcal{J}_A} + \Gamma_{\mathcal{I}_A,\mathcal{J}_A}) + \mathbf{1}\mathbf{1}^\top - \frac{1}{k}(A_{\mathcal{I}_A,\mathcal{J}_A} + \Gamma_{\mathcal{I}_A,\mathcal{J}_A})^\top \mathbf{1}\mathbf{1}^\top (A_{\mathcal{I}_A,\mathcal{J}_A} + \Gamma_{\mathcal{I}_A,\mathcal{J}_A})$;
- $\widetilde{(A+\Gamma)^\top}_{\mathcal{I}_A,\mathcal{J}_A} := (A_{\mathcal{I}_A,\mathcal{J}_A} + \Gamma_{\mathcal{I}_A,\mathcal{J}_A})(A_{\mathcal{I}_A,\mathcal{J}_A} + \Gamma_{\mathcal{I}_A,\mathcal{J}_A})^\top + \mathbf{1}\mathbf{1}^\top - \frac{1}{k}(A_{\mathcal{I}_A,\mathcal{J}_A} + \Gamma_{\mathcal{I}_A,\mathcal{J}_A})\mathbf{1}\mathbf{1}^\top (A_{\mathcal{I}_A,\mathcal{J}_A} + \Gamma_{\mathcal{I}_A,\mathcal{J}_A})^\top$

Invertiblility after perturbation: By Theorem E.7.1 and the fact that $\mathbf{q}^*$ is the unique NE strategy of the column player, $\overline{A_{\mathcal{I}_A,\mathcal{J}_A}}$ is invertible. By Proposition E.7.1, $\overline{(A+\Gamma)_{\mathcal{I}_A,\mathcal{J}_A}}$ is also invertible.

Analytical solution: By Proposition E.7.3, because $\widetilde{(A+\Gamma)_{\mathcal{I}_A,\mathcal{J}_A}}$ is invertible, $\widetilde{(A+\Gamma)_{\mathcal{I}_A,\mathcal{J}_A}}$ is also invertible. Let $\mathbf{q}_{\mathcal{J}_A} := \left(\widetilde{(A+\Gamma)_{\mathcal{I}_A,\mathcal{J}_A}}\right)^{-1}\mathbf{1}$. Let $\mathbf{q}$ be

$$\mathbf{q}_j = \begin{cases} \mathbf{q}_j & j \in \mathcal{J}_A \\ 0 & \text{o.w.} \end{cases} \tag{E.36}$$

We similarly define $\mathbf{p}$, with $\mathbf{p}_{\mathcal{I}_A} := \left(\widetilde{(A+\Gamma)^\top_{\mathcal{I}_A,\mathcal{J}_A}}\right)^{-1}\mathbf{1}$.

$(\mathbf{p},\mathbf{q})$ is a valid strategy pair: We first show that $\mathbf{q}$ is a valid strategy. By Proposition E.7.3, $\sum_{j\in\mathcal{J}_A} q_j = 1$, thus

$$\sum_{j\in[n]} q_j = 1 \tag{E.37}$$

$\widetilde{(A+\Gamma)_{\mathcal{I}_A,\mathcal{J}_A}}$ is a perturbed version of $\widetilde{A_{\mathcal{I}_A,\mathcal{J}_A}}$:

$$\begin{aligned}
&\left\|\widetilde{(A+\Gamma)_{\mathcal{I}_A,\mathcal{J}_A}} - \widetilde{A_{\mathcal{I}_A,\mathcal{J}_A}}\right\|_{\max} && \text{(E.38)}\\
=&\|A^\top_{\mathcal{I}_A,\mathcal{J}_A}\Gamma_{\mathcal{I}_A,\mathcal{J}_A} + \Gamma^\top_{\mathcal{I}_A,\mathcal{J}_A}A_{\mathcal{I}_A,\mathcal{J}_A} + \Gamma^\top_{\mathcal{I}_A,\mathcal{J}_A}\Gamma_{\mathcal{I}_A,\mathcal{J}_A} && \text{(E.39)}\\
&- \frac{1}{k}\left(A^\top_{\mathcal{I}_A,\mathcal{J}_A}\mathbf{1}\mathbf{1}^\top\Gamma_{\mathcal{I}_A,\mathcal{J}_A} + \Gamma^\top_{\mathcal{I}_A,\mathcal{J}_A}\mathbf{1}\mathbf{1}^\top A_{\mathcal{I}_A,\mathcal{J}_A} + \Gamma^\top_{\mathcal{I}_A,\mathcal{J}_A}\mathbf{1}\mathbf{1}^\top\Gamma_{\mathcal{I}_A,\mathcal{J}_A}\right)\|_{\max} && \text{(E.40)}\\
\le&2\left\|A^\top_{\mathcal{I}_A,\mathcal{J}_A}\Gamma_{\mathcal{I}_A,\mathcal{J}_A}\right\|_{\max} + \left\|\Gamma^\top_{\mathcal{I}_A,\mathcal{J}_A}\Gamma_{\mathcal{I}_A,\mathcal{J}_A}\right\|_{\max} + \frac{2}{k}\left\|A^\top_{\mathcal{I}_A,\mathcal{J}_A}\mathbf{1}\mathbf{1}^\top\Gamma_{\mathcal{I}_A,\mathcal{J}_A}\right\|_{\max} && \text{(E.41)}\\
&+ \frac{1}{k}\left\|\Gamma^\top_{\mathcal{I}_A,\mathcal{J}_A}\mathbf{1}\mathbf{1}^\top\Gamma_{\mathcal{I}_A,\mathcal{J}_A}\right\|_{\max} && \text{(E.42)}\\
\le&2\gamma\left\|A_{\mathcal{I}_A,\mathcal{J}_A}\right\|_1 + k\gamma^2 + \frac{2}{k}\left\|A_{\mathcal{I}_A,\mathcal{J}_A}\right\|_1 k\gamma + \frac{1}{k}k^2\gamma^2 && \text{(E.43)}\\
=&4\gamma\left\|A_{\mathcal{I}_A,\mathcal{J}_A}\right\|_1 + 2k\gamma^2 && \text{(E.44)}
\end{aligned}$$

By Proposition E.7.2, we have: $\mathbf{q}_{\mathcal{J}_A} > 0$. Thus $\mathbf{q}$ is a valid strategy. Similarly, $\mathbf{p}$ is also a valid strategy.

Constant value in support of NE: By Proposition E.7.3, $A_{\mathcal{I}_A,\mathcal{J}_A}\mathbf{q}_{\mathcal{J}_A} = A^\top_{\mathcal{I}_A,\mathcal{J}_A}\mathbf{p}_{\mathcal{I}_A} = \mathbf{p}^\top_{\mathcal{I}_A}A_{\mathcal{I}_A,\mathcal{J}_A}\mathbf{q}_{\mathcal{J}_A}\mathbf{1}$.

Switch out gap in perturbed game: By Proposition E.7.2,

$$\left\|\mathbf{q}^*_{\mathcal{J}_A} - \mathbf{q}_{\mathcal{J}_A}\right\|_\infty = \left\|\left(\widetilde{(A+\Gamma)_{\mathcal{I}_A,\mathcal{J}_A}}\right)^{-1}\mathbf{1} - \left(\widetilde{A_{\mathcal{I}_A,\mathcal{J}_A}}\right)^{-1}\mathbf{1}\right\|_\infty \tag{E.45}$$

$$\leq 2k^{3/2}\left(4\gamma\left\|A_{\mathcal{I}_A,\mathcal{J}_A}\right\|_1 + 2k\gamma^2\right)\left\|\widetilde{A_{\mathcal{I}_A,\mathcal{J}_A}}^{-1}\right\|_2\left\|\widetilde{A_{\mathcal{I}_A,\mathcal{J}_A}}^{-1}\mathbf{1}\right\|_\infty \tag{E.46}$$

for all $i \notin \mathcal{I}_A$,

$$\left|\mathbf{e}_i^\top A\mathbf{q}^* - \mathbf{e}_i^\top(A+\Gamma)\mathbf{q}\right| \leq \left|\mathbf{e}_i^\top A(\mathbf{q}^* - \mathbf{q})\right| + \left|\mathbf{e}_i^\top\Gamma\mathbf{q}\right| \tag{E.47}$$

$$\leq \left\|A_{\mathcal{I}_A{}^c,\cdot}(\mathbf{q}^* - \mathbf{q})\right\|_\infty + \left\|\Gamma\mathbf{q}\right\|_\infty \tag{E.48}$$

$$\leq \left\|A_{\mathcal{I}_A{}^c,\mathcal{J}_A}\right\|_\infty\left\|\mathbf{q}^*_{\mathcal{J}_A} - \mathbf{q}_{\mathcal{J}_A}\right\|_\infty + \gamma \tag{E.49}$$

$$\left|\mathbf{p}^{*\top}A\mathbf{q}^* - \mathbf{p}^\top(A+\Gamma)\mathbf{q}\right| = \left\|A_{\mathcal{I}_A,\mathcal{J}_A}\mathbf{q}^*_{\mathcal{J}_A} - (A_{\mathcal{I}_A,\mathcal{J}_A} + \Gamma_{\mathcal{I}_A,\mathcal{J}_A})\mathbf{q}_{\mathcal{J}_A}\right\|_\infty \tag{E.50}$$

$$\leq \left\|A_{\mathcal{I}_A,\mathcal{J}_A}(\mathbf{q}^*_{\mathcal{J}_A} - \mathbf{q}_{\mathcal{J}_A})\right\|_\infty + \left\|\Gamma_{\mathcal{I}_A,\mathcal{J}_A}\mathbf{q}_{\mathcal{J}_A}\right\|_\infty \tag{E.51}$$

$$\leq \left\|A_{\mathcal{I}_A,\mathcal{J}_A}\right\|_\infty\left\|\mathbf{q}^*_{\mathcal{J}_A} - \mathbf{q}_{\mathcal{J}_A}\right\|_\infty + \gamma \tag{E.52}$$

$$\mathbf{p}^\top(A+\Gamma)\mathbf{q} - \mathbf{e}_i^\top(A+\Gamma)\mathbf{q} \tag{E.53}$$

$$= \mathbf{p}^\top(A+\Gamma)\mathbf{q} - \mathbf{p}^{*\top}A\mathbf{q} + \mathbf{p}^{*\top}A\mathbf{q} - \mathbf{e}_i^\top A\mathbf{q}^* + \mathbf{e}_i^\top A\mathbf{q}^* - \mathbf{e}_i^\top(A+\Gamma)\mathbf{q} \tag{E.54}$$

$$\geq -\left|\mathbf{p}^\top(A+\Gamma)\mathbf{q} - \mathbf{p}^{*\top}A\mathbf{q}^*\right| + \mathbf{p}^{*\top}A\mathbf{q}^* - \mathbf{e}_i^\top A\mathbf{q}^* - \left|\mathbf{e}_i^\top A\mathbf{q}^* - \mathbf{e}_i^\top(A+\Gamma)\mathbf{q}\right| \tag{E.55}$$

$$\geq \Delta_{\mathcal{I}_A} - \left(\left\|A_{\mathcal{I}_A,\mathcal{J}_A}\right\|_\infty\left\|\mathbf{q}^*_{\mathcal{J}_A} - \mathbf{q}_{\mathcal{J}_A}\right\|_\infty + \gamma + \left\|A_{\mathcal{I}_A{}^c,\mathcal{J}_A}\right\|_\infty\left\|\mathbf{q}^*_{\mathcal{J}_A} - \mathbf{q}_{\mathcal{J}_A}\right\|_\infty + \gamma\right) \tag{E.56}$$

$$= \Delta_{\mathcal{I}_A} - \left(\left\|A_{\mathcal{I}_A,\mathcal{J}_A}\right\|_\infty + \left\|A_{\mathcal{I}_A{}^c,\mathcal{J}_A}\right\|_\infty\right)\left\|\mathbf{q}^*_{\mathcal{J}_A} - \mathbf{q}_{\mathcal{J}_A}\right\|_\infty - 2\gamma \tag{E.57}$$

$$\geq \Delta_{\mathcal{I}_A} - \left(\left\|A_{\mathcal{I}_A,\mathcal{J}_A}\right\|_\infty + \left\|A_{\mathcal{I}_A{}^c,\mathcal{J}_A}\right\|_\infty\right)2k^{3/2}\left(4\gamma\left\|A_{\mathcal{I}_A,\mathcal{J}_A}\right\|_1 + 2k\gamma^2\right) \tag{E.58}$$

$$\cdot \left\|\widetilde{A_{\mathcal{I}_A,\mathcal{J}_A}}^{-1}\right\|_2\left\|\widetilde{A_{\mathcal{I}_A,\mathcal{J}_A}}^{-1}\mathbf{1}\right\|_\infty - 2\gamma \tag{E.59}$$

$$> 0 \tag{E.60}$$

Similarly, $\mathbf{p}^\top(A+\Gamma)\mathbf{e}_j > \mathbf{p}^\top(A+\Gamma)\mathbf{q}, \forall j \notin \mathcal{J}_A$ Thus $(\mathbf{p},\mathbf{q})$ is NE.

$A+\Gamma$ **has unique NE:** By Theorem E.7.1, $(\mathbf{p},\mathbf{q})$ is the unique NE of $A+\Gamma$ and

thus $(\mathbf{p}, \mathbf{q}) = (\tilde{\mathbf{p}}, \tilde{\mathbf{q}})$.

NE recovery: One immediate observation is $\mathbf{supp}\,(\tilde{\mathbf{p}}) = \mathcal{I}_A$, $\mathbf{supp}\,(\tilde{\mathbf{q}}) = \mathcal{J}_A$.

By Proposition E.7.2,

$$d_{\mathrm{TV}}(\mathbf{q}^*, \tilde{\mathbf{q}}) = \frac{1}{2}\left\|\mathbf{q}^* - \tilde{\mathbf{q}}\right\|_1 = \frac{1}{2}\left\|\mathbf{q}^*_{\mathcal{J}_A} - \tilde{\mathbf{q}}_{\mathcal{J}_A}\right\|_1 \tag{E.61}$$

$$\leq k^{3/2}\left(4\gamma\left\|A_{\mathcal{I}_A,\mathcal{J}_A}\right\|_1 + 2k\gamma^2\right)\left\|\widetilde{A_{\mathcal{I}_A,\mathcal{J}_A}}^{-1}\right\|_2\left\|\widetilde{A_{\mathcal{I}_A,\mathcal{J}_A}}^{-1}\mathbf{1}\right\|_1 \tag{E.62}$$

Similarly,

$$d_{\mathrm{TV}}(\mathbf{p}^*, \tilde{\mathbf{p}}) \leq k^{3/2}\left(4\gamma\left\|A^\top_{\mathcal{I}_A,\mathcal{J}_A}\right\|_1 + 2k\gamma^2\right)\left\|\widetilde{A^\top_{\mathcal{I}_A,\mathcal{J}_A}}^{-1}\right\|_2\left\|\widetilde{A^\top_{\mathcal{I}_A,\mathcal{J}_A}}^{-1}\mathbf{1}\right\|_1 \tag{E.63}$$

■

E.6 Proof of Proposition 7.5.1

Proof of Proposition 7.5.1 follows similar steps in Proof of Theorem 4.3 in Cui and Du (2022) but replaces bonus with the estimation error upper bound of the trimmed mean estimation in Theorem 1 in Lugosi and Mendelson (2021).

E.7 Useful Results

Matrix Stability and Block Linear System

Proposition E.7.1 (Invertibility)**.** Let $A \in \mathbb{R}^{n\times n}$ be an invertible matrix. Let $E \in \mathbb{R}^{n\times n}$ s.t. $|E_{i,j}| \leq \epsilon$. Then if $n\epsilon < \sigma_{\min}$, where $\sigma_{\min}$ is the eigenvalue of A with the smallest absolute value, then $A + E$ is invertible.

Proof. Because

$$\det(A + E) = \det(A)\det(I + A^{-1}E) \tag{E.64}$$

$A + E$ is invertible if and only if $I + A^{-1}E$ is invertible.

We prove this by contradiction. Suppose $A + E$ is not invertible, then $I + A^{-1}E$ is not invertible. Then there exists $x \neq 0$, s.t. $(I + A^{-1}E)x = 0$. I.e. $x = -A^{-1}Ex$. But

$$\begin{aligned}\left\|A^{-1}Ex\right\|_2 \leq& \left\|A^{-1}E\right\|_2 \|x\|_2 \leq \left\|A^{-1}\right\|_2 \|E\|_2 \|x\|_2 \leq \sigma_{\min}^{-1} n \|E\|_{\max} \|x\|_2 &\text{(E.65)}\\ \leq& \sigma_{\min}^{-1} n\epsilon \|x\|_2 < \|x\|_2 . &\text{(E.66)}\end{aligned}$$

which is a contradiction. ■

Lemma E.7.1 (Residue of power series for inverse matrix)**.** *If* $B \in \mathbb{R}^{n\times n}$, $\|B\|_2 < 1$

$$\left\|B - B^2(I + B)^{-1}\right\|_\infty \leq \frac{\sqrt{n}\|B\|_2}{1 - \|B\|_2} \tag{E.67}$$

$$\left\|B - B^2(I + B)^{-1}\right\|_1 \leq \frac{\sqrt{n}\|B\|_2}{1 - \|B\|_2} \tag{E.68}$$

Proof. Let x be the eigenvector of the smallest eigenvalue, σ' of $I + B$, then

$$\begin{aligned}\sigma' =& \sigma' \|x\|_2 = \|(I + B)x\|_2 \geq \|x\|_2 - \|Bx\|_2 &\text{(E.69)}\\ \geq& 1 - \|B\|_2 \|x\|_2 = 1 - \|B\|_2 &\text{(E.70)}\end{aligned}$$

thus

$$\begin{aligned}\left\|B - B^2(I + B)^{-1}\right\|_2 \leq& \|B\|_2 + \left\|B^2(I + B)^{-1}\right\|_2 &\text{(E.71)}\\ \leq& \|B\|_2 + \|B\|_2^2 \left\|(I + B)^{-1}\right\|_2 &\text{(E.72)}\\ =& \|B\|_2 + \|B\|_2^2 \frac{1}{\sigma'} &\text{(E.73)}\\ \leq& \|B\|_2 + \frac{\|B\|_2^2}{1 - \|B\|_2} = \frac{\|B\|_2}{1 - \|B\|_2} &\text{(E.74)}\end{aligned}$$

thus

$$\left\|B - B^2(I+B)^{-1}\right\|_\infty \leq \sqrt{n}\left\|B - B^2(I+B)^{-1}\right\|_2 \leq \frac{\sqrt{n}\,\|B\|_2}{1-\|B\|_2} \tag{E.75}$$

$$\left\|B - B^2(I+B)^{-1}\right\|_1 \leq \sqrt{n}\left\|B - B^2(I+B)^{-1}\right\|_2 \leq \frac{\sqrt{n}\,\|B\|_2}{1-\|B\|_2} \tag{E.76}$$

■

we use $\min(x)$ to denote the minimum element of vector x.

Proposition E.7.2 (Support)**.** Suppose

$$\epsilon < \frac{1}{2n^{3/2}\left\|A^{-1}\right\|_2}\frac{\min(A^{-1}\mathbf{1})}{\left\|A^{-1}\mathbf{1}\right\|_\infty} \tag{E.77}$$

then $(A+E)^{-1}\mathbf{1}$ has only positive entries and

$$\left\|(A+E)^{-1}\mathbf{1} - A^{-1}\mathbf{1}\right\|_\infty \leq 2n^{3/2}\epsilon\left\|A^{-1}\right\|_2\left\|A^{-1}\mathbf{1}\right\|_\infty \tag{E.78}$$

$$\left\|(A+E)^{-1}\mathbf{1} - A^{-1}\mathbf{1}\right\|_1 \leq 2n^{3/2}\epsilon\left\|A^{-1}\right\|_2\left\|A^{-1}\mathbf{1}\right\|_1 \tag{E.79}$$

Proof.

$$\left\|A^{-1}E\right\|_2 \leq \left\|A^{-1}\right\|_2\|E\|_2 \leq \left\|A^{-1}\right\|_2 n\epsilon < \frac{1}{2\sqrt{n}}\frac{\min(A^{-1}\mathbf{1})}{\left\|A^{-1}\mathbf{1}\right\|_\infty} \leq \frac{1}{2} \tag{E.80}$$

$$(A+E)^{-1}\mathbf{1} = (I+A^{-1}E)^{-1}A^{-1}\mathbf{1} = \left(I - A^{-1}E + (A^{-1}E)^2(I+A^{-1}E)^{-1}\right)A^{-1}\mathbf{1} \tag{E.81}$$

$$= A^{-1}\mathbf{1} - \left(A^{-1}E - (A^{-1}E)^2(I+A^{-1}E)^{-1}\right)A^{-1}\mathbf{1} \tag{E.82}$$

By Lemma E.7.1,

$$\left\|\left(A^{-1}E - (A^{-1}E)^2(I+A^{-1}E)^{-1}\right)A^{-1}\mathbf{1}\right\|_\infty \tag{E.83}$$

$$\leq \left\|A^{-1}E - (A^{-1}E)^2(I+A^{-1}E)^{-1}\right\|_\infty\left\|A^{-1}\mathbf{1}\right\|_\infty \tag{E.84}$$

$$\leq \frac{\sqrt{n}\left\|A^{-1}E\right\|_2}{1-\left\|A^{-1}E\right\|_2}\left\|A^{-1}\mathbf{1}\right\|_\infty \tag{E.85}$$

$$< 2\sqrt{n}\left\|A^{-1}E\right\|_2\left\|A^{-1}\mathbf{1}\right\|_\infty \tag{E.86}$$

We have

$$\min\Big(A^{-1}\mathbf{1} - \Big(A^{-1}E - (A^{-1}E)^2(I+A^{-1}E)^{-1}\Big)A^{-1}\mathbf{1}\Big) \tag{E.87}$$

$$\geq \min(A^{-1}\mathbf{1}) - \left\|\Big(A^{-1}E - (A^{-1}E)^2(I+A^{-1}E)^{-1}\Big)A^{-1}\mathbf{1}\right\|_\infty \tag{E.88}$$

$$> \min(A^{-1}\mathbf{1}) - 2\sqrt{n}\left\|A^{-1}E\right\|_2\left\|A^{-1}\mathbf{1}\right\|_\infty > 0 \tag{E.89}$$

I.e. $(A+E)^{-1}\mathbf{1} > 0$.

$$\left\|(A+E)^{-1}\mathbf{1} - A^{-1}\mathbf{1}\right\|_1 = \left\|\Big(A^{-1}E - (A^{-1}E)^2(I+A^{-1}E)^{-1}\Big)A^{-1}\mathbf{1}\right\|_1 \tag{E.90}$$

$$\leq \left\|\Big(A^{-1}E - (A^{-1}E)^2(I+A^{-1}E)^{-1}\Big)\right\|_1\left\|A^{-1}\mathbf{1}\right\|_1 \tag{E.91}$$

$$\leq \frac{\sqrt{n}\left\|A^{-1}E\right\|_2}{1-\left\|A^{-1}E\right\|_2}\left\|A^{-1}\mathbf{1}\right\|_1 \tag{E.92}$$

$$< 2\sqrt{n}\left\|A^{-1}E\right\|_2\left\|A^{-1}\mathbf{1}\right\|_1 \tag{E.93}$$

$$\leq 2\sqrt{n}\left\|A^{-1}\right\|_2 n\epsilon\left\|A^{-1}\mathbf{1}\right\|_1 \tag{E.94}$$

$$= 2n^{3/2}\epsilon\left\|A^{-1}\right\|_2\left\|A^{-1}\mathbf{1}\right\|_1 \tag{E.95}$$

■

The following result adapts the idea in a stackexchange discussion[1]:

Proposition E.7.3 (Block linear system)**.** Let $A \in \mathbb{R}^{n\times n}$. If $\begin{pmatrix} A & -\mathbf{1} \\ \mathbf{1}^\top & 0 \end{pmatrix}$ is invertible and $\mathbf{x}, v$ satisfy

$$\begin{pmatrix} A & -\mathbf{1} \\ \mathbf{1}^\top & 0 \end{pmatrix}\begin{pmatrix} \mathbf{x} \\ v \end{pmatrix} = \begin{pmatrix} \mathbf{0} \\ 1 \end{pmatrix}, \tag{E.96}$$

[1]See *https://math.stackexchange.com/questions/411492/inverse-of-a-block-matrix-with-singular-diagonal-blocks*

then

$$\mathbf{x} = \left(A^\top A + \mathbf{1}\mathbf{1}^\top - \frac{1}{n}A^\top \mathbf{1}\mathbf{1}^\top A\right)^{-1}\mathbf{1}. \tag{E.97}$$

Proof. Let $X := \begin{pmatrix} A & -\mathbf{1} \\ \mathbf{1}^\top & 0 \end{pmatrix}$. Because X is invertible, we have:

$$X^{-1} = (X^\top X)^{-1}X^\top = \begin{pmatrix} A^\top A + \mathbf{1}\mathbf{1}^\top & -A^\top \mathbf{1} \\ -\mathbf{1}^\top A & n \end{pmatrix}^{-1} \begin{pmatrix} A^\top & \mathbf{1} \\ -\mathbf{1}^\top & 0 \end{pmatrix} \tag{E.98}$$

Because X is invertible, we know $X^\top X$ is p.d., thus $A^\top A + \mathbf{1}\mathbf{1}^\top$ is p.d. and thus invertible. Let $S := A^\top A + \mathbf{1}\mathbf{1}^\top - \frac{1}{n}A^\top \mathbf{1}\mathbf{1}^\top A$. By block matrix inversion (with Schur complement),

$$X^{-1} = \begin{pmatrix} S^{-1} & * \\ * & * \end{pmatrix} \begin{pmatrix} A^\top & \mathbf{1} \\ -\mathbf{1}^\top & 0 \end{pmatrix}. \tag{E.99}$$

Thus

$$\begin{pmatrix} \mathbf{x} \\ v \end{pmatrix} = X^{-1} \begin{pmatrix} \mathbf{0} \\ 1 \end{pmatrix} = \begin{pmatrix} S^{-1} & * \\ * & * \end{pmatrix} \begin{pmatrix} A^\top & \mathbf{1} \\ -\mathbf{1}^\top & 0 \end{pmatrix} \begin{pmatrix} \mathbf{0} \\ 1 \end{pmatrix} = \begin{pmatrix} S^{-1} & * \\ * & * \end{pmatrix} \begin{pmatrix} \mathbf{1} \\ 0 \end{pmatrix} = \begin{pmatrix} S^{-1}\mathbf{1} \\ * \end{pmatrix} \tag{E.100}$$

■

Remark E.7.1. If A is invertible, then

$$\mathbf{x} = \left(A^\top A + \mathbf{1}\mathbf{1}^\top - \frac{1}{n}A^\top \mathbf{1}\mathbf{1}^\top A\right)^{-1}\mathbf{1} = \frac{1}{\mathbf{1}^\top A^{-1}\mathbf{1}}A^{-1}\mathbf{1}. \tag{E.101}$$

The equation can be derived by:

$$\frac{1}{\mathbf{1}^\top A^{-1}\mathbf{1}}\left(A^\top A + \mathbf{1}\mathbf{1}^\top - \frac{1}{n}A^\top \mathbf{1}\mathbf{1}^\top A\right)A^{-1}\mathbf{1} \tag{E.102}$$

$$= \frac{1}{\mathbf{1}^\top A^{-1}\mathbf{1}}\left(A^\top + \mathbf{1}\mathbf{1}^\top A^{-1} - \frac{1}{n}A^\top \mathbf{1}\mathbf{1}^\top\right)\mathbf{1} \tag{E.103}$$

$$= \frac{1}{\mathbf{1}^\top A^{-1}\mathbf{1}}\left(A^\top \mathbf{1} + \mathbf{1}\mathbf{1}^\top A^{-1}\mathbf{1} - \frac{1}{n}A^\top \mathbf{1}\mathbf{1}^\top \mathbf{1}\right) \tag{E.104}$$

$$= \frac{1}{\mathbf{1}^\top A^{-1}\mathbf{1}}\left(\mathbf{1}\mathbf{1}^\top A^{-1}\mathbf{1}\right) = \mathbf{1}. \tag{E.105}$$

Uniqueness of NE

Theorem E.7.1. *Let* $(\mathbf{p}^*, \mathbf{q}^*)$ *be a strategy pair. Let* $\mathcal{I} = \mathbf{supp}(\mathbf{p}^*)$, $\mathcal{J} = \mathbf{supp}(\mathbf{q}^*)$. *Then* $(\mathbf{p}^*, \mathbf{q}^*)$ *is the unique NE of* A *if and only if the following two conditions hold:*

- ***Condition 1:***

$$\mathbf{e}_i^\top A\mathbf{q}^* = \mathbf{p}^{*\top} A\mathbf{q}^*, \forall i \in \mathcal{I} \tag{E.106}$$
$$\mathbf{p}^{*\top} A\mathbf{e}_j = \mathbf{p}^{*\top} A\mathbf{q}^*, \forall j \in \mathcal{J} \tag{E.107}$$
$$\mathbf{e}_i^\top A\mathbf{q}^* < \mathbf{p}^{*\top} A\mathbf{q}^*, \forall i \notin \mathcal{I} \tag{E.108}$$
$$\mathbf{p}^{*\top} A\mathbf{e}_j > \mathbf{p}^{*\top} A\mathbf{q}^*, \forall j \notin \mathcal{J} \tag{E.109}$$

- ***Condition 2:*** *the following block matrix is invertible:*

$$\begin{pmatrix} A_{\mathcal{I},\mathcal{J}} & \mathbf{1} \\ \mathbf{1} & 0 \end{pmatrix}. \tag{E.110}$$

Proof. "$\Leftarrow$": by Condition 1, we can show that $(\mathbf{p}^*, \mathbf{q}^*)$ is an NE. Suppose $(\mathbf{p}, \mathbf{q})$ is another NE.

Firstly, we show that $\mathbf{supp}(\mathbf{p}) \subseteq \mathcal{I}$ and $\mathbf{supp}(\mathbf{q}) \subseteq \mathcal{J}$. We prove this by contradiction.

Because $(\mathbf{p}^*, \mathbf{q}^*)$ and $(\mathbf{p}, \mathbf{q})$ are two NEs, $\mathbf{p}^\top A\mathbf{q}^* = \mathbf{p}^{*\top} A\mathbf{q}^* = v^*$. Suppose $\mathbf{supp}(\mathbf{p}) \cap ([m] \setminus \mathcal{I}) \neq \varnothing$. By **Condition 1**, $\mathbf{p}^\top A\mathbf{q}^* < \mathbf{p}^{*\top} A\mathbf{q}^*$, which contradicts with: $\mathbf{p}^\top A\mathbf{q}^* = \mathbf{p}^{*\top} A\mathbf{q}^*$. Thus $\mathbf{supp}(\mathbf{p}) \subseteq \mathcal{I}$. Similarly, we can show $\mathbf{supp}(\mathbf{q}) \subseteq \mathcal{J}$.

Thus it's sufficient to consider the following sub-LPs: primal LP for the row player:

$$\max_{v, \mathbf{p}'_\mathcal{I}} v \tag{E.111}$$
$$\text{s.t.} \mathbf{p}'_\mathcal{I} \in \Delta(\mathcal{I}) \tag{E.112}$$
$$A_{\mathcal{I},\cdot}^\top \mathbf{p}'_\mathcal{I} \geq v \tag{E.113}$$

dual LP for the column player:

$$\min_{v,\mathbf{q}'_{\mathcal{J}}} v \tag{E.114}$$

$$\text{s.t.} \mathbf{q}'_{\mathcal{J}} \in \Delta(\mathcal{J}) \tag{E.115}$$

$$A_{\cdot,\mathcal{J}}\mathbf{q}'_{\mathcal{J}} \leq v \tag{E.116}$$

where we use $\mathbf{p}'_{\mathcal{I}}$ to denote the vector obtained by selecting the entries of $\mathbf{p}'$ from index set $\mathcal{I}$. $\mathbf{q}'_{\mathcal{J}}$ is defined similarly. $A_{\mathcal{I},\mathcal{J}}$ is a submatrix of A obtained by selecting the rows in set $\mathcal{I}$ and columns in set $\mathcal{J}$.

We now show that $(\mathbf{p}, \mathbf{q}) = (\mathbf{p}^*, \mathbf{q}^*)$. We prove this by contradiction.

W.l.o.g., assume $\mathbf{q} \neq \mathbf{q}^*$. Consider the dual LP for the column player. Suppose constraints $A_{\mathcal{I},\mathcal{J}}\mathbf{q}'_{\mathcal{J}} \leq v$ are all active at $\mathbf{q}$, i.e. $A_{\mathcal{I},\mathcal{J}}\mathbf{q}_{\mathcal{J}} = v$. Then $(\mathbf{q}_{\mathcal{J}}, v^*)$ is a solution to:

$$\begin{pmatrix} A_{\mathcal{I},\mathcal{J}} & -\mathbf{1} \\ \mathbf{1} & 0 \end{pmatrix} \begin{pmatrix} \mathbf{q}'_{\mathcal{J}} \\ v \end{pmatrix} = \begin{pmatrix} \mathbf{0} \\ 1 \end{pmatrix} \tag{E.117}$$

By definition $(\mathbf{q}^*_{\mathcal{J}}, v^*)$ is also a solution to (E.117). By **Condition 2**, we have $\mathbf{q}_{\mathcal{J}} = \mathbf{q}^*_{\mathcal{J}}$, and thus $\mathbf{q} = \mathbf{q}^*$, which leads to a contradiction. Thus constraints $A_{\mathcal{I},\mathcal{J}}\mathbf{q}'_{\mathcal{J}} \leq v$ are not all active at $\mathbf{q}_{\mathcal{J}}$. This means $(\mathbf{q}_{\mathcal{J}}, v^*)$ satisfies:

$$\begin{pmatrix} A_{\mathcal{I},\mathcal{J}} & -\mathbf{1} \\ \mathbf{1} & 0 \end{pmatrix} \begin{pmatrix} \mathbf{q}_{\mathcal{J}} \\ v^* \end{pmatrix} = \begin{pmatrix} -\boldsymbol{\alpha} \\ 1 \end{pmatrix}, \tag{E.118}$$

where $\boldsymbol{\alpha} \geq 0$ and has at least one non-zero entry. Thus

$$A_{\mathcal{I},\mathcal{J}}\mathbf{q}_{\mathcal{J}} = \mathbf{1}v^* - \boldsymbol{\alpha} \tag{E.119}$$

Because $(\mathbf{p}^*, \mathbf{q}^*)$ and $(\mathbf{p}, \mathbf{q})$ are two NEs, we know $(\mathbf{p}^*, \mathbf{q})$ is also an NE. Thus

$$v^* = \mathbf{p}^{*\top} A \mathbf{q} = \mathbf{p}^{*\top}_{\mathcal{I}} A_{\mathcal{I},\mathcal{J}} \mathbf{q}_{\mathcal{J}} = \mathbf{p}^{*\top}_{\mathcal{I}} (\mathbf{1}v^* - \boldsymbol{\alpha}) = v^* - \mathbf{p}^{*\top}_{\mathcal{I}} \boldsymbol{\alpha} \tag{E.120}$$

because $\mathbf{p}^{*\top}_{\mathcal{I}} > 0$ and $\boldsymbol{\alpha}$ has at least one strict positive entry, we have $v^* > v^* - \mathbf{p}^{*\top}_{\mathcal{I}} \boldsymbol{\alpha}$, which leads to a contradiction. Thus $\mathbf{q} = \mathbf{q}^*$.

"⇒": By the definition of NE, we naturally have:

$$\mathbf{e}_i^\top A\mathbf{q}^* = \mathbf{p}^{*\top} A\mathbf{q}^* = v^*, \forall i \in \mathcal{I} \tag{E.121}$$

$$\mathbf{p}^{*\top} A\mathbf{e}_j = \mathbf{p}^{*\top} A\mathbf{q}^* = v^*, \forall j \in \mathcal{J} \tag{E.122}$$

By Corollary 3A of Goldman and Tucker (2016), there exists an NE $(\mathbf{p}, \mathbf{q})$, s.t. if $\mathbf{e}_i^\top A\mathbf{q} = v^*$, then $i \in \mathcal{I}$; if $\mathbf{p}^\top A\mathbf{e}_j = v^*$, then $j \in \mathcal{J}$. These together with (E.121) and (E.122) show that $(\mathbf{p}, \mathbf{q})$ satisfies **Condition 1**. Because $(\mathbf{p}^*, \mathbf{q}^*)$ is the unique NE, we have $(\mathbf{p}^*, \mathbf{q}^*) = (\mathbf{p}, \mathbf{q})$, thus $(\mathbf{p}^*, \mathbf{q}^*)$ satisfies **Condition 1**.

If $|\mathcal{I}| = |\mathcal{J}| = 1$, **Condition 2** naturally holds. Suppose **Condition 2** does not hold, i.e. one of the following holds:

- $|\mathcal{I}| = |\mathcal{J}| > 1$ and $\begin{pmatrix} A_{\mathcal{I},\mathcal{J}} & \mathbf{1} \\ \mathbf{1} & 0 \end{pmatrix}$ is not invertible
- $|\mathcal{I}| < |\mathcal{J}|$
- $|\mathcal{I}| > |\mathcal{J}|$

we can construct another NE. If $\begin{pmatrix} A_{\mathcal{I},\mathcal{J}} & \mathbf{1} \\ \mathbf{1} & 0 \end{pmatrix}$ is not invertible or $|\mathcal{I}| < |\mathcal{J}|$ (the case when $|\mathcal{I}| > |\mathcal{J}|$ can be analyzed similarly) , homogeneous linear system:

$$\begin{pmatrix} A_{\mathcal{I},\mathcal{J}} & -\mathbf{1} \\ \mathbf{1} & 0 \end{pmatrix} \begin{pmatrix} \Delta\mathbf{q}_{\mathcal{J}} \\ \Delta v \end{pmatrix} = \begin{pmatrix} \mathbf{0} \\ 0 \end{pmatrix} \tag{E.123}$$

has a nonzero solution, $\begin{pmatrix} \Delta\bar{\mathbf{q}}_{\mathcal{J}} \\ \Delta\bar{v} \end{pmatrix}$ (at least one entry of $\Delta\bar{\mathbf{q}}_{\mathcal{J}}$ has to be nonzero, otherwise, it's a zero solution).

Because:

1. $0 < \mathbf{q}_j^* < 1$, for all $j \in \mathcal{J}$;
2. $\mathbf{e}_i^\top A\mathbf{q}^* < v^*$ for all $i \notin \mathcal{I}$ (by **Condition 1**),

we can find an $\alpha > 0$ small enough, s.t. $\begin{pmatrix} \Delta\tilde{\mathbf{q}}_{\mathcal{J}} \\ \Delta\tilde{v} \end{pmatrix} := \alpha \begin{pmatrix} \Delta\bar{\mathbf{q}}_{\mathcal{J}} \\ \Delta\bar{v} \end{pmatrix}$ satisfies:

1. $\mathbf{q}^*_{\mathcal{J}} + \Delta\tilde{\mathbf{q}}_{\mathcal{J}}$ is a valid probability distribution on $\mathcal{J}$ and $\mathbf{q}^*_{\mathcal{J}} + \Delta\tilde{\mathbf{q}}_{\mathcal{J}} \neq \mathbf{q}^*_{\mathcal{J}}$;
2. Let $\Delta\tilde{\mathbf{q}} \in \mathbb{R}^n$, s.t. $\Delta\tilde{\mathbf{q}}_j = \begin{cases} \Delta\tilde{\mathbf{q}}_j & j \in \mathcal{J} \\ 0 & \text{o.w.} \end{cases}$, then

$$\mathbf{e}_i^\top A(\mathbf{q}^* + \Delta\tilde{\mathbf{q}}) < v^*, \quad \forall i \notin \mathcal{I} \tag{E.124}$$

Note that

$$\mathbf{p}^{*\top} A(\mathbf{q}^* + \Delta\tilde{\mathbf{q}}) = v^* + \mathbf{p}^{*\top} A\Delta\tilde{\mathbf{q}} = v^* + \mathbf{p}^{*\top}_{\mathcal{I}} A_{\mathcal{I},\mathcal{J}} \Delta\tilde{\mathbf{q}}_{\mathcal{J}} = v^* + v^* \mathbf{1}^\top \Delta\tilde{\mathbf{q}}_{\mathcal{J}} = v^* \tag{E.125}$$

We now show that $(\mathbf{p}^*, \mathbf{q}^* + \Delta\tilde{\mathbf{q}})$ is also an NE: for all $\mathbf{q} \in \Delta(n)$,

$$\mathbf{p}^{*\top} A\mathbf{q} \geq v^* = \mathbf{p}^{*\top} A(\mathbf{q}^* + \Delta\tilde{\mathbf{q}}) \tag{E.126}$$

and for all $\mathbf{p} \in \Delta(m)$, by (E.124),

$$\mathbf{p}^\top A(\mathbf{q}^* + \Delta\tilde{\mathbf{q}}) \leq v^* = \mathbf{p}^{*\top} A(\mathbf{q}^* + \Delta\tilde{\mathbf{q}}) \tag{E.127}$$

■

F.1 Proof of Theorem 8.3.1

In this section, we prove Theorem 8.3.1. This section is organized as follows. First, in Section F.1, we consider the case where $m \leq 4$. In the remainder of this section, we will assume $m \geq 5$. First, in Section F.1, we will show that (8.7) can be solved for α and state some properties about the solution. Then, in Section F.1, we will prove the Nash incentive compatibility result, in Section F.1 we will prove individual rationality, and in Section F.1, we will prove the result on efficiency.

When $m \leq 4$

First, consider the (easy) case $m \leq 4$. At $s_i^\star$, the total amount of data collected is $\sigma/\sqrt{c}$ as each agent will be collecting $n_i^\star = \frac{\sigma}{m\sqrt{c}}$ (see (8.8)). As there is no corrupted dataset, $h_i^\star$ simply reduces to the sample mean of $X_i \cup Y_{-i}$. The individual rationality property follows from the following simple calculation:

$$p_i(M_{\text{C3D}}, s^\star) = \left(1 + \frac{1}{m}\right)\sqrt{c}\sigma < 2\sqrt{c}\sigma = p_{\min}^{\text{IR}}.$$

Similarly, the bound on the ratio between the penalties can also be obtained via the following calculation:

$$\text{PR} = \frac{m\left(1 + \frac{1}{m}\right)\sqrt{c}\sigma}{2\sigma\sqrt{cm}} < \sqrt{m} \leq 2.$$

Finally, to show NIC, consider agent i and assume that all other agents have followed the recommended strategies, i.e collected $\sigma/(m\sqrt{c})$. Then, the agent will have an *uncorrupted* dataset $Y_{-i} = \bigcup_{j \neq i} X_j$ of $n_{-i}^\star = (m-1)\sigma/(m\sqrt{c})$ points with no corruption. Regardless of what she chooses to submit, the best estimator she could use with the union of this dataset Y_{-i} and the data she collects X_i and will be the sample mean as it is minimax optimal. The number of points that minimizes her

penalty is,

$$\operatorname*{argmin}_{n_i}\Big(\sup_{\mu}\mathbb{E}\left[(h_i(X_i,Y_i,Y_{-i})-\mu)^2\;\middle|\;\mu\right]\;+\;cn_i\Big)=\operatorname*{argmin}_{n_i\in\mathbb{R}}\Big(\frac{\sigma^2}{n_i+n_{-i}^\star}+cn_i\Big)=\frac{\sigma}{m\sqrt{c}}$$

Finally, as A_i does not depend on f_i under these conditions, there is no incentive to fabricate or falsify data, i.e choosing anything other than $f^\star=\mathbf{I}$ does not lower her utility.

In the remainder of this section, will study the harder case, $m\geq 4$.

Existence of a solution to (8.7) and some of its properties

In this section, we show that $G\Big(\frac{\sigma^{1/2}}{(cm)^{1/4}}\Big)<0$ and $G\Big(\Big(1+\frac{C_m}{m}\Big)\frac{\sigma^{1/2}}{(cm)^{1/4}}\Big)>0$, where $C_m=20$ when $m\leq 20$ and $C_m=5$ when $m>20$. This means equation $G(\alpha)=0$ has solution in $\Big(\frac{\sigma^{1/2}}{(cm)^{1/4}},\;\Big(1+\frac{C_m}{m}\Big)\frac{\sigma^{1/2}}{(cm)^{1/4}}\Big)$.

First, in Lemma F.7.5, we derive an asymptotic expansion of the Gaussian complementary error function, and construct lower and upper bounds for $G(\alpha)$ that are easier to work with. We have restated these lower ($\mathrm{Erfc}_{\mathrm{LB}}$) and upper ($\mathrm{Erfc}_{\mathrm{UB}}$) bounds below.

$$\mathrm{Erfc}_{\mathrm{UB}}(x):=\frac{1}{\sqrt{\pi}}\left(\frac{\exp(-x^2)}{x}-\frac{\exp(-x^2)}{2x^3}+\frac{3\exp(-x^2)}{4x^5}\right)\tag{F.1}$$

$$\mathrm{Erfc}_{\mathrm{LB}}(x):=\frac{1}{\sqrt{\pi}}\left(\frac{\exp(-x^2)}{x}-\frac{\exp(-x^2)}{2x^3}\right)\tag{F.2}$$

We can now use this to derive the following lower (G_{LB}) and upper (G_{UB}) bounds on G. Here, we have used the fact that $4(m+1)\frac{\alpha^2}{\sigma\sqrt{m/c}}-1>0$ when $\alpha\geq(\sigma/\sqrt{cm})^{1/2}$. We have:

$$\begin{aligned}G_{\mathrm{LB}}(\alpha):=&\left(\frac{m-4}{m-2}\frac{4\alpha^2}{\sigma/\sqrt{cm}}-1\right)\frac{4\alpha}{\sqrt{\sigma}(m/c)^{1/4}}\\&-\left(4(m+1)\frac{\alpha^2}{\sigma\sqrt{m/c}}-1\right)\sqrt{2\pi}\exp\left(\frac{\sigma\sqrt{m/c}}{8\alpha^2}\right)\mathrm{Erfc}_{\mathrm{UB}}\left(\frac{\sqrt{\sigma}(m/c)^{1/4}}{2\sqrt{2}\alpha}\right),\end{aligned}$$

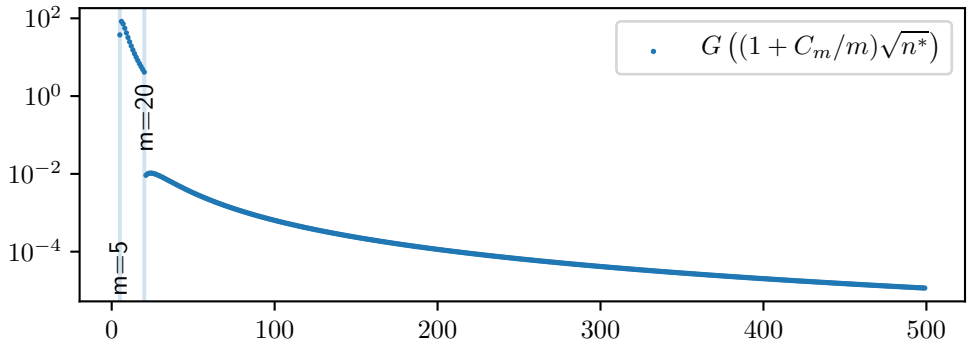


Figure F.1: Plot for $G\Big(\Big(1+\frac{C_m}{m}\Big)\frac{\sigma^{1/2}}{(cm)^{1/4}}\Big)$. See `G_em_plot.py`. The discontinuity at $m = 20$ is due to the different values for C_m when $m \le 20$ and when $m > 20$.

$$G_{\mathrm{UB}}(\alpha) := \left(\frac{m-4}{m-2}\frac{4\alpha^2}{\sigma/\sqrt{cm}} - 1\right)\frac{4\alpha}{\sqrt{\sigma}(m/c)^{1/4}} - \left(4(m+1)\frac{\alpha^2}{\sigma\sqrt{m/c}} - 1\right)\sqrt{2\pi}\exp\left(\frac{\sigma\sqrt{m/c}}{8\alpha^2}\right)\mathrm{Erfc}_{\mathrm{LB}}\left(\frac{\sqrt{\sigma}(m/c)^{1/4}}{2\sqrt{2}\alpha}\right).$$

By first, substituting $\sigma/\sqrt{cm}$ for α in the expressions for G_{UB} and $\mathrm{Erfc}_{\mathrm{UB}}$, and then via a sequence of algebraic manipulations, we can verify that

$$\begin{aligned} &G\left(\frac{\sigma^{1/2}}{(cm)^{1/4}}\right) \le G_{\mathrm{UB}}\left(\frac{\sigma^{1/2}}{(cm)^{1/4}}\right) \\ =&\frac{4\left(\frac{4(m-4)}{m-2}-1\right)\left(\frac{\sigma}{\sqrt{cm}}\right)^{1/2}}{\sqrt{\sigma}\left(\frac{m}{c}\right)^{1/4}} - \sqrt{2}\left(\frac{4(m+1)}{\sqrt{\frac{m}{c}}\sqrt{cm}}-1\right)\left(\frac{2\sqrt{2}\left(\frac{\sigma}{\sqrt{cm}}\right)^{1/2}}{\sqrt{\sigma}\left(\frac{m}{c}\right)^{1/4}} - \frac{8\sqrt{2}\left(\frac{\sigma}{\sqrt{cm}}\right)^{3/2}}{\sigma^{3/2}\left(\frac{m}{c}\right)^{3/4}}\right) \\ =&-\frac{128}{(m-2)m^{5/2}} < 0. \end{aligned}$$

Next, we will show that $G\Big(\Big(1+\frac{C_m}{m}\Big)\frac{\sigma^{1/2}}{(cm)^{1/4}}\Big) > 0$ by studying the lower bound G_{LB}. For $m \in [5, 500]$, we can verify individually that $G\Big(\Big(1+\frac{C_m}{m}\Big)\frac{\sigma^{1/2}}{(cm)^{1/4}}\Big) > 0$ (See Figure F.1). For $m > 500$, we have:

$$
\begin{aligned}
&G\left(\left(1+\frac{C_m}{m}\right)\frac{\sigma^{1/2}}{(cm)^{1/4}}\right) = G\left(\left(1+\frac{5}{m}\right)\frac{\sigma^{1/2}}{(cm)^{1/4}}\right) \geq G_{\mathrm{LB}}\left(\left(1+\frac{5}{m}\right)\frac{\sigma^{1/2}}{(cm)^{1/4}}\right)\\
=&\frac{4\left(\frac{4\left(\frac{5}{m}+1\right)^2(m-4)}{m-2}-1\right)\left(\frac{5}{m}+1\right)\left(\frac{\sigma}{\sqrt{cm}}\right)^{1/2}}{\sqrt{\sigma}\left(\frac{m}{c}\right)^{1/4}}\\
&-\sqrt{2}\left(\frac{4\left(\frac{5}{m}+1\right)^2(m+1)}{\sqrt{\frac{m}{c}}\sqrt{cm}}-1\right)\left(\frac{2\sqrt{2}\left(\frac{5}{m}+1\right)\left(\frac{\sigma}{\sqrt{cm}}\right)^{1/2}}{\sqrt{\sigma}\left(\frac{m}{c}\right)^{1/4}}-\frac{8\sqrt{2}\left(\frac{5}{m}+1\right)^3\left(\frac{\sigma}{\sqrt{cm}}\right)^{3/2}}{\sigma^{3/2}\left(\frac{m}{c}\right)^{3/4}}\right.\\
&\left.+\frac{96\sqrt{2}\left(\frac{5}{m}+1\right)^5\left(\frac{\sigma}{\sqrt{cm}}\right)^{5/2}}{\sigma^{5/2}\left(\frac{m}{c}\right)^{5/4}}\right)\\
=&\frac{64(m+5)^3\left(m^6-191m^5-1566m^4-3920m^3+2100m^2+19500m+15000\right)}{(m-2)m^{21/2}}.
\end{aligned}
$$

When $m > 500$,

$$
\begin{aligned}
&m^6-191m^5-1566m^4-3920m^3 = m^3(m^3-191m^2-1566m-3920)\\
>&m^3((200+200+100)m^2-191m^2-1566m-3920)\\
>&m^3(200m^2+10^5m+2.5\times10^7-191m^2-1566m-3920)>0.
\end{aligned}
$$

Combining the results from the two previous displays, we have, $G\left(\left(1+\frac{C_m}{m}\right)\frac{\sigma^{1/2}}{(cm)^{1/4}}\right) > 0$ which completes the proof for this section.

Algorithm 10 is Nash incentive compatible

In this section, we will prove the following lemma which states that $s_i^\star$, as defined in (8.8) is a Nash equilibrium in M_{C3D}.

Lemma F.1.1 (NIC). *The recommended strategies $s^\star = \{(n_i^\star, f_i^\star, h_i^\star)\}_i$ as defined in (8.8) in mechanism M_{C3D} (Algorithm 10) satisfies:*

$$
p_i(M_{\text{C3D}}, s^\star) \leq p_i(M_{\text{C3D}}, (s_i, s^\star_{-i}))
$$

for all $i \in [m]$ *and* $s_i \in \mathbb{N} \times \mathcal{F} \times \mathcal{H}$.

The Proof of Lemma F.1.1 relies on the following two lemmas:

Lemma F.1.2 (Optimal Estimation and Submission)**.** *For all* $i \in [m]$ *and* $(n_i, f_i, h_i) \in \mathbb{N} \times \mathcal{F} \times \mathcal{H}$.

$$p_i(M_{\text{C3D}}, ((n_i, f_i^\star, h_i^\star), s_{-i}^\star)) \leq p_i(M_{\text{C3D}}, ((n_i, f_i, h_i), s_{-i}^\star)).$$

See the Proof of Lemma F.1.2 in Section F.1

Lemma F.1.3 (Optimal Sample Size)**.** *For all* $i \in [m]$ *and* $n_i \in \mathbb{N}$.

$$p_i(M_{\text{C3D}}, ((n_i^\star, f_i^\star, h_i^\star), s_{-i}^\star)) \leq p_i(M_{\text{C3D}}, ((n_i, f_i^\star, h_i^\star), s_{-i}^\star)).$$

See the Proof of Lemma F.1.3 in Section F.1

Proof of Lemma F.1.1. By Lemma F.1.2 and F.1.3, we have, for all $i \in [m]$ and $s_i' = (n_i, f_i, h_i) \in \mathbb{N} \times \mathcal{F} \times \mathcal{H}$,

$$\begin{aligned} p_i(M_{\text{C3D}}, s^\star) &= p_i(M_{\text{C3D}}, ((n_i^\star, f_i^\star, h_i^\star), s_{-i}^\star)) \leq p_i(M_{\text{C3D}}, ((n_i, f_i^\star, h_i^\star), s_{-i}^\star)) \\ &\leq p_i(M_{\text{C3D}}, ((n_i, f_i, h_i), s_{-i}^\star)) = p_i(M_{\text{C3D}}, (s_i', s_{-i}^\star)) \end{aligned}$$

■

Proof of Lemma F.1.2

In this section, we will prove Lemma F.1.2, which, intuitively states that, regardless of the amount of data collected, agent i should submit the data as is ($f_i^\star = \mathbf{I}$) and use the weighted average estimator in (8.8) to estimate μ. We will do so via the following three step procedure, inspired by well–known techniques for proving minimax optimality of estimators (e.g see Theorem 1.12, Chapter 5 of Lehmann and Casella (2006)).

1. First, we construct a sequence of prior distributions $\{\Lambda_\ell\}_{\ell \geq 1}$ for μ and calculate the sequence of Bayesian risks under the prior distributions:

$$R_\ell := \inf_{f_i \in \mathcal{A}, h_i \in \mathcal{H}} \mathbb{E}_{\mu \sim \Lambda_\ell}\Big[\mathbb{E}\Big[(h_i(X_i, f_i(X_i), A_i) - \mu)^2 \big| \mu\Big]\Big], \quad \ell \geq 1.$$

2. Then, we will show that $\lim_{\ell \to \infty} R_\ell = \sup_\mu \mathbb{E}[(h_i^\star(X_i, f_i^\star(X_i), A_i) - \mu)^2 \,|\, \mu]$.

3. Finally, as the Bayesian risk is a lower bound on maximum risk, we will conclude that $(f_i^\star, h_i^\star)$ is minimax optimal.

Without loss of generality, we focus only on the deterministic f_i and h_i. If either of them are stochastic, we can condition on the external source of randomness and treat them as deterministic functions. Our proof holds for any realization of this external source of randomness, and hence it will hold in expectation as well. Similarly, Z_i is randomly chosen in Algorithm 10. In the following, we condition on this randomness and the entire proof will carry through.

Note that $Y_i = f_i(X_i)$. We will use both of them interchangeably in the subsequent proof.

Step 1 (Bounding the Bayes' risk under the sequence of priors): We will use a sequence of normal priors $\Lambda_\ell := \mathcal{N}(0, \ell^2)$ for all $\ell \geq 1$. To bound the Bayes' risk under these priors, we will first note that for a fixed $f_i \in \mathcal{F}$,

$$x|\mu \sim \mathcal{N}(\mu, \sigma^2) \quad \forall x \in X_i \cup Z_i; \tag{F.3}$$

$$x|\mu, \eta_i^2 \sim \mathcal{N}(\mu, \sigma^2 + \eta_i^2) \quad \forall x \in Z_i'. \tag{F.4}$$

Here, recall that η_i^2 is a function of Y_i and Z_i. Because both $Y_i = f_i(X_i)$ and η_i^2 are deterministic functions of X_i, Z_i when f_i is fixed, the posterior distribution for μ conditioned on (X_i, Y_i, A_i) can be calculated as follows:

$$\begin{aligned} p(\mu|X_i, Y_i, A_i) &= p\Big(\mu|X_i, Y_i, Z_i, Z_i', \eta_i^2\Big) = p(\mu|X_i, Z_i, Z_i') \\ &\propto p(\mu, X_i, Z_i, Z_i') = p(Z_i'|X_i, Z_i, \mu)p(X_i, Z_i|\mu)p(\mu) = p(Z_i'|X_i, Z_i, \mu)p(X_i|\mu)p(Z_i|\mu)p(\mu) \end{aligned}$$

$$\propto \exp\left(-\frac{1}{2(\sigma^2+\eta_i^2)}\sum_{x\in Z_i'}(x-\mu)^2\right)\exp\left(-\frac{1}{2\sigma^2}\sum_{x\in X_i\cup Z_i}(x-\mu)^2\right)\exp\left(-\frac{\mu^2}{2\ell^2}\right)$$
$$\propto \exp\left(-\frac{1}{2}\left(\frac{|Z_i'|}{\sigma^2+\eta_i^2}+\frac{|X_i|+|Z_i|}{\sigma^2}+\frac{1}{\ell^2}\right)\mu^2\right)\exp\left(\frac{1}{2}2\left(\frac{\sum_{x\in Z_i'}x}{\sigma^2+\eta_i^2}+\frac{\sum_{x\in X_i\cup Z_i}x}{\sigma^2}\right)\mu\right)$$
$$= \exp\left(-\frac{1}{2}\left(\frac{1}{\sigma_\ell^2}\mu^2-2\frac{\mu_\ell}{\sigma_\ell^2}\mu\right)\right)\propto\exp\left(-\frac{1}{2\sigma_\ell^2}(\mu-\mu_\ell)^2\right),$$

where

$$\mu_\ell=\frac{\frac{\sum_{x\in Z_i'}x}{\sigma^2+\eta_i^2}+\frac{\sum_{x\in X_i\cup Z_i}x}{\sigma^2}}{\frac{|Z_i'|}{\sigma^2+\eta_i^2}+\frac{|X_i|+|Z_i|}{\sigma^2}+\frac{1}{\ell^2}},\quad\text{and}\quad\sigma_\ell^2=\frac{1}{\frac{|Z_i'|}{\sigma^2+\eta_i^2}+\frac{|X_i|+|Z_i|}{\sigma^2}+\frac{1}{\ell^2}}.\tag{F.5}$$

We can therefore conclude that (despite the non i.i.d nature of the data), the posterior for μ is Gaussian with mean and variance as shown above. We have:

$$\mu|X_i,Y_i,A_i\sim\mathcal{N}(\mu_\ell,\sigma_\ell^2).$$

Next, following standard steps (See Corollary 1.2 in Chapter 4 of Lehmann and Casella (2006)), we know that $\mathbb{E}_\mu\Big[(h_i(X_i,Y_i,A_i)-\mu)^2|X_i,Y_i,A_i\Big]$ is minimized when $h_i(X_i,Y_i,A_i)=\mathbb{E}_\mu[\mu|X_i,Y_i,A_i]=\mu_\ell$. This shows that for any $f_i\in h_i$, the optimal h_i is simply the posterior mean of μ under the prior Λ_ℓ conditioned on $(X_i,f_i(X_i),A_i)$. We can rewrite the minimum averaged risk over $\mathcal{H}$ by switching the order of expectation:

$$\begin{aligned}&\inf_{h_i\in\mathcal{H}}\mathbb{E}_{\mu\sim\Lambda_\ell}\Big[\mathbb{E}\Big[(h_i(X_i,Y_i,A_i)-\mu)^2|\mu\Big]\Big]\\&=\inf_{h_i\in\mathcal{H}}\mathbb{E}_{X_i,Z_i,Z_i'}\Big[\mathbb{E}_\mu\Big[(h_i(X_i,Y_i,A_i)-\mu)^2|X_i,Z_i,Z_i'\Big]\Big]\\&=\mathbb{E}_{X_i,Z_i,Z_i'}\Big[\mathbb{E}_\mu\Big[(\mu_\ell-\mu)^2|X_i,Z_i,Z_i'\Big]\Big]=\mathbb{E}_{X_i,Z_i,Z_i'}\Big[\sigma_\ell^2\Big]\\&=\mathbb{E}_{X_i,Z_i}\left[\frac{1}{\frac{|Z_i'|}{\sigma^2+\eta_i^2}+\frac{|X_i|+|Z_i|}{\sigma^2}+\frac{1}{\ell^2}}\right],\end{aligned}\tag{F.6}$$

the expectation in the last step involves only X_i, Z_i because σ_ℓ^2 depends only on X_i, Z_i and $|Z_i'|$, but not the instantiation of Z_i'.

Next, we will show that (F.6) is minimized for the following choice of f_i which shrinks each points in X_i by an amount that depends on the prior Λ_ℓ's variance ℓ^2:

$$f_i(X_i) = \left\{ \frac{|X_i|/\sigma^2}{|X_i|/\sigma^2 + 1/\ell^2}\, x \text{ , for each } x \in X_i \right\}. \tag{F.7}$$

Remark F.1.1. An interesting observation (albeit not critical to the proof) here is that f_i in (F.7) converges pointwise to $f_i^\star$, i.e. **I**, as $\ell \to \infty$. This shows that the optimal submission function under the prior converges to $f_i^\star$. We can make a similar observation about the posterior mean in (F.5), where μ_ℓ converges to $h_i^\star$ as $\ell \to \infty$.

To prove (F.7), we first define the following quantities.

$$\widehat{\mu}(X_i) := \frac{1}{|X_i|}\sum_{x\in X_i} x, \quad \widehat{\mu}(Y_i) := \frac{1}{|Y_i|}\sum_{x\in Y_i} x, \quad \widehat{\mu}(Z_i) := \frac{1}{|Z_i|}\sum_{s\in Z_i} x.$$

We will also find it useful to express η_i^2 as follows. Here α is as defined in (8.7). We have:

$$\eta_i^2 = \alpha^2(\widehat{\mu}(Y_i) - \widehat{\mu}(Z_i))^2$$

The following calculations show that:

conditioned on X_i, $\widehat{\mu}(Z_i) - \mu$ and $\mu - \frac{|X_i|/\sigma^2}{|X_i|/\sigma^2+1/\ell^2}\widehat{\mu}(X_i)$ are independent Gaussian random variables[1]:

$$\begin{aligned}
&p(\widehat{\mu}(Z_i) - \mu, \mu | X_i) \propto p(\widehat{\mu}(Z_i) - \mu, \mu, X_i) \\
=&p(\widehat{\mu}(Z_i) - \mu, X_i|\mu)p(\mu) = p(\widehat{\mu}(Z_i) - \mu|\mu)p(X_i|\mu)p(\mu) \\
\propto& \exp\left(-\frac{1}{2}\frac{|Z_i|}{\sigma^2}(\widehat{\mu}(Z_i) - \mu)^2\right)\exp\left(-\frac{1}{2\sigma^2}\sum_{x\in X_i}(x-\mu)^2\right)\exp\left(-\frac{1}{2\ell^2}\mu^2\right)
\end{aligned}$$

[1]This is akin to the observation that given $u, v \sim \mathcal{N}(0,1)$, then $u - v$ and $u + v$ are independent.

$$\propto \underbrace{\exp\left(-\frac{1}{2}\frac{|Z_i|}{\sigma^2}(\widehat{\mu}(Z_i)-\mu)^2\right)}_{\propto p(\widehat{\mu}(Z_i)-\mu|X_i)} \underbrace{\exp\left(-\frac{1}{2}\left(\frac{|X_i|}{\sigma^2}+\frac{1}{\ell^2}\right)\left(\mu-\frac{|X_i|/\sigma^2}{|X_i|/\sigma^2+1/\ell^2}\widehat{\mu}(X_i)\right)^2\right)}_{\propto p\left(\mu-\frac{|X_i|/\sigma^2}{|X_i|/\sigma^2+1/\ell^2}\widehat{\mu}(X_i)|X_i\right)}$$

Thus conditioning on X_i, we can write

$$\begin{pmatrix}\widehat{\mu}(Z_i)-\mu \\ \mu-\frac{|X_i|/\sigma^2}{|X_i|/\sigma^2+1/\ell^2}\widehat{\mu}(X_i)\end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix}0\\0\end{pmatrix}, \begin{pmatrix}\frac{\sigma^2}{|Z_i|} & 0 \\ 0 & \frac{1}{|X_i|/\sigma^2+1/\ell^2}\end{pmatrix}\right).$$

which leads us to

$$\widehat{\mu}(Z_i)-\frac{|X_i|/\sigma^2}{|X_i|/\sigma^2+1/\ell^2}\widehat{\mu}(X_i)\Big| X_i \sim \mathcal{N}\left(0, \underbrace{\frac{\sigma^2}{|Z_i|}+\frac{1}{|X_i|/\sigma^2+1/\ell^2}}_{=:\tilde{\sigma}_\ell^2}\right) \tag{F.8}$$

Next, we will rewrite the squared difference in η_i^2 as follows:

$$\begin{aligned}\frac{\eta_i^2}{\alpha^2} =&(\widehat{\mu}(Y_i)-\widehat{\mu}(Z_i))^2 \\ =&\left(\underbrace{\widehat{\mu}(Z_i)-\frac{|X_i|/\sigma^2}{|X_i|/\sigma^2+1/\ell^2}\widehat{\mu}(X_i)}_{=\tilde{\sigma}_\ell e}+\underbrace{\left(\frac{|X_i|/\sigma^2}{|X_i|/\sigma^2+1/\ell^2}\widehat{\mu}(X_i)-\widehat{\mu}(Y_i)\right)}_{=:\phi(X_i,f_i)}\right)^2.\end{aligned}$$

Here, we observe that the first part of the RHS above is equal to $\tilde{\sigma}_\ell$, where e is a normal noise $e|X_i \sim \mathcal{N}(0,1)$ and $\tilde{\sigma}_\ell$ is as defined in (F.8). For brevity, we will denote the second part of the RHS as $\phi(X_i, f_i)$, which intuitively characterizes the difference between X_i and Y_i. Importantly, $\phi(X_i, f_i) = 0$ when f_i is chosen to be (F.7).

Using e and ϕ, we can rewrite (F.6) using conditional expectation:

$$\begin{aligned}
&\mathbb{E}_{X_i,Z_i}\left[\frac{1}{\frac{|Z_i'|}{\sigma^2+\eta_i^2}+\frac{|X_i|+|Z_i|}{\sigma^2}+\frac{1}{\ell^2}}\right] = \mathbb{E}_{X_i}\left[\mathbb{E}_{Z_i|X_i}\left[\frac{1}{\frac{|Z_i'|}{\sigma^2+\eta_i^2}+\frac{|X_i|+|Z_i|}{\sigma^2}+\frac{1}{\ell^2}}\right]\right] \\
&= \mathbb{E}_{X_i}\left[\mathbb{E}_{e|X_i}\left[\frac{1}{\frac{|Z_i'|}{\sigma^2+\alpha^2(\tilde{\sigma}_\ell e+\phi(X_i,f_i))^2}+\frac{|X_i|+|Z_i|}{\sigma^2}+\frac{1}{\ell^2}}\right]\right] \\
&= \mathbb{E}_{X_i}\left[\int_{-\infty}^{\infty} \underbrace{\frac{1}{\frac{|Z_i'|}{\sigma^2+\alpha^2\tilde{\sigma}_\ell^2(e+\phi(X_i,f_i)/\tilde{\sigma}_\ell)^2}+\frac{|X_i|+|Z_i|}{\sigma^2}+\frac{1}{\ell^2}}}_{=:F_1(e+\phi(X_i,f_i)/\tilde{\sigma}_\ell)} \underbrace{\frac{1}{\sqrt{2\pi}}\exp\left(-\frac{e^2}{2}\right)}_{=:F_2(e)} de\right],
\end{aligned} \tag{F.9}$$

where we use the fact that $e|X_i \sim \mathcal{N}(0,1)$ in the last step. To proceed, we will consider the inner expectation in the RHS above. For any fixed X_i, $F_1(\cdot)$ (as marked on the RHS) is an even function that monotonically increases on $[0,\infty)$ bounded by $\frac{\sigma}{|X_i|+|Z_i|}$ and $F_2(\cdot)$ (as marked on the RHS) is an even function that monotonically decreases on $[0,\infty)$. That means, for any $a \in \mathbb{R}$,

$$\int_{-\infty}^{\infty} F_1(e-a)F_2(e)de \le \int_{-\infty}^{\infty} \frac{\sigma}{|X_i|+|Z_i|}F_2(e)de = \frac{\sigma}{|X_i|+|Z_i|} < \infty.$$

By a corollary of the Hardy-Littlewood inequality in Lemma F.7.2, we have

$$\int_{-\infty}^{\infty} F_1(e+\phi(X_i,f_i)/\tilde{\sigma}_\ell)F_2(e)de \ge \int_{-\infty}^{\infty} F_1(e)F_2(e)de, \tag{F.10}$$

the equality is achieved when $\phi(X_i,f_i)/\tilde{\sigma}_\ell = 0$. In particular, the equality holds when f_i is chosen as specified in (F.7).

Now, to complete Step 1, we combine (F.6), (F.9) and (F.10) to obtain

$$\begin{aligned}
&\inf_{h_i\in\mathcal{H}} \mathbb{E}_{\mu\sim\Lambda_\ell}\Big[\mathbb{E}\Big[(h_i(X_i,Y_i,A_i)-\mu)^2|\mu\Big]\Big] = \mathbb{E}_{X_i}\left[\int_{-\infty}^{\infty} F_1(e+\phi(X_i,f_i)/\tilde{\sigma}_\ell)F_2(e)de\right] \\
&\ge \mathbb{E}_{X_i}\left[\int_{-\infty}^{\infty} F_1(e)F_2(e)de\right] = \int_{-\infty}^{\infty} F_1(e)F_2(e)de,
\end{aligned} \tag{F.11}$$

where the last step is because conditioning on each realization of X_i, the term inside the expectation is a constant. Using (F.11), we can write the Bayes risk R_ℓ under any prior Λ_ℓ as:

$$R_\ell := \inf_{f_i \in \mathcal{A}, h_i \in \mathcal{H}} \mathbb{E}_{\mu\sim\Lambda_\ell}\Big[\mathbb{E}\Big[(h_i(X_i, Y_i, A_i) - \mu)^2 \big| \mu\Big]\Big] = \int_{-\infty}^{\infty} F_1(e)F_2(e)de$$
$$= \mathbb{E}_{e\sim\mathcal{N}(0,1)}\left[\frac{1}{\frac{|Z_i'|}{\sigma^2+\alpha^2\tilde{\sigma}_\ell^2 e^2} + \frac{|X_i|+|Z_i|}{\sigma^2} + \frac{1}{\ell^2}}\right]$$

Because the term inside the expectation is bounded by $\frac{\sigma^2}{|X_i|+|Z_i|}$ and $\lim_{\ell\to\infty}\tilde{\sigma}_\ell^2 = \frac{\sigma^2}{|Z_i|} + \frac{\sigma^2}{|X_i|}$, we can use dominated convergence theorem to show that:

$$R_\infty := \lim_{\ell\to\infty} R_\ell = \mathbb{E}_{e\sim\mathcal{N}(0,1)}\left[\frac{1}{\frac{|Z_i'|}{\sigma^2+\alpha^2\left(\frac{\sigma^2}{|Z_i|}+\frac{\sigma^2}{|X_i|}\right)e^2} + \frac{|X_i|+|Z_i|}{\sigma^2}}\right] \tag{F.12}$$

Step 2: Maximum risk of $(f_i^\star, h_i^\star)$: Next, we will compute the maximum risk of the $(f_i^\star, h_i^\star)$ (see (8.8)) and show that it is equal to the RHS of (F.12). First note that we can write,

$$\begin{pmatrix}\widehat{\mu}(X_i) - \mu\\ \widehat{\mu}(Z_i) - \mu\end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix}0\\0\end{pmatrix}, \begin{pmatrix}\frac{\sigma^2}{|X_i|} & 0\\ 0 & \frac{\sigma^2}{|Z_i|}\end{pmatrix}\right).$$

By a linear transformation of this Gaussian vector, we obtain

$$\begin{pmatrix}\frac{|X_i|}{\sigma^2}(\widehat{\mu}(X_i) - \mu) + \frac{|Z_i|}{\sigma^2}(\widehat{\mu}(Z_i) - \mu)\\ \widehat{\mu}(X_i) - \widehat{\mu}(Z_i)\end{pmatrix} = \begin{pmatrix}\frac{|X_i|}{\sigma^2} & \frac{|Z_i|}{\sigma^2}\\ 1 & -1\end{pmatrix}\begin{pmatrix}\widehat{\mu}(X_i) - \mu\\ \widehat{\mu}(Z_i) - \mu\end{pmatrix}$$
$$\sim\mathcal{N}\left(\begin{pmatrix}0\\0\end{pmatrix}, \begin{pmatrix}\frac{|X_i|+|Z_i|}{\sigma^2} & 0\\ 0 & \frac{\sigma^2}{|X_i|} + \frac{\sigma^2}{|Z_i|}\end{pmatrix}\right),$$

which means $\frac{|X_i|}{\sigma^2}(\widehat{\mu}(X_i) - \mu) + \frac{|Z_i|}{\sigma^2}(\widehat{\mu}(Z_i) - \mu)$ and $\frac{\eta_i}{\alpha} = \widehat{\mu}(X_i) - \widehat{\mu}(Z_i)$ are independent Gaussian random variables. Therefore, the the maximum risk of $(f_i^\star, h_i^\star)$

is:

$$
\begin{aligned}
&\sup_{\mu} \mathbb{E}\Big[(h_i^\star(X_i, Y_i, A_i) - \mu)^2 | \mu\Big] \\
=&\sup_{\mu} \mathbb{E}_{\eta_i}\left[\mathbb{E}\left[\left(\frac{\frac{\sum_{x \in Z_i'} x}{\sigma^2+\eta_i^2} + \frac{|X_i|}{\sigma^2}\widehat{\mu}(X_i) + \frac{|Z_i|}{\sigma^2}\widehat{\mu}(Z_i)}{\frac{|Z_i'|}{\sigma^2+\eta_i^2} + \frac{|X_i|+|Z_i|}{\sigma^2}} - \mu\right)^2 \middle| \eta_i\right]\right] \\
=&\sup_{\mu} \mathbb{E}_{\eta_i}\left[\mathbb{E}\left[\left(\frac{\frac{\sum_{x \in Z_i'} (x-\mu)}{\sigma^2+\eta_i^2} + \frac{|X_i|}{\sigma^2}(\widehat{\mu}(X_i) - \mu) + \frac{|Z_i|}{\sigma^2}(\widehat{\mu}(Z_i) - \mu)}{\frac{|Z_i'|}{\sigma^2+\eta_i^2} + \frac{|X_i|+|Z_i|}{\sigma^2}}\right)^2 \middle| \eta_i\right]\right] \\
=&\sup_{\mu} \mathbb{E}_{\eta_i}\left[\frac{\mathbb{E}\left[\left(\frac{\sum_{x \in Z_i'} (x-\mu)}{\sigma^2+\eta_i^2} + \frac{|X_i|}{\sigma^2}(\widehat{\mu}(X_i) - \mu) + \frac{|Z_i|}{\sigma^2}(\widehat{\mu}(Z_i) - \mu)\right)^2 \middle| \eta_i\right]}{\left(\frac{|Z_i'|}{\sigma^2+\eta_i^2} + \frac{|X_i|+|Z_i|}{\sigma^2}\right)^2}\right] \\
=&\sup_{\mu} \mathbb{E}_{\eta_i}\left[\frac{1}{\left(\frac{|Z_i'|}{\sigma^2+\eta_i^2} + \frac{|X_i|+|Z_i|}{\sigma^2}\right)^2}\left(\frac{|Z_i'|\,(\sigma^2+\eta_i^2)}{(\sigma^2+\eta_i^2)^2} + \frac{|X_i|+|Z_i|}{\sigma^2}\right)\right] \\
=&\mathbb{E}_{\eta_i}\left[\frac{1}{\frac{|Z_i'|}{\sigma^2+\eta_i^2} + \frac{|X_i|+|Z_i|}{\sigma^2}}\right] = \mathbb{E}\left[\frac{1}{\frac{|Z_i'|}{\sigma^2+\alpha^2\left(\widehat{\mu}(Z_i)-\widehat{\mu}(X_i)\right)^2} + \frac{|X_i|+|Z_i|}{\sigma^2}}\right]
\end{aligned}
$$

Because $\widehat{\mu}(Z_i) - \widehat{\mu}(X_i) \sim \mathcal{N}\Big(0, \frac{\sigma^2}{|X_i|} + \frac{\sigma^2}{|Z_i|}\Big)$, we can further write the maximum risk as:

$$
\sup_{\mu} \mathbb{E}\Big[(h_i^\star(X_i, Y_i, A_i) - \mu)^2 | \mu\Big] = \mathbb{E}_{e \sim \mathcal{N}(0,1)}\left[\frac{1}{\frac{|Z_i'|}{\sigma^2+\alpha^2\left(\frac{\sigma^2}{|Z_i|}+\frac{\sigma^2}{|X_i|}\right)e^2} + \frac{|X_i|+|Z_i|}{\sigma^2}}\right] = R_\infty
$$

Here, we have observed that the final expression in the above equation is exactly the same as the Bayes' risk in the limit in (F.12) from Step 1.

Step 3: Minimax optimality of $(f_i^\star, h_i^\star)$: As the maximum is larger than the average, we can write, for any prior Λ_ℓ, and any $(f_i, h_i) \in \mathcal{F} \times \mathcal{H}$,

$$\sup_\mu \mathbb{E}\Big[(h_i(X_i, f_i(X_i), A_i) - \mu)^2 | \mu\Big] \geq \mathbb{E}_{\Lambda_\ell}\Big[\mathbb{E}\Big[(h_i(X_i, f_i(X_i), A_i) - \mu)^2 | \mu\Big]\Big] \geq R_\ell.$$

As this is true for all ℓ, by taking the limit we have, for all $(f_i, h_i) \in \mathcal{F} \times \mathcal{H}$,

$$\sup_\mu \mathbb{E}\Big[(h_i(X_i, f_i(X_i), A_i) - \mu)^2 | \mu\Big] \geq R_\infty = \sup_\mu \mathbb{E}\Big[(h_i^\star(X_i, f_i^\star(X_i), A_i) - \mu)^2 | \mu\Big].$$

That is, the recommended $(f_i^\star, h_i^\star)$ has a smaller maximum risk than all other $(f_i, h_i) \in \mathcal{F} \times \mathcal{H}$. This establishes that for any n_i,

$$p_i(M_{\text{C3D}}, ((n_i, f_i^\star, h_i^\star), s_{-i}^\star)) = \inf_{f_i \in \mathcal{A}} \inf_{h_i \in \mathcal{H}} p_i(M_{\text{C3D}}, ((n_i, f_i, h_i), s_{-i}^\star)).$$

Proof of Lemma F.1.3

In the previous section, we showed that for any n_i, the optimal (f_i, h_i) were $(f_i^\star, h_i^\star)$ as given in (8.8). Now, we show that for the given $(f_i^\star, h_i^\star)$, the optimal number of samples is $n_i^\star = \sigma/\sqrt{cm}$. For this, we will show that p_i is a convex function of n_i and then show that its gradient is 0 at $n_i^\star$.

First, noting that

$$\widehat{\mu}(Z_i) - \widehat{\mu}(X_i) \sim \mathcal{N}\left(0, \frac{\sigma^2}{|X_i|} + \frac{\sigma^2}{|Z_i|}\right),$$

we can rewrite the penalty term as:

$$p(n_i) := p_i\Big(M_{\text{C3D}}, ((n_i, f_i^\star, h_i^\star), s_{-i}^\star)\Big) = \mathbb{E}\left[\frac{1}{\frac{|Z_i'|}{\sigma^2 + \alpha^2\left(\widehat{\mu}(Z_i) - \widehat{\mu}(X_i)\right)^2} + \frac{|X_i| + |Z_i|}{\sigma^2}}\right] + cn_i$$

$$
\begin{aligned}
&= \mathbb{E}_{x\sim\mathcal{N}(0,1)}\left[\frac{1}{\frac{|Z_i'|}{\sigma^2+\alpha^2\left(\frac{\sigma^2}{|X_i|}+\frac{\sigma^2}{|Z_i|}\right)x^2}+\frac{|X_i|+|Z_i|}{\sigma^2}}\right]+cn_i \\
&= \mathbb{E}_{x\sim\mathcal{N}(0,1)}\Bigg[\underbrace{\frac{1}{\frac{(m-2)n_i^\star}{\sigma^2+\alpha^2\left(\frac{\sigma^2}{n_i}+\frac{\sigma^2}{n_i^\star}\right)x^2}+\frac{n_i+n_i^\star}{\sigma^2}}}_{=:l(n_i,x;\alpha)}\Bigg]+cn_i
\end{aligned}
\tag{F.13}
$$

Convexity of penalty function: To show that $p(n_i)$ is convex in n_i, let us consider $l(n_i, x; \alpha)$. Fixing α and x, we have

$$
\frac{\partial}{\partial n_i}l(n_i,x;\alpha) = -\sigma^2\frac{1+\frac{(m-2)n_i^\star}{\left(1+\alpha^2\left(\frac{1}{n_i}+\frac{1}{n_i^\star}\right)x^2\right)^2}\frac{\alpha^2x^2}{n_i^2}}{\left(\frac{(m-2)n_i^\star}{1+\alpha^2\left(\frac{1}{n_i}+\frac{1}{n_i^\star}\right)x^2}+n_i+n_i^\star\right)^2} = -\sigma^2\frac{1+\frac{(m-2)n_i^\star\alpha^2x^2}{\left(n_i+\alpha^2\left(1+\frac{n_i}{n_i^\star}\right)x^2\right)^2}}{\left(\frac{(m-2)n_i^\star}{1+\alpha^2\left(\frac{1}{n_i}+\frac{1}{n_i^\star}\right)x^2}+n_i+n_i^\star\right)^2}
\tag{F.14}
$$

As $\frac{\partial}{\partial n_i}l(n_i, x; \alpha)$ is an increasing function of n_i, we have that $l(n_i, x; \alpha)$ is a convex function in n_i. As expectation preserves convexity (see Lemma F.7.3), $p(n_i)$ is a convex function.

Penalty is minimized when $n_i = n_i^\star$. Lemma F.7.6 provides an expression for the derivative of $p(n_i)$ (obtained purely via algebraic manipulations). Using this, we have

$$
\begin{aligned}
p'(n_i^\star) = &-\frac{\sigma^2}{64\frac{\alpha^2}{m-2}\frac{\alpha}{\sqrt{mn_i^\star}}mn_i^\star}\Bigg(\frac{4\alpha}{\sqrt{mn_i^\star}}\left(\frac{4\alpha^2m}{(m-2)n_i^\star}-1\right) \\
&-\exp\left(\frac{mn_i^\star}{8\alpha^2}\right)\left(\frac{4\alpha^2}{mn_i^\star}(m+1)-1\right)\sqrt{2\pi}\,\mathrm{Erfc}\left(\frac{1}{2\sqrt{2}\sqrt{\frac{\alpha^2}{mn_i^\star}}}\right)\Bigg) \\
&+c \qquad\qquad \text{(By Lemma F.7.6)}
\end{aligned}
$$

$$
\begin{aligned}
&= -\frac{\sigma^2}{64\frac{\alpha^2}{m-2}\frac{\alpha}{\sqrt{mn_i^\star}}mn_i^\star}\left(\frac{4\alpha}{\sqrt{mn_i^\star}}\left(\frac{4\alpha^2(m-4)}{(m-2)n_i^\star}-1\right)\right.\\
&\quad\left.-\exp\left(\frac{mn_i^\star}{8\alpha^2}\right)\left(\frac{4\alpha^2}{mn_i^\star}(m+1)-1\right)\sqrt{2\pi}\,\mathrm{Erfc}\left(\frac{1}{2\sqrt{2}\sqrt{\frac{\alpha^2}{mn_i^\star}}}\right)\right)\\
&= G(\alpha) = 0.
\end{aligned}
$$

Here, the second step uses the fact that $n_i^\star = \frac{\sigma}{\sqrt{cm}}$. Finally, we have observed that the expression is equal to $G(\alpha)$ as defined in (8.7) which is 0 by our choice of α. Since $p'(n_i^\star) = 0$ and $p(\cdot)$ is convex, we can conclude that $p(n_i)$ is minimized when $n_i = n_i^\star$. Therefore,

$$p_i(M_{\text{C3D}}, ((n_i^\star, f_i^\star, h_i^\star), s_{-i}^\star)) \leq p_i(M_{\text{C3D}}, ((n_i, f_i^\star, h_i^\star), s_{-i}^\star)).$$

Algorithm 10 is individually rational

As outlined in the main text, the NIC property implies IR since 'working on her own' is a valid strategy in the mechanism. Precisely, if an agent collects any number of points n_i, chooses not to submit anything $f_i(\cdot) = \varnothing$, and then uses the sample average of the points she collected $h_i(X_i, \varnothing, A_i) = |X_i|^{-1}\sum_{x\in X_i} x$, then $(n_i, f_i, h_i) \in \mathcal{S}$.

Below, we will prove this more formally and also show that the agent's penalty is strictly smaller when participating. For any fixed n_i, without participating in the mechanism, the smallest penalty the agent can achieve is by using empirical mean estimation and the penalty is:

$$\frac{\sigma^2}{n_i} + cn_i$$

When participating, the agent gets an additional $n_i^\star$ number of clean data along with some noisy data, provided that all other agents are following $s_{-i}^\star$. By using

the empirical mean over the clean data, the penalty is:

$$\frac{\sigma^2}{n_i + n_i^\star} + cn_i < \frac{\sigma^2}{n_i} + cn_i$$

Now, since the weighted average estimator in $s_i^\star$ is minimax optimal, the agent gets even smaller maximum risk and hence smaller penalty. In other words, for any n_i,

$$p_i(M_{\text{C3D}}, s^\star) \le p_i(M_{\text{C3D}}, ((n_i, f_i^\star, h_i^\star), s_{-i}^\star)) \le \frac{\sigma^2}{n_i + n_i^\star} + cn_i < \frac{\sigma^2}{n_i} + cn_i$$

By minimizing the RHS with respect to n_i, we get $p_i(M_{\text{C3D}}, s^\star) < p_{\min}^{\text{IR}}$. Thus Algorithm 10 is IR.

Algorithm 10 is approximately efficient

In this section, we will bound the penalty ratio PR for M_{C3D} at the strategy profiles $s_i^\star$.

First, noting that $G(\alpha) = 0$ (see (8.7)), we can rearrange the terms in the equation to obtain:

$$\exp\left(\frac{mn_i^\star}{8\alpha^2}\right) \operatorname{Erfc}\left(\frac{1}{2\sqrt{2}\sqrt{\frac{\alpha^2}{mn_i^\star}}}\right) = \frac{1}{\sqrt{2\pi}} \frac{\frac{4\alpha}{\sqrt{mn_i^\star}}\left(\frac{4\alpha^2(m-4)}{(m-2)n_i^\star} - 1\right)}{\frac{4\alpha^2}{mn_i^\star}(m+1) - 1} \tag{F.15}$$

Next, we will use the expression for $p(n_i) = p_i(M_{\text{C3D}}, (s_{-i}^\star, (n_i, f_i^\star, h_i^\star)))$ in Lemma F.7.6 and the equation in (F.15) to simplify $p(n_i^\star)$ as follows:

$$p(n_i^\star) = \frac{\sqrt{\frac{\alpha^2}{mn_i^\star}}\sigma^2\left(2m\sqrt{2\pi}\sqrt{\frac{\alpha^2}{mn_i^\star}} - \exp\left(\frac{mn_i^\star}{8\alpha^2}\right)(m-2)\pi \operatorname{Erfc}\left(\frac{1}{2\sqrt{2}\sqrt{\frac{\alpha^2}{mn_i^\star}}}\right)\right)}{4\sqrt{2\pi}\alpha^2} + cn_i^\star$$

(By Lemma F.7.6)

$$
\begin{aligned}
=&\frac{\sqrt{\frac{\alpha^2}{mn_i^\star}}\sigma^2\left(2m\sqrt{2\pi}\sqrt{\frac{\alpha^2}{mn_i^\star}}-(m-2)\pi\frac{1}{\sqrt{2\pi}}\frac{\frac{4\alpha}{\sqrt{mn_i^\star}}\left(\frac{4\alpha^2(m-4)}{(m-2)n_i^\star}-1\right)}{\frac{4\alpha^2}{mn_i^\star}(m+1)-1}\right)}{4\sqrt{2\pi}\alpha^2}+cn_i^\star \quad \text{(By (F.15))}\\
=&\frac{\sigma^2\left(m-(m-2)\frac{\frac{4\alpha^2(m-4)}{(m-2)n_i^\star}-1}{\frac{4\alpha^2}{mn_i^\star}(m+1)-1}\right)}{2mn_i^\star}+cn_i^\star\\
=&\frac{\sigma^2}{2mn_i^\star}\frac{\frac{4\alpha^2}{n_i^\star}(m+1)-m-\frac{4\alpha^2}{n_i^\star}(m-4)+(m-2)}{\frac{4\alpha^2}{n_i^\star}\frac{m+1}{m}-1}+cn_i^\star\\
=&\frac{\sigma^2}{2mn_i^\star}\frac{\frac{20\alpha^2}{n_i^\star}-2}{\frac{4\alpha^2}{n_i^\star}\frac{m+1}{m}-1}+cn_i^\star=\frac{\sigma^2}{mn_i^\star}\frac{\frac{10\alpha^2}{n_i^\star}-1}{\frac{4\alpha^2}{n_i^\star}\frac{m+1}{m}-1}+cn_i^\star\\
=&\sigma\sqrt{\frac{c}{m}}\left(\frac{\frac{10\alpha^2}{n_i^\star}-1}{\frac{4\alpha^2}{n_i^\star}\frac{m+1}{m}-1}+1\right)
\end{aligned}
$$

From our conclusion in Section F.1, we have $\alpha^2 > \frac{\sigma}{\sqrt{cm}} = n_i^\star$, i.e. $\frac{\alpha^2}{n_i^\star} > 1$. Therefore, we have:

$$
\begin{aligned}
\mathrm{PR}(M_{\text{C3D}}, s^\star) =&\frac{mp(n_i^\star)}{2\sigma\sqrt{cm}}=\frac{1}{2}\left(\frac{\frac{10\alpha^2}{n_i^\star}-1}{\frac{4\alpha^2}{n_i^\star}\frac{m+1}{m}-1}+1\right)\\
<&\frac{1}{2}\left(\frac{\frac{10\alpha^2}{n_i^\star}-1+\frac{10\alpha^2}{n_i^\star}\frac{1}{m}+\left(\frac{2\alpha^2}{n_i^\star}\frac{m+1}{m}-2\right)}{\frac{4\alpha^2}{n_i^\star}\frac{m+1}{m}-1}+1\right)=2.
\end{aligned}
$$

F.2 Proof of Theorem 8.4.1

We will use M_{PCS} to denote the mechanism in Section 8.4, as it *pools* the datsets, but *checks* for the *size* of the dataset submitted by each agent. For clarity, we have stated M_{PCS} algorithmically in Algorithm 18. We will also re-state the recommended strategies $s_i^\star = \{(n_i^\star, f_i^\star, h_i^\star)\}_i$ below:

$$
n_i^\star = \frac{\sigma}{\sqrt{cm}}, \qquad f_i^\star = \mathbf{I}, \qquad h_i^\star(X_i, Y_i, A_i) = \frac{1}{|X_i \cup A_i|}\sum_{u \in X_i \cup A_i} u \tag{F.16}
$$

Algorithm 18 M_{PCS}

Mechanism designer publishes:
The allocation space $\mathcal{A} = \bigcup_{n\geq 0} \mathbb{R}^n$, and the procedure in lines 6 –11.
Each agent i**:**
Choose strategy $s_i = (n_i, f_i, h_i)$.
Sample n_i points $X_i = \{x_{i,j}\}_{j=1}^{n_i}$ and submit $Y_i = f_i(X_i)$ to the mechanism.
Mechanism:
For each agent $i \in [m]$: # can be done simultaneously for all agents
$A_i \leftarrow \bigcup_{j\neq i} Y_j$ if $|Y_i| \geq \sigma/\sqrt{cm}$, $A_i \leftarrow \varnothing$ otherwise.
Return A_i to each agent.
Each agent i**:**
Compute estimate $h_i(X_i, Y_i, A_i)$.

Throughout this section, $s_i^\star$ will refer to (F.16) (and not (8.8)).

We will first prove that $s_i^\star$ is a Nash equilibrium. Because the sample mean achieves minimax error for Normal mean estimation Lehmann and Casella (2006), we immediately have, for all $(n_i, f_i, h_i) \in \mathcal{S}$.

$$p_i(M_{\text{PCS}}, ((n_i, f_i, h_i^\star), s_{-i}^\star)) \leq p_i(M_{\text{PCS}}, ((n_i, f_i, h_i), s_{-i}^\star)).$$

Because the agent can only submit the raw dataset or a subset, and the agent's allocation only depends on the size of the dataset, the size of the dataset she receives can always be maximized by submittng the whole data set she collects, i.e. chooses $f_i = \mathbf{I}$. Therefore, we have for all $(n_i, f_i, h_i) \in \mathcal{S}$,

$$p_i(M_{\text{PCS}}, ((n_i, f_i^\star, h_i^\star), s_{-i}^\star)) \leq p_i(M_{\text{PCS}}, ((n_i, f_i, h_i^\star), s_{-i}^\star)) \leq p_i(M_{\text{PCS}}, ((n_i, f_i, h_i), s_{-i}^\star)).$$

Finally, we can use the fact that the maximum risk of the sample mean estimator using n points is σ^2/n to show that the penalty is minimized when $n_i = n_i^\star = \sigma/\sqrt{cm}$. In particular, we have that if $n_i < \sigma/\sqrt{cm}$,

$$p_i(M_{\text{PCS}}, ((n_i, f_i^\star, h_i^\star), s_{-i}^\star)) = \frac{\sigma^2}{n_i} + cn_i > 2\sigma\sqrt{c}.$$

And if $n_i \geq \sigma/\sqrt{cm}$,

$$p_i(M_{\text{PCS}}, ((n_i, f_i^\star, h_i^\star), s_{-i}^\star)) = \frac{\sigma^2}{n_i + (m-1)\sigma/\sqrt{cm}} + cn_i \geq 2\sigma\sqrt{\frac{c}{m}}$$

Because $2\sigma\sqrt{c} \geq 2\sigma\sqrt{c/m}$, $p_i(M_{\text{PCS}}, ((n_i, f_i^\star, h_i^\star), s_{-i}^\star))$ is minimized when $n_i = \sigma/\sqrt{cm}$. We thus conclude that $s^\star$ is a Nash equilibrium. That is, for all $(n_i, f_i, h_i) \in \mathbb{N} \times \mathcal{F} \times \mathcal{H}$

$$p_i(M_{\text{PCS}}, s^\star) \leq p_i(M_{\text{PCS}}, ((n_i, f_i, h_i), s_{-i}^\star)).$$

Next, the IR and efficiency properties follow trivially from the fact that $p_i(M_{\text{PCS}}, s^\star) = 2\sigma\sqrt{c/m}$ for each agent i. In particular, $p_i(M_{\text{PCS}}, s^\star) < p_{\min}^{\text{IR}}$ and $P(M_{\text{PCS}}, s^\star) = 2\sigma\sqrt{cm}$.

F.3 Proof of Theorem 8.4.2

We will use M_{CDED} to denote our mechanism in Section 8.4, as it *corrupts* the *deployed estimate* based on the *difference*. We have stated this mechanism formally in Algorithm 19. We will also re-state the recommended strategies $s_i^\star = \{(n_i^\star, f_i^\star)\}_i$ below:

$$n_i^\star = \frac{\sigma}{\sqrt{cm}}, \qquad f_i^\star = \mathbf{I}. \tag{F.17}$$

Throughout this section, $s_i^\star$ will refer to (F.17) (and not (8.8) or (F.16)).

We will now present the proof of Theorem 8.4.2. First, in Section F.3, we show that $s^\star$ is a Nash equilibrium of M_{CDED} as the Nash incentive compatibility result. Then, in Section F.3, we show individual rationality at $s_i^\star$. In Section F.3, we conclude by showing that M_{CDED} is approximately efficient by showing that its social penalty at most a $(1+\epsilon)$ factor of the global minimum.

Algorithm 19 is Nash incentive compatible

Step 1. We will first show that fixing any n_i, the best strategy is to submit the raw data, i.e. for all $(n_i, f_i) \in \mathbb{N} \times \mathcal{F}$.

$$p_i(M_{\text{CDED}}, ((n_i, f_i^\star), s_{-i}^\star)) \leq p_i(M_{\text{CDED}}, ((n_i, f_i), s_{-i}^\star)). \tag{F.18}$$

Let $e_{z,i} = \epsilon_{z,i}/\eta_i$, where η_i, and $\epsilon_{z,i}$ are as given in lines 9 and 10 respectively. We have that $e_{z,i}$'s are i.i.d. standard Normal samples. Because the cost term cn_i is fixed when n_i is fixed, we only need to consider the risk term. We will first define,

$$\widehat{\mu}(X_i) := \frac{1}{|X_i|} \sum_{x \in X_i} x, \quad \widehat{\mu}(Y_i) := \frac{1}{|Y_i|} \sum_{x \in Y_i} x, \quad \widehat{\mu}(Y_{-i}) := \frac{1}{|Y_{-i}|} \sum_{x \in Y_{-i}} x. \tag{F.19}$$

Via some algebraic manipulations, we can express the maximum risk as:

$$\begin{aligned}
&\sup_\mu \mathbb{E}\left[\left(\frac{1}{|Y_i| + (m-1)n_i^\star}\left(\sum_{y \in Y_i}(y-\mu) + \sum_{z \in Y_{-i}}(z + e_{z,i}\eta_i - \mu)\right)\right)^2 \middle| \mu\right] \\
=&\frac{1}{\left(|Y_i| + (m-1)n_i^\star\right)^2} \sup_\mu \mathbb{E}\left[\left(\sum_{y \in Y_i}(y-\mu)\right)^2 + \left(\sum_{z \in Y_{-i}}(z + e_{z,i}\eta_i - \mu)\right)^2 \middle| \mu\right] \\
=&\frac{1}{\left(|Y_i| + (m-1)n_i^\star\right)^2} \sup_\mu \mathbb{E}\Big[\left(|Y_i|\left(\widehat{\mu}(Y_i) - \mu\right)\right)^2 + \left(\sum_{z \in Y_{-i}}(z-\mu)\right)^2 \\
&+ \left(\sum_{z \in Y_{-i}} e_{z,i}\eta_i\right)^2 \Big| \mu\Big] \\
=&\frac{1}{\left(|Y_i| + (m-1)n_i^\star\right)^2} \sup_\mu \mathbb{E}\Big[\left(|Y_i|\left(\widehat{\mu}(Y_i) - \mu\right)\right)^2 + (m-1)n_i^\star\beta_\epsilon^2(\widehat{\mu}(Y_i) - \widehat{\mu}(Y_{-i}))^{2k_\epsilon} \Big| \mu\Big] \\
&+ \frac{(m-1)n_i^\star\sigma^2}{\left(|Y_i| + (m-1)n_i^\star\right)^2}
\end{aligned}$$

Recall that β_ϵ also involves $|Y_i|$. Note that as we have fixed n_i and $s_{-i} = s_{-i}^\star$, the maximum risk depends only on $|Y_i|$ and $\widehat{\mu}(Y_i)$, that is, the agent's maximum risk and hence penalty only depends on the number of points she submitted, and their

average value. Hence, to find the optimal submission Y_i, we will first fix the size of the agent's submission $|Y_i|$ and optimize for the sample mean $\widehat{\mu}(Y_i)$ (step 1.1), and then we will optimize for $|Y_i|$ (step 1.2).

Step 1.1. Since the other agents have each collected $\sigma/\sqrt{cm} = n_i^\star$ points and submitted it truthfully, we have $\widehat{\mu}(Y_{-i}) \sim \mathcal{N}\Big(\mu, \frac{\sigma^2}{(m-1)n_i^\star}\Big)$. Via a binomial expansion , we can write,

$$\begin{aligned}
\mathbb{E}\Big[\big(\widehat{\mu}(Y_i) - \widehat{\mu}(Y_{-i})\big)^{2k_\epsilon}\Big] =& \mathbb{E}\Big[\big((\widehat{\mu}(Y_i) - \mu) - (\widehat{\mu}(Y_{-i}) - \mu)\big)^{2k_\epsilon}\Big] \\
=& \sum_{j=0}^{2k_\epsilon} (-1)^j \binom{2k_\epsilon}{j} \mathbb{E}\Big[(\widehat{\mu}(Y_i) - \mu)^j\Big] \mathbb{E}\Big[(\widehat{\mu}(Y_{-i}) - \mu)^{2k_\epsilon - j}\Big] \\
=& \sum_{j=0}^{k_\epsilon} \binom{2k_\epsilon}{2j} \mathbb{E}\Big[(\widehat{\mu}(Y_i) - \mu)^{2j}\Big] \mathbb{E}\Big[(\widehat{\mu}(Y_{-i}) - \mu)^{2k_\epsilon - 2j}\Big]
\end{aligned}$$

Thus the maximum risk can be written as:

$$\sup_{\mu} \mathbb{E}\left[\sum_{j=0}^{k_\epsilon} A_j (\widehat{\mu}(Y_i) - \mu)^{2j} \middle| \mu\right] \tag{F.20}$$

where $A_0, \ldots, A_{k_\epsilon}$ is a sequence of positive coefficients.

Similar to the proof of Theorem 8.3.1, we construct a lower bound on the maximum risk using a sequence of Bayesian risks. Let $\Lambda_\ell := \mathcal{N}(0, \ell^2)$, $\ell = 1, 2, \ldots$ be a sequence of prior for μ. For fixed ℓ, the posterior distribution is:

$$\begin{aligned}
p(\mu|X_i) \propto& p(X_i|\mu)p(\mu) \propto \exp\left(-\frac{1}{2\sigma^2} \sum_{x \in X_i} (x - \mu)^2\right) \exp\Big(-\frac{1}{2\ell^2}\mu^2\Big) \\
\propto& \exp\Big(-\frac{1}{2}\Big(\frac{n_i}{\sigma^2} + \frac{1}{\ell^2}\Big)\mu^2 + \frac{1}{2} 2 \frac{\sum_{x \in X_i} x}{\sigma^2} \mu\Big).
\end{aligned}$$

This means the posterior of μ given X_i is Gaussian with:

$$\mu|X_i \sim \mathcal{N}\left(\frac{n_i \widehat{\mu}(X_i)/\sigma^2}{n_i/\sigma^2 + 1/\ell^2}, \frac{1}{n_i/\sigma^2 + 1/\ell^2}\right) =: \mathcal{N}\Big(\mu_\ell, \sigma_\ell^2\Big).$$

Algorithm 19 M_{CDED}

Require: Approximation parameter $\epsilon > 0$ # to obtain a $1+\epsilon$ bound on PR.

1: **Mechanism designer publishes:** The procedure in lines 5 –11.
2: **Each agent** i**:**
3: Choose strategy $s_i = (n_i, f_i)$.
4: Sample n_i points $X_i = \{x_{i,j}\}_{j=1}^{n_i}$ and submit $Y_i = f_i(X_i)$ to the mechanism.
5: **Mechanism:**
6: $k_\epsilon \leftarrow \lceil \frac{1}{2\epsilon} \rceil, \quad \beta_\epsilon \leftarrow \sqrt{\frac{\left(\sum_{i=1}^m |Y_i|\right)^2 (m-1)^{k_\epsilon - 1}}{k_\epsilon (2k_\epsilon - 1)!! \sigma^{k_\epsilon} c^{\frac{k_\epsilon - 2}{2}} m^{3k_\epsilon/2}}}$
7: **For** each agent $i \in [m]$: # can be done simultaneously for all agents
8: $Y_{-i} \leftarrow \bigcup_{j \neq i} Y_j$.
9: $\eta_i^2 \leftarrow \beta_\epsilon^2 \left(\frac{1}{|Y_i|} \sum_{y \in Y_i} y - \frac{1}{|Y_{-i}|} \sum_{y \in Y_{-i}} y \right)^{2k_\epsilon}$.
10: $Z_i \leftarrow \{z + \epsilon_{z,i}, \text{ for all } z \in Y_{-i} \text{ where } \epsilon_{z,i} \sim \mathcal{N}(0, \eta_i^2)\}$
11: Deploy estimate $\left(\frac{1}{|Y_i \cup Z_i|} \sum_{u \in Y_i \cup Z_i} u \right)$ for agent i.

Therefore, the posterior risk is:

$$\begin{aligned}
\mathbb{E}\left[\sum_{j=0}^{k_\epsilon} A_j (\widehat{\mu}(Y_i) - \mu)^{2j} \,\middle|\, X_i \right] &= \mathbb{E}\left[\sum_{j=0}^{k_\epsilon} A_j ((\widehat{\mu}(Y_i) - \mu_\ell) - (\mu - \mu_\ell))^{2j} \,\middle|\, X_i \right] \\
&= \int_{-\infty}^{\infty} \underbrace{\sum_{j=0}^{k_\epsilon} A_j (e - (\widehat{\mu}(Y_i) - \mu_\ell))^{2j}}_{=:F_1(e - (\widehat{\mu}(Y_i) - \mu_\ell))} \underbrace{\frac{1}{\sigma_\ell \sqrt{2\pi}} \exp\left(-\frac{e^2}{2\sigma_\ell^2} \right)}_{=:F_2(e)} de
\end{aligned}$$

Because:

- $F_1(\cdot)$ is even function and increases on $[0, \infty)$;
- $F_2(\cdot)$ is even function and decreases on $[0, \infty$, and $\int_{\mathbb{R}} F_2(e) de < \infty$
- For any $a \in \mathbb{R}$, $\int_{\mathbb{R}} F_1(e - a) F_2(e) de < \infty$,

By the corollary of Hardy-Littlewood inequality in Lemma F.7.2,

$$\int_{\mathbb{R}} F_1(e - a) F_2(e) de \geq \int_{\mathbb{R}} F_1(e) F_2(e) de,$$

which means the posterior risk is minimized when $\widehat{\mu}(Y_i) = \mu_\ell$. In Lemma F.7.4, we have stated expressions for the expected value of the power of a normal random variable. Using this, we can write the Bayes risk as:

$$R_\ell := \mathbb{E}\left[\sum_{j=0}^{k_\epsilon} A_j \mathbb{E}\left[(\mu - \mu_\ell)^{2j} \middle| X_i\right]\right] = \sum_{j=0}^{k_\epsilon} A_j (2j-1)!!\, \sigma_\ell^{2j}$$

and the limit of Bayesian risk as $\ell \to \infty$ is

$$R_\infty := \lim_{\ell \to \infty} \sum_{j=0}^{k_\epsilon} A_j (2j-1)!!\, \frac{\sigma^{2j}}{n_i^j}$$

When $\widehat{\mu}(Y_i) = \widehat{\mu}(X_i)$, the maximum risk is:

$$\sup_\mu \mathbb{E}\left[\sum_{j=0}^{k_\epsilon} A_j (\widehat{\mu}(Y_i) - \mu)^{2j} \middle| \mu\right] = \sup_\mu \mathbb{E}\left[\sum_{j=0}^{k_\epsilon} A_j (\widehat{\mu}(X_i) - \mu)^{2j} \middle| \mu\right]$$
$$= \sum_{j=0}^{k_\epsilon} A_j (2j-1)!!\, \sigma^{2j} n_i^{-j} = R_\infty.$$

This means, fixing n_i and $|Y_i|$, agent i achieves minimax risk when choosing $\widehat{\mu}(Y_i) = \widehat{\mu}(X_i)$; as the maximum is larger than the average, this follows using a similar argument to Step 3 in Section F.1.

Step 1.2. Next, we will show that the best size of the submission is $|Y_i| = |X_i| = n_i$, assuming $\widehat{\mu}(Y_i) = \widehat{\mu}(X_i)$. For this, we will first use $n_i^\star$ to rewrite β_ϵ^2 as

$$\beta_\epsilon^2 = \frac{n_i^{\star k_\epsilon - 2}(m-1)^{k_\epsilon - 1}(|Y_i| + (m-1)n_i^\star)^2}{k_\epsilon (2k_\epsilon - 1)!!\, m^{k_\epsilon + 1}\sigma^{2k_\epsilon - 2}}.$$

Because

$$\widehat{\mu}(X_i) - \widehat{\mu}(Y_{-i}) \sim \mathcal{N}\left(0, \left(\frac{1}{n_i} + \frac{1}{(m-1)n_i^\star}\right)\sigma^2\right),$$

the risk term in the penalty can be rewritten and lower bounded as follows:

$$\begin{aligned}
&\frac{1}{\left(|Y_i|+(m-1)n_i^\star\right)^2}\left(|Y_i|^2\,\sigma^2/n_i+(m-1)n_i^\star\beta_\epsilon^2(2k_\epsilon-1)!!\left(\frac{1}{n_i}+\frac{1}{(m-1)n_i^\star}\right)^{k_\epsilon}\sigma^{2k_\epsilon}\right)\\
&+\frac{(m-1)n_i^\star\sigma^2}{\left(|Y_i|+(m-1)n_i^\star\right)^2}\\
=&\frac{|Y_i|^2\,\frac{\sigma^2}{n_i}+(m-1)n_i^\star\sigma^2}{\left(|Y_i|+(m-1)n_i^\star\right)^2}+\frac{n_i^{\star k_\epsilon-1}(m-1)^{k_\epsilon}}{k_\epsilon m^{k_\epsilon+1}}\left(\frac{1}{n_i}+\frac{1}{(m-1)n_i^\star}\right)^{k_\epsilon}\sigma^2\\
\geq&\frac{\sigma^2}{n_i+(m-1)n_i^\star}+\frac{n_i^{\star k_\epsilon-1}(m-1)^{k_\epsilon}}{k_\epsilon m^{k_\epsilon+1}}\left(\frac{1}{n_i}+\frac{1}{(m-1)n_i^\star}\right)^{k_\epsilon}\sigma^2.
\end{aligned}$$

Here, the last step follows from the fact that

$$\begin{aligned}
&\frac{|Y_i|^2\,\frac{\sigma^2}{n_i}+(m-1)n_i^\star\sigma^2}{\left(|Y_i|+(m-1)n_i^\star\right)^2}=\frac{|Y_i|^2\,\frac{\sigma^2}{n_i}+(m-1)n_i^\star\sigma^2}{n_i\frac{|Y_i|^2}{n_i}+2\,|Y_i|\,(m-1)n_i^\star+(m-1)^2n_i^{\star2}}\\
\geq&\frac{|Y_i|^2\,\frac{\sigma^2}{n_i}+(m-1)n_i^\star\sigma^2}{n_i\frac{|Y_i|^2}{n_i}+\left(n_i+\frac{|Y_i|^2}{n_i}\right)(m-1)n_i^\star+(m-1)^2n_i^{\star2}}=\frac{|Y_i|^2\,\frac{\sigma^2}{n_i}+(m-1)n_i^\star\sigma^2}{(n_i+(m-1)n_i^\star)\left(\frac{|Y_i|^2}{n_i}+(m-1)n_i^\star\right)}\\
=&\frac{\sigma^2}{n_i+(m-1)n_i^\star}.
\end{aligned}$$

Equality holds in this inequality if and only if $|Y_i|=n_i$.

In conclusion, fixing n_i, the agent can minimize her penalty by submitting n_i points with the same sample mean as the dataset X_i she collected. One way to achieve this is set $f_i=\mathbf{I}$. This completes the proof of (F.18).

Step 2: Our next step is to show that the agent's best strategy is to collect $n_i^\star$ data points. That is, we will show for all $n_i\in\mathbb{N}$.

$$p_i(M_{\text{CDED}},((n_i^\star,f_i^\star),s_{-i}^\star))\leq p_i(M_{\text{CDED}},((n_i,f_i^\star),s_{-i}^\star)). \tag{F.21}$$

In the following, we will use $p(n_i)$ as a shorthand for $p_i(M_{\text{CDED}},((n_i,f_i^\star),s_{-i}^\star))$. The

penalty can be rewritten as:

$$p(n_i) = \frac{\sigma^2}{n_i + (m-1)n_i^\star} + \frac{n_i^{\star k_\epsilon - 1}(m-1)^{k_\epsilon}}{k_\epsilon m^{k_\epsilon+1}}\left(\frac{1}{n_i} + \frac{1}{(m-1)n_i^\star}\right)^{k_\epsilon}\sigma^2 + cn_i$$

We need to show that $p_i(n_i)$ achieves minimum at $n_i = n_i^\star$. The derivative of $p_i(\cdot)$ is:

$$p'(n_i) = -\frac{\sigma^2}{(n_i + (m-1)n_i^\star)^2} + \frac{n_i^{\star k_\epsilon - 1}(m-1)^{k_\epsilon}}{m^{k_\epsilon+1}}\left(\frac{1}{n_i} + \frac{1}{(m-1)n_i^\star}\right)^{k_\epsilon-1}\sigma^2\left(-\frac{1}{n_i^2}\right) + c$$

Because $p'(n_i)$ increase in n_i, $p(n_i)$ is convex. Moreover, because

$$\begin{aligned} p'(n_i^\star) =& -\frac{\sigma^2}{m^2 n_i^{\star 2}} + \frac{n_i^{\star k_\epsilon - 1}(m-1)^{k_\epsilon}}{m^{k_\epsilon+1}}\left(\frac{1}{n_i^\star} + \frac{1}{(m-1)n_i^\star}\right)^{k_\epsilon-1}\sigma^2\left(-\frac{1}{n_i^{\star 2}}\right) + c \\ =& -\frac{\sigma^2}{m^2 n_i^{\star 2}} - \frac{(m-1)\sigma^2}{m^2 n_i^{\star 2}} + c = -\frac{\sigma^2}{m n_i^{\star 2}} + c = 0, \end{aligned}$$

we know $p(n_i)$ reaches minimum at $n_i = n_i^\star$. This concludes the proof for (F.21).

Algorithm 19 is individually rational

The penalty of an agent at the recommended strategies can be expressed as:

$$\begin{aligned} p_i(M_{\text{CDED}}, s_i^\star) =& p(n_i^\star) = \frac{\sigma^2}{m n_i^\star} + \frac{n_i^{\star k_\epsilon - 1}(m-1)^{k_\epsilon}}{k_\epsilon m^{k_\epsilon+1}}\left(\frac{1}{n_i^\star} + \frac{1}{(m-1)n_i^\star}\right)^{k_\epsilon}\sigma^2 + cn_i^\star \\ =& \frac{\sigma^2}{m n_i^\star} + \frac{n_i^{\star k_\epsilon - 1}(m-1)^{k_\epsilon}}{k_\epsilon m^{k_\epsilon+1}}\frac{m^{k_\epsilon}}{n_i^{\star k_\epsilon}(m-1)^{k_\epsilon}}\sigma^2 + cn_i^\star \\ =& \frac{\sigma^2}{m n_i^\star} + \frac{1}{k_\epsilon}\frac{\sigma^2}{m n_i^\star} + cn_i^\star = \left(2 + \frac{1}{k_\epsilon}\right)\frac{\sigma\sqrt{c}}{\sqrt{m}}. \end{aligned} \tag{F.22}$$

We have that M_{CDED} is IR when $m \geq 2$, via the following simple calculation:

$$\left(2 + \frac{1}{k_\epsilon}\right)\frac{\sigma\sqrt{c}}{\sqrt{m}} \leq \left(2 + \frac{1}{2}\right)\frac{\sigma\sqrt{c}}{\sqrt{2}} < 2\sigma\sqrt{c} = p_{\min}^{\text{IR}}$$

Algorithm 19 is approximately efficient

Using the expression for $p_i(M_{\text{CDED}}, s_i^\star)$ in (F.22), the penalty ratio can be bounded by:

$$\text{PR}(M_{\text{CDED}}, s^\star) = \frac{\left(2 + \frac{1}{k_\epsilon}\right)\sigma\sqrt{cm}}{2\sigma\sqrt{cm}} = 1 + \frac{1}{2k_\epsilon} \leq 1 + \epsilon.$$

F.4 Additional Materials for Section 8.4

Mechanism detail

See Algorithm 19.

Using a weighted average under the original strategy space from Section 8.2

In this section, we will consider a variation of M_{CDED} when applied to our original strategy space $\mathbb{N} \times \mathcal{F} \times \mathcal{H}$. For this, we will assume that M_{CDED} will return $A_i = Z_i$ as the agent's allocation, and then an agent can use X_i, Y_i, Z_i to estimate μ. In this situation, below we show that the agent can achieve a smaller penalty using a weighted average over $X_i \cup Z_i$ instead of the sample mean used by the mechanism. Here, the weights are proportional to the inverse of the variance of each data point. (Our mechanism purposefully uses the sub-optimal sample mean in the restricted strategy space $\mathbb{N} \times \mathcal{F}$ as a way to shape the agent's penalty and incentivize good behavior.)

This shows that M_{CDED} (with the above modification) is not NIC in this more general strategy space. The agent can obtain a lower penalty using a better estimator (such as the weighted average we show over here) and achieve a lower penalty. More importantly, as the agent knows that she can achieve a lower estimation error via a better estimator instead of more data, she can leverage this insight to collect less data and reduce her penalty even further.

We should emphasize that it is unclear if this weighted average is minimax optimal. It is also unclear if there exists a Nash equilibrium for M_{CDED} (or any straightforward modification of M_{CDED}) in the expanded strategy space.

The weighted average estimator: We will now present the weighted average estimator that achieves a lower maximum risk. To show this, first note that for all $x \in X_i$, $\mathbb{V}[x] = \sigma^2$; when $(n_i, f_i) = (n_i^\star, f_i^\star)$, for all $x \in Z_i$,

$$\begin{aligned}
\mathbb{V}[x] =& \mathbb{E}\Big[(z + \epsilon_{z,i} - \mu)^2\Big] = \sigma^2 + \beta_\epsilon^2 \mathbb{E}\Big[(\widehat{\mu}(X_i) - \widehat{\mu}(Y_{-i}))^{2k_\epsilon}\Big] \\
=& \sigma^2 + \frac{n_i^{\star k_\epsilon - 2}(m-1)^{k_\epsilon - 1}(mn_i^\star)^2}{k_\epsilon (2k_\epsilon - 1)!!\, m^{k_\epsilon+1}\sigma^{2k_\epsilon-2}}(2k_\epsilon - 1)!! \left(\frac{1}{n_i^\star} + \frac{1}{(m-1)n_i^\star}\right)^{k_\epsilon} \sigma^{2k_\epsilon} \\
=& \sigma^2 + \frac{n_i^{\star k_\epsilon}(m-1)^{k_\epsilon-1}}{k_\epsilon m^{k_\epsilon - 1}} \frac{m^{k_\epsilon}}{(m-1)^{k_\epsilon} n_i^{\star k_\epsilon}} \sigma^2 \\
=& \sigma^2 + \frac{1}{k_\epsilon}\frac{m}{m-1}\sigma^2
\end{aligned}$$

Consider the following weighted-average estimator:

$$h_i(X_i, Y_i, (Z_i, \eta_i^2)) = \frac{\frac{1}{\sigma^2}\sum_{x \in X_i} x + \frac{1}{\sigma^2 + \frac{1}{k_\epsilon}\frac{m}{m-1}\sigma^2}\sum_{x \in Z_i} x}{\frac{n_i^\star}{\sigma^2} + \frac{(m-1)n_i^\star}{\sigma^2 + \frac{1}{k_\epsilon}\frac{m}{m-1}\sigma^2}}$$

The maximum risk of h_i is

$$\begin{aligned}
\mathbb{E}\left[\Big(h_i(X_i, Y_i, (Z_i, \eta_i^2)) - \mu\Big)^2\right] =& \frac{1}{\frac{n_i^\star}{\sigma^2} + \frac{(m-1)n_i^\star}{\sigma^2 + \frac{1}{k_\epsilon}\frac{m}{m-1}\sigma^2}} = \frac{1}{1 + \frac{m-1}{1 + \frac{1}{k_\epsilon}\frac{m}{m-1}}} \frac{\sigma^2}{n_i^\star} = \frac{1 + \frac{1}{k_\epsilon}\frac{m}{m-1}}{m + \frac{1}{k_\epsilon}\frac{m}{m-1}} \frac{\sigma^2}{n_i^\star} \\
<& \frac{\left(1 + \frac{1}{k_\epsilon}\right)\left(1 + \frac{1}{k_\epsilon}\frac{1}{m-1}\right)}{m + \frac{1}{k_\epsilon}\frac{m}{m-1}} \frac{\sigma^2}{n_i^\star} = \left(1 + \frac{1}{k_\epsilon}\right)\frac{\sigma^2}{mn_i^\star} \qquad \text{(F.23)}
\end{aligned}$$

Note that the RHS of (F.23) is the risk of the sample average deployed by M_{CDED}. This means, suppose all other agents choose $s^\star$, then agent i can choose a weighted average to reduce her penalty without collecting more data.

F.5 High dimensional mean estimation with bounded variance

In this section, we will study estimating a d–dimensional mean $\mu(\theta) \in \mathbb{R}^d$ for distributions θ with bounded variance. We will focus on our original setting in Section 8.2, but will outline the modifications to the formalism to accommodate the generality. For $x \in \mathbb{R}^d$, let $x^{(i)}$ denote the i^{th} dimension.

Modifications to the setting in Section 8.2: First, we should change the definitions of $\mathcal{F}, \mathcal{H}$ and $\mathcal{M}$ in equations 8.1 and (8.2) to account for the fact that the data is d dimensional. For instance, the space of functions mapping the dataset collected to the dataset submitted should be defined as $\mathcal{F} = \{f : \bigcup_{n\geq 0} \mathbb{R}^{d\times n} \to \bigcup_{n\geq 0} \mathbb{R}^{d\times n}\}$. Next, let $\Theta = \{\theta;\ \textbf{supp}\,(\theta) \subset \mathbb{R}^d,\ \mathbb{E}_{x\sim\theta}\left[(x^{(i)} - \mu(\theta)^{(i)})^2\right] \leq \sigma^2, \forall\, i \in [d]\}$ be the class of all d–dimensional distributions where the variance along each dimension is bounded by σ^2. Here, the maximum variance σ^2 is known and is public information. Note that we do not assume that the individual dimensions are independent. An agent's penalty p_i is defined similar to (8.3) but considers the maximum risk over Θ, i.e

$$p_i(M, s) = \sup_{\theta\in\Theta} \mathbb{E}[\|h_i(X_i, Y_i, A_i) - \mu(\theta)\|_2^2 \mid \theta] \ + \ cn_i. \tag{F.24}$$

Finally, the social penalty and ratio PR are as defined in (8.5), but with the above definition for p_i.

Mechanism: Our mechanism for this problem is the same as the one outlined in Algorithm 10, with the following cosmetic modifications. First, the allocation space should now be $\mathcal{A} = \bigcup_{n\geq 0} \mathbb{R}^{d\times n} \times \bigcup_{n\geq 0} \mathbb{R}^{d\times n} \times \mathbb{R}_+^d$. The noise modulating parameter α is determined by a similar equation as in (8.7), but with c replaced with c/d. In line 12 of Algorithm 10, we should set the size of the dataset Z_i to be $\min\{|Y_{-i}|, \sigma\sqrt{d/(cm)}\}$. Finally, the operations in lines 13 and 14 should be interpreted as d–dimensional operations that are performed elementwise. The

recommended strategy $s_i^\star = (n_i^\star, f_i^\star, h_i^\star)$ for agent i is as follows:

$$n_i^\star = \begin{cases} \frac{\sigma}{m}\sqrt{\frac{d}{c}} & \text{if } m \le 4, \\ \sigma\sqrt{\frac{d}{cm}} & \text{if } m \ge 5 \end{cases}, \qquad f_i^\star = \mathbf{I}, \tag{F.25}$$

$$h_i^\star(X_i, Y_i, (Z_i, Z_i', \eta_i^2)) = \frac{\frac{1}{\sigma^2}\sum_{u\in X_i\cup Z_i} u + \frac{1}{\sigma^2+\tau_i^2}\sum_{u\in Z_i'} u}{\frac{1}{\sigma^2}|X_i \cup Z_i'| + \frac{1}{\sigma^2+\tau_i^2}|Z_i'|}, \qquad \text{where, } \tau_i^2 = \frac{2\alpha^2\sigma^2}{n_i^\star} \in \mathbb{R}_+.$$

Above, one difference worth highlighting is the change in the recommended estimator $h_i^\star$. Previously, the weighting used the η_i^2 term returned by the mechanism, which is a function of Y_i and Z_i. This data-dependent weighting was necessary to obtain an *exactly* (i.e including constants) minimax optimal estimator for the corrupted dataset, which in turn was necessary to achieve an exact Nash equilibrium. However, bounding the risk when using a data-dependent weighting is challenging when the Gaussian assumption does not hold. Instead, here we use a deterministic weighting via the quantity τ_i^2. While this is not exactly minimax optimal, we can show that its maximum risk is very close to a lower bound, which helps us obtain an approximate Nash equilibrium. It is worth pointing out that designing exactly minimax optimal estimators, even under i.i.d assumptions, is challenging for general classes of distributions (Lehmann and Casella, 2006).

The following theorem states the main properties of this mechanism.

Theorem F.5.1. *The following statements are true about the mechanism M_{C3D} in Algorithm 10 with the above modifications. (i) The strategy profile $s^\star$ as defined in (F.25) is an approximate Nash equilibrium, i.e if all agents except i are following $s^\star$, then for any alternative strategy s_i for agent i, we have $p_i(M_{\text{C3D}}, s^\star) \le p_i(M_{\text{C3D}}, (s_{-i}^\star, s_i))(1 + 5/m)$ (ii) The mechanism is individually rational at $s^\star$. (iii) The mechanism is approximately efficient at $s_i^\star$, with $\text{PR}(M_{\text{C3D}}, s^\star) < 2 + 10/m$.*

We see that even under this more general setting, our mechanism retains its main properties with only a slight weakening of the results. We now have approximate, instead of exact, NIC, with the benefit of deviation diminishing as there are more

agents. Similarly, the bound on the efficiency is only slightly weaker than the one in Theorem 8.3.1.

Proof of Theorem F.5.1

When $m \leq 4$, the claims follow using the exact steps in Section F.1. Therefore, we focus on the case $m \geq 5$. Moreover, some of the key steps of this proof follows along similar lines to Theorem 8.3.1, so we will provide an outline and focus on the differences.

Approximate Nash incentive compatibility. We will first prove the statement (i) of Theorem F.5.1, which states that $s_i^\star$, as defined in (F.25), is an approximate Nash equilibrium for M_{C3D}. That is, we will show that the maximum possible reduction in penalty for an agent i when deviating from $s_i^\star$ is small, provided that all other agents are following $s_{-i}^\star$.

For this, we will first lower bound the penalty p_i (F.24) using the family of independent Gaussian distributions. Let $\Theta_{\mathcal{N}} = \left\{\mathcal{N}(\mu, \sigma^2 I_d) : \mu \in \mathbb{R}^d\right\}$ denote the space of d–dimensional normal distributions with identity covariance matrix. For any mechanism M and strategy profile $s \in \mathcal{S}^m$, we define the penalty of agent i restricted to $\Theta_{\mathcal{N}}$ as:

$$p_i^{\mathcal{N}}(M, s) = \sup_{\theta \in \Theta_{\mathcal{N}}} \mathbb{E}[\|h_i(X_i, Y_i, A_i) - \mu(\theta)\|_2^2 \mid \theta] \; + \; cn_i.$$

Since $\Theta_{\mathcal{N}} \subset \Theta$, it is straightforward to see that for all $M \in \mathcal{M}$ and $s \in \mathcal{S}^m$,

$$p_i^{\mathcal{N}}(M, s) \leq p_i(M, s). \tag{F.26}$$

We will now use this result to lower bound the penalty of an agent for any other alternative strategy. First note that, by independence, the mean estimation problem on $\Theta_{\mathcal{N}}$ can be viewed as d independent copies of the univariate normal mean estimation problem considered in Theorem 8.3.1 but with c replaced with c/d. Let $\tilde{h}_i^\star$ be the weighted average that applies the estimator in (8.8) along each dimension. And let $\tilde{s}_i^\star = (n_i^\star, f_i^\star, \tilde{h}_i^\star)$. We can now lower bound the penalty of agent i when

following any (alternative) strategy $s_i \in \mathcal{S}$, provided that other agents are following $s^\star_{-i}$. We have:

$$\begin{aligned}
p_i(M_{\text{C3D}}, (s_i, s^\star_{-i})) &= p_i\Big(M_{\text{C3D}}, \Big(s_i, \Big(n^\star_{-i}, f^\star_{-i}, h^\star_{-i}\Big)\Big)\Big) \\
&\geq p_i^{\mathcal{N}}\Big(M_{\text{C3D}}, \Big(s_i, \Big(n^\star_{-i}, f^\star_{-i}, h^\star_{-i}\Big)\Big)\Big) && \text{(By (F.26))} \\
&= p_i^{\mathcal{N}}\Big(M_{\text{C3D}}, \Big(s_i, \Big(n^\star_{-i}, f^\star_{-i}, \tilde{h}^\star_{-i}\Big)\Big)\Big) \\
&\qquad \text{(As agent } i\text{'s penalty will not be affected by other agents' estimators)} \\
&\geq p_i^{\mathcal{N}}\Big(M_{\text{C3D}}, \Big(\Big(n^\star_i, f^\star_i, \tilde{h}^\star_i\Big), \Big(n^\star_{-i}, f^\star_{-i}, \tilde{h}^\star_{-i}\Big)\Big)\Big) \\
&\qquad \text{(By adapting the analysis in Section F.1.)} \\
&= p_i^{\mathcal{N}}(M_{\text{C3D}}, \tilde{s}^\star) && \text{(F.27)}
\end{aligned}$$

Above, the second step uses (F.26) and the third step uses the fact that other agent's *estimator* will not affect agent i's penalty. The fourth step uses the fact that for estimation problems in $\Theta_{\mathcal{N}}$, the strategy profile $\tilde{s}^\star = \{(n^\star_i, f^\star_i, \tilde{h}^\star_i)\}_i$ is a Nash equilibrium; in Section F.1, we showed this for the one dimensional case, but this proof can be easily adapted to d dimensions since we are assuming an identity covariance matrix in $\Theta_{\mathcal{N}}$. Finally, by adapting the analysis in Section F.1, we can obtain the following expression for agent i's penalty $p_i^{\mathcal{N}}(M_{\text{C3D}}, \tilde{s}^\star)$ in $\Theta_{\mathcal{N}}$:

$$p_i^{\mathcal{N}}(M_{\text{C3D}}, \tilde{s}^\star) = \; d\sigma\sqrt{\frac{c/d}{m}}\left(\frac{\frac{10\alpha^2}{n^\star_i} - 1}{\frac{4\alpha^2}{n^\star_i}\frac{m+1}{m} - 1} + 1\right) \tag{F.28}$$

To state the approximate NIC result, we will now upper bound the penalty of the agent when following $s^\star_i$. Using the bounded variance assumption, we have:

$$p_i(M, s^\star) = \sup_{\theta \in \Theta} \mathbb{E}\left[\left\| \frac{\frac{1}{\sigma^2}\sum_{u \in X_i \cup Z_i} u + \frac{1}{\sigma^2+\tau_i^2}\sum_{u \in Z'_i} u}{\frac{1}{\sigma^2}|X_i \cup Z'_i| + \frac{1}{\sigma^2+\tau_i^2}|Z'_i|} - \mu(\theta)\right\|_2^2 \middle| \theta\right] + cn^\star_i$$

$$= \sup_{\theta\in\Theta} \sum_{k=1}^{d} \mathbb{E}\left[\left(\frac{\frac{1}{\sigma^2}\sum_{u\in X_i\cup Z_i}\left(u^{(k)} - \mu(\theta)^{(k)}\right) + \frac{1}{\sigma^2+\tau_i^2}\sum_{u\in Z_i'}\left(u^{(k)} - \mu(\theta)^{(k)}\right)}{\frac{1}{\sigma^2}|X_i\cup Z_i'| + \frac{1}{\sigma^2+\tau_i^2}|Z_i'|} \right)^2 \middle| \theta \right] + cn_i^\star$$

$$= \sup_{\theta\in\Theta} \sum_{k=1}^{d} \frac{\frac{1}{\sigma^2}\sum_{u\in X_i\cup Z_i}\mathbb{E}\left[\left(u^{(k)} - \mu(\theta)^{(k)}\right)\right] + \frac{1}{\sigma^2+\tau_i^2}\sum_{u\in Z_i'}\mathbb{E}\left[\left(u^{(k)} - \mu(\theta)^{(k)}\right)\right]}{\frac{1}{\sigma^2}|X_i\cup Z_i'| + \frac{1}{\sigma^2+\tau_i^2}|Z_i'|} + cn_i^\star \tag{F.29}$$

$$\leq \frac{d}{\frac{2n_i^\star}{\sigma^2} + \frac{(m-2)n_i^\star}{\sigma^2+\frac{2\alpha^2\sigma^2}{n_i^\star}}} + cn_i^\star = \frac{\sigma^2}{n_i^\star}\frac{d}{2+\frac{m-2}{1+\frac{2\alpha^2}{n_i^\star}}} + cn_i^\star = \sigma\sqrt{\frac{cd}{m}}\left(\frac{m}{2+\frac{m-2}{1+\frac{2\alpha^2}{n_i^\star}}} + 1\right), \tag{F.30}$$

where (F.29) is because: for all $k \in [d]$, $\forall x_1^{(k)}, x_2^{(k)} \in X_i \cup Z_i$, $\forall z_1^{(k)}, z_2^{(k)} \in Z_i'$, $x_1^{(k)} - \mu^{(k)}, x_2^{(k)} - \mu^{(k)}, z_1^{(k)} - \mu^{(k)}, z_2^{(k)} - \mu^{(k)}$ are uncorrelated pairwise. The final inequality is due to the bounded variance assumption.

Next, for brevity, let us write $A_m := \frac{\alpha}{\sqrt{n_i^\star}}$ where α is as defined in (8.7). By adapting the analysis in Section F.1, we can show that

$$A_m := \frac{\alpha}{\sqrt{n_i^\star}} \in \left(1, 1 + \frac{C_m}{m}\right), \qquad \text{where,} \quad C_m = \begin{cases} 20, & \text{if } m \leq 20 \\ 5, & \text{if } m > 20 \end{cases}. \tag{F.31}$$

By combining the results in (F.27), (F.30), and (F.31), we obtain the following bound:

$$\begin{aligned} \frac{p_i(M_{\text{C3D}}, s^\star)}{\inf_{s_i} p_i(M_{\text{C3D}}, (s_i, s_{-i}^\star))} - 1 \leq & \frac{p_i(M_{\text{C3D}}, s^\star)}{p_i^{\mathcal{N}}(M_{\text{C3D}}, \tilde{s}^\star)} - 1 \\ \leq & \frac{\sigma\sqrt{\frac{cd}{m}}\left(\frac{m}{2+\frac{m-2}{1+2A_m^2}} + 1\right)}{d\sigma\sqrt{\frac{c/d}{m}}\left(\frac{10A_m^2-1}{4A_m^2\frac{m+1}{m}-1} + 1\right)} - 1 = \frac{\frac{m}{2+\frac{m-2}{1+\frac{2\alpha^2}{n_i^\star}}} + 1}{\frac{\frac{10\alpha^2}{n_i^\star}-1}{\frac{4\alpha^2}{n_i^\star}\frac{m+1}{m}-1} + 1} - 1 \\ = & \frac{4A_m^2((A_m^2-1)m + 1 - 4A_m^2)m}{(4A_m^2+m)((7A_m^2-1)m + 2A_m^2)} =: E(m). \end{aligned} \tag{F.32}$$

Let $E(m)$ denote the final upper bound obtained above. Next, we will prove $E(m) <$

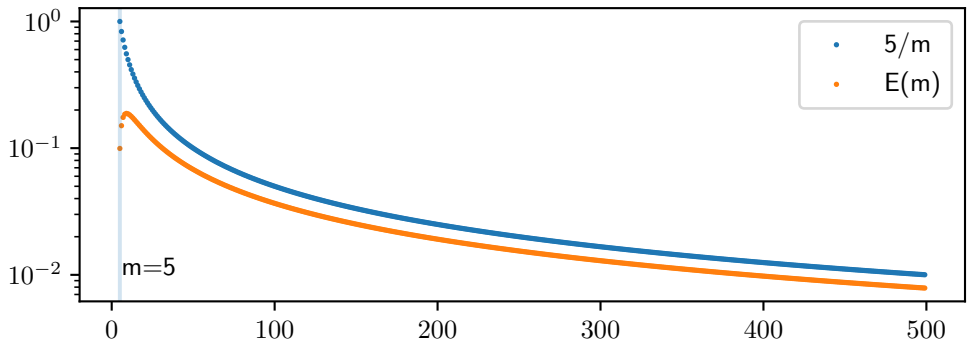


Figure F.2: $E(m)$ plot. See `G_em_plot.py`.

$5/m$. When $m \in [5, 500]$, this can be individually verified for each value of $E(m)$ (see Figure F.2). When $m \geq 500$, we have $A_m \leq 1.01$ (see (F.31)). From this we can conclude,

$$E(m) \leq \frac{4 \times 1.01^2 \times (2.01 \times \frac{5}{m} m - 3)m}{6m^2} < \frac{5}{m}. \tag{F.33}$$

Combining the results in (F.32) and (F.33), we obtain the following approximate NIC result:

$$\forall\, i \in [m],\, s_i \in \mathcal{S}, \quad p_i(M_{\text{C3D}}, s^\star) \leq p_i(M_{\text{C3D}}, (s_i, s^\star_{-i})) \left(1 + \frac{5}{m}\right).$$

Individual rationality: This proof is very similar to the proof in Section F.1. In particular, using calculations similar to (F.30), we can show that regardless of the choice of n_i, the agent's penalty is strictly smaller when using the uncorrupted (Z_i) and corrupted (Z_i') datasets along with the weighted average in (F.25).

Approximate efficiency: To bound the penalty ratio, first note that by (F.28) and using the same reasoning as Section F.1, we have that

$$\frac{\sum_i p_i^{\mathcal{N}}(M_{\text{C3D}}, \tilde{s}^\star)}{\inf_{M \in \mathcal{M}, s \in \mathcal{S}^m} \sum_i p_i^{\mathcal{N}}(M, s)} = \frac{m p_i^{\mathcal{N}}(M_{\text{C3D}}, \tilde{s}^\star)}{\inf_{M \in \mathcal{M}, s \in \mathcal{S}^m} \sum_i p_i^{\mathcal{N}}(M, s)} = \frac{m p_i^{\mathcal{N}}(M_{\text{C3D}}, \tilde{s}^\star)}{2\sigma\sqrt{cmd}} \leq 2. \tag{F.34}$$

Next, as $\Theta_{\mathcal{N}} \subset \Theta$, and noting that $P(M,s) = \sum_i p_i(M,s)$ for all M, s, we can also write,

$$\inf_{M\in\mathcal{M},s\in\mathcal{S}^m} \sum_i p_i^{\mathcal{N}}(M,s) \leq \inf_{M\in\mathcal{M},s\in\mathcal{S}^m} P(M,s). \tag{F.35}$$

We can combine the above results to obtain the following upper bound on PR:

$$\begin{aligned}
\mathrm{PR}(M_{\text{C3D}}, s^\star) &= \frac{P(M_{\text{C3D}}, s^\star)}{\inf_{M\in\mathcal{M},s\in\mathcal{S}^m} P(M,s)} \leq \frac{mp_i(M_{\text{C3D}}, s^\star)}{\inf_{M\in\mathcal{M},s\in\mathcal{S}^m} \sum_i p_i^{\mathcal{N}}(M,s)} && \text{(By (F.35))} \\
&= \frac{mp_i^{\mathcal{N}}(M_{\text{C3D}}, \tilde{s}^\star)}{\inf_{M\in\mathcal{M},s\in\mathcal{S}^m} \sum_i p_i^{\mathcal{N}}(M,s)} \frac{p_i(M_{\text{C3D}}, s^\star)}{p_i^{\mathcal{N}}(M_{\text{C3D}}, \tilde{s}^\star)} \\
&\leq 2\,\frac{p_i(M_{\text{C3D}}, s^\star)}{p_i^{\mathcal{N}}(M_{\text{C3D}}, \tilde{s}^\star)} && \text{(By (F.34))} \\
&= 2(1 + E(m))) && \text{(By definition of } E(m)\text{, see (F.32))} \\
&< 2 + \frac{10}{m}. && \text{(By (F.33))}
\end{aligned}$$

This establishes approximate efficiency for M_{C3D} for the high dimensional setting.

F.6 Application to Bayesian Settings

While our results study the Normal mean estimation in frequentist statistics, the main ideas can also be applied to the Bayesian setting. When the Normal mean admits a zero-mean normal prior, the major proof steps remain the same. Specifically, our current analysis constructs a sequence of Gaussian priors and takes the limit to prove the minimax optimality. In the Bayesian setting, one can simply skip the step in (F.12), which takes the limit w.r.t. the prior sequence. The other steps remain the same.

F.7 Useful Results

In this section, we will state some useful results that we have used throughout this proof.

Lemma F.7.1 (Hardy-Littlewood inequality, Lemma 1.6 in Burchard (2009))**.** *Let f and g be non-negative measurable functions that vanish at infinity. Let f^* and g^* to denote the symmetric decreasing rearrangement of f and g. If $\int f^* g^* < \infty$, then,*

$$\int fg \leq \int f^* g^*.$$

Next, we will use the above result to derive a corollary that will be useful in our proofs.

Lemma F.7.2 (A corollary of Hardy-Littlewood)**.** *Let f, g be nonnegative even functions such that,*

- *f is monotonically increasing on $[0, \infty)$.*
- *g is monotonically decreasing on $[0, \infty)$, and has a finite integral $\int_{\mathbb{R}} g(x)dx < \infty$.*
- *$\forall a$, $\int_{\mathbb{R}} f(x-a)g(x)dx < \infty$.*

Then for all a,

$$\int_{\mathbb{R}} f(x)g(x)dx \leq \int_{\mathbb{R}} f(x-a)g(x)dx$$

Proof. We will break this proof into two cases. The first is when $\sup f < \infty$ and the second is when $\sup f = \infty$. First consider the case $\sup f < \infty$. Let

$$M := \lim_{x \to \infty} f(x).$$

By using Lemma F.7.1, $\forall a$,

$$\int_{\mathbb{R}} (M - f(x))g(x)dx \geq \int_{\mathbb{R}} (M - f(x-a))g(x)dx.$$

The result follows after rearrangement.

If $\sup f = \infty$, let $f_n(x) := \min\{f(x), n\}$. For all n and a, by Lemma F.7.1,

$$\int_{\mathbb{R}} (n - f_n(x))g(x)dx \geq \int_{\mathbb{R}} (n - f_n(x-a))g(x)dx,$$

thus

$$\int_{\mathbb{R}} f_n(x)g(x)dx \leq \int_{\mathbb{R}} f_n(x-a)g(x)dx.$$

Note that $|f_n(x)g(x)| \leq f(x)g(x)$, the result follows by letting $n \to \infty$ on both sides and using dominated convergence theorem. ■

Below, we provide a brief example on using Lemma F.7.2 to calculate the Bayes risk in a normal mean estimation problem with i.i.d data. While it is not necessary to use Hardy-Littlewood for this problem, this example will illustrate how we have used it in our proofs.

Example F.7.1. Consider the Normal mean estimation problem given samples $X_{[n]} \sim \mathcal{N}(\mu, \sigma^2)$, where μ admits a prior distribution $\mathcal{N}(0, \ell^2)$. The goal is to minimize the average risk:

$$\mathbb{E}_{\mu \sim \mathcal{N}(0,\ell^2)} \Big[\mathbb{E}_{X_{[n]} \sim \mathcal{N}(\mu,\sigma^2)} [L(\hat{\mu} - \mu) | \mu] \Big],$$

where the loss function, $L(\cdot)$, is an even function that increases on $[0, \infty)$. By a standard argument, one can show that the posterior distribution of μ conditioned on $X_{[n]}$ is Gaussian with data-dependent parameters $\bar{\mu}, \bar{\sigma}^2$:

$$\mu | X_{[n]} \sim \mathcal{N}(\overline{\mu}, \overline{\sigma}^2).$$

The posterior risk is:

$$\mathbb{E}_{\mu|X_{[n]}}[L(\hat{\mu} - \mu)] = \mathbb{E}_{\mu|X_{[n]}}[L((\mu - \overline{\mu}) + (\overline{\mu} - \hat{\mu}))] = \int_{\mathbb{R}} \underbrace{L(x + (\overline{\mu} - \hat{\mu}))}_{=: f(x + (\overline{\mu} - \hat{\mu}))} \underbrace{\frac{\exp\left(-\frac{x^2}{2\overline{\sigma}^2}\right)}{\overline{\sigma}\sqrt{2\pi}}}_{=: g(x)} dx$$

By applying Lemma F.7.2 with f and g, the posterior risk above is minimized when $\hat{\mu} = \overline{\mu}$. So is the average risk.

The next Lemma shows that convexity is preserved under expectation under certain conditions.

Lemma F.7.3. *Let y be a random variable and $f(x, y)$ be a function s.t.*

- *$f(x, y)$ is convex in x;*
- *$\mathbb{E}_y[|f(x, y)|] < \infty$ for all x.*

Then $\mathbb{E}_y[f(x, y)]$ is also convex in x.

Proof. For any x_1, x_2, we have

$$\frac{\mathbb{E}_y[f(x_1, y)] + \mathbb{E}_y[f(x_2, y)]}{2} = \mathbb{E}_y\left[\frac{f(x_1, y) + f(x_2, y)}{2}\right] \geq \mathbb{E}_y\left[f\left(\frac{x_1 + x_2}{2}, y\right)\right]$$

■

Lemma F.7.4 (Centered moments of normal random variable)**.** *Let $X \sim \mathcal{N}(\mu, \sigma^2)$ be a normal random variable and $p \in \mathbb{Z}_+$, then*

$$\mathbb{E}[(X - \mu)^p] = \begin{cases} 0 & \text{if } p \text{ is odd} \\ \sigma^p (p-1)!! & \text{if } p \text{ is even} \end{cases}.$$

Some technical results

Next, we will state some technical results that were obtained purely using algebraic manipulations and are not central to the main proof ideas. The first result states upper and lower bounds on the Gaussian complementary error function using an asymptotic expansion.

Lemma F.7.5 (Erfc bound)**.** *For all $x > 0$,*

$$\operatorname{Erfc}(x) \leq \frac{1}{\sqrt{\pi}}\left(\frac{\exp(-x^2)}{x} - \frac{\exp(-x^2)}{2x^3} + \frac{3\exp(-x^2)}{4x^5}\right) \tag{F.36}$$

$$\operatorname{Erfc}(x) \geq \frac{1}{\sqrt{\pi}}\left(\frac{\exp(-x^2)}{x} - \frac{\exp(-x^2)}{2x^3}\right) \tag{F.37}$$

Proof. By integration by parts:

$$\begin{aligned}
\frac{\sqrt{\pi}}{2}\operatorname{Erfc}(x) &= \int_x^\infty \exp\left(-t^2\right)dt = \left(-\frac{\exp(-t^2)}{2t}\right)\Big|_x^\infty - \int_x^\infty \frac{\exp(-t^2)}{2t^2}dt \\
&= \frac{\exp(-x^2)}{2x} - \left(\left(-\frac{\exp(-t^2)}{4t^3}\right)\Big|_x^\infty - \int_x^\infty \frac{3\exp(-t^2)}{4t^4}dt\right) \\
&= \frac{\exp(-x^2)}{2x} - \frac{\exp(-x^2)}{4x^3} + \underbrace{\int_x^\infty \frac{3\exp(-t^2)}{4t^4}dt}_{\geq 0} \qquad \text{(F.38)} \\
&= \frac{\exp(-x^2)}{2x} - \frac{\exp(-x^2)}{4x^3} + \left(-\frac{3\exp(-t^2)}{8t^5}\right)\Big|_x^\infty - \int_x^\infty \frac{15\exp(-t^2)}{8t^6}dt \\
&= \frac{\exp(-x^2)}{2x} - \frac{\exp(-x^2)}{4x^3} + \frac{3\exp(-x^2)}{8x^5} \underbrace{- \int_x^\infty \frac{15\exp(-t^2)}{8t^6}dt}_{\leq 0} \qquad \text{(F.39)}
\end{aligned}$$

The results follow by (F.38) and (F.39). ∎

Our next result, states an expression for the function $p(n_i)$ and its derivative as defined in (F.13).

Lemma F.7.6 (Value and derivative of penalty function at $s^\star$)**.** *Let*

$$p(n_i) = p_i(M_{\text{C3D}}, ((n_i, f_i^\star, h_i^\star), s_{-i}^\star))$$

(see (F.13)) and $s_i^\star, f_i^\star, h_i^\star$ be as specified in (8.8). The penalty of agent i in Algorithm 10

satisfies:

$$p(n_i^\star) = \frac{\sqrt{\frac{\alpha^2}{mn_i^\star}}\sigma^2\left(2m\sqrt{2\pi}\sqrt{\frac{\alpha^2}{mn_i^\star}} - \exp\left(\frac{mn_i^\star}{8\alpha^2}\right)(m-2)\pi\,\mathrm{Erfc}\left(\frac{1}{2\sqrt{2}\sqrt{\frac{\alpha^2}{mn_i^\star}}}\right)\right)}{4\sqrt{2\pi}\alpha^2} + cn_i^\star \tag{F.40}$$

$$\begin{aligned} p'(n_i^\star) = &-\frac{\sigma^2}{64\frac{\alpha^2}{m-2}\frac{\alpha}{\sqrt{mn_i^\star}}mn_i^\star}\left(\frac{4\alpha}{\sqrt{mn_i^\star}}\left(\frac{4\alpha^2 m}{(m-2)n_i^\star} - 1\right)\right. \\ &\left. - \exp\left(\frac{mn_i^\star}{8\alpha^2}\right)\left(\frac{4\alpha^2}{mn_i^\star}(m+1) - 1\right)\sqrt{2\pi}\,\mathrm{Erfc}\left(\frac{1}{2\sqrt{2}\sqrt{\frac{\alpha^2}{mn_i^\star}}}\right)\right) + c. \end{aligned} \tag{F.41}$$

This proof involves several algebraic manipulations, so we will provide an outline of our proof strategy. First, we will rearrange the denominator inside the expectation in (F.13), to write the LHS of (F.40) as $J + K\mathbb{E}\left[\frac{1}{L+x^2}\right]$, and the LHS of (F.41) as $J' + K'\mathbb{E}\left[\frac{1}{L+x^2}\right] + K''\mathbb{E}\left[\frac{1}{(L+x^2)^2}\right]$, where the expectation is with respect to a standard normal $\mathcal{N}(0,1)$ variable, J, K, K', K'', L are quantities that depend on $n_i, m, c, \sigma^2, \alpha^2$, and importantly, L is strictly larger than 0. Using properties of the normal distribution, in Lemma F.7.7, we prove the following result:

$$\mathbb{E}\left[\frac{1}{L+x^2}\right] = \sqrt{\frac{\pi}{2L}}\exp\left(\frac{L}{2}\right)\mathrm{Erfc}\left(\sqrt{\frac{L}{2}}\right) \tag{F.42}$$

$$\mathbb{E}\left[\frac{1}{(L+x^2)^2}\right] = \frac{\sqrt{\pi}}{2\sqrt{2}L^{3/2}}(1-L)\exp\left(\frac{L}{2}\right)\mathrm{Erfc}\left(\sqrt{\frac{L}{2}}\right) + \frac{1}{2L} \tag{F.43}$$

By plugging in these expressions and then substituting $n_i = n_i^\star$, we obtain (F.40) and (F.41).

Proof of Lemma F.7.6. We will rewrite $p(n_i^\star)$ and $p'(n_i^\star)$ as the Gaussian integral of

rational functions and use (F.42) to calculate their values. By (F.13),

$$
\begin{aligned}
p(n_i^\star) =& \mathbb{E}_{x\sim\mathcal{N}(0,1)}\left[\frac{1}{\frac{(m-2)n_i^\star}{\sigma^2+\alpha^2\left(\frac{\sigma^2}{n_i^\star}+\frac{\sigma^2}{n_i^\star}\right)x^2}+\frac{n_i^\star+n_i^\star}{\sigma^2}}\right]+cn_i^\star \\
=& \mathbb{E}_{x\sim\mathcal{N}(0,1)}\left[\frac{1}{\frac{(m-2)n_i^\star}{\sigma^2+\alpha^2\frac{2\sigma^2}{n_i^\star}x^2}+\frac{2n_i^\star}{\sigma^2}}\right]+cn_i^\star = \frac{\sigma^2}{n_i^\star}\mathbb{E}_{x\sim\mathcal{N}(0,1)}\left[\frac{1}{\frac{m-2}{1+\frac{2\alpha^2}{n_i^\star}x^2}+2}\right]+cn_i^\star \\
=& \frac{\sigma^2}{n_i^\star}\mathbb{E}_{x\sim\mathcal{N}(0,1)}\left[\frac{1}{2}-\frac{m-2}{2}\frac{1}{\frac{4\alpha^2}{n_i^\star}x^2+m}\right]+cn_i^\star \\
=& \frac{\sigma^2}{2n_i^\star}-\frac{\sigma^2}{n_i^\star}\frac{m-2}{2}\frac{n_i^\star}{4\alpha^2}\mathbb{E}_{x\sim\mathcal{N}(0,1)}\left[\frac{1}{x^2+\frac{mn_i^\star}{4\alpha^2}}\right]+cn_i^\star \\
=& \frac{\sigma^2}{2n_i^\star}-\frac{\sigma^2}{4\alpha^2}\frac{m-2}{2}\exp\left(\frac{mn_i^\star}{8\alpha^2}\right)\mathrm{Erfc}\left(\sqrt{\frac{mn_i^\star}{8\alpha^2}}\right)\sqrt{\frac{\pi}{\frac{mn_i^\star}{2\alpha^2}}}+cn_i^\star \\
& \left(\text{In (F.42), let } L=\tfrac{mn_i^\star}{4\alpha^2}\right) \\
=& \text{RHS of (F.40)}.
\end{aligned}
$$

To prove the second statement of Lemma F.7.6, by (F.14) and the dominated convergence theorem, we have:

$$
\begin{aligned}
p'(n_i^\star) =& \mathbb{E}_{x\sim\mathcal{N}(0,1)}\left[-\sigma^2\frac{1+\frac{(m-2)n_i^\star}{\left(1+\alpha^2\left(\frac{1}{n_i^\star}+\frac{1}{n_i^\star}\right)x^2\right)^2}\frac{\alpha^2x^2}{n_i^{\star 2}}}{\left(\frac{(m-2)n_i^\star}{1+\alpha^2\left(\frac{1}{n_i^\star}+\frac{1}{n_i^\star}\right)x^2}+n_i^\star+n_i^\star\right)^2}\right]+c \\
& \text{(By (F.14) and dominated convergence theorem)} \\
=& -\sigma^2\mathbb{E}_{x\sim\mathcal{N}(0,1)}\left[\frac{1+\frac{(m-2)n_i^\star}{\left(1+\frac{2\alpha^2}{n_i^\star}x^2\right)^2}\frac{\alpha^2x^2}{n_i^{\star 2}}}{\left(\frac{(m-2)n_i^\star}{1+\frac{2\alpha^2}{n_i^\star}x^2}+2n_i^\star\right)^2}\right]+c
\end{aligned}
$$

$$
\begin{aligned}
&= -\frac{\sigma^2}{n_i^{\star 2}}\mathbb{E}_{x\sim\mathcal{N}(0,1)}\left[\frac{1+\frac{(m-2)n_i^\star\alpha^2x^2}{\left(n_i^\star+2\alpha^2x^2\right)^2}}{\left(\frac{(m-2)n_i^\star}{n_i^\star+2\alpha^2x^2}+2\right)^2}\right]+c\\
&= -\frac{\sigma^2}{4n_i^{\star 2}}\mathbb{E}_{x\sim\mathcal{N}(0,1)}\left[\frac{4(n_i^\star+2\alpha^2x^2)^2+4(m-2)n_i^\star\alpha^2x^2}{((m-2)n_i^\star+2(n_i^\star+2\alpha^2x^2))^2}\right]+c\\
&= -\frac{\sigma^2}{4n_i^{\star 2}}\mathbb{E}_{x\sim\mathcal{N}(0,1)}\left[1+\frac{-(m-2)^2n_i^{\star 2}-4(m-2)n_i^\star(n_i^\star+2\alpha^2x^2)+4(m-2)n_i^\star\alpha^2x^2}{((m-2)n_i^\star+2(n_i^\star+2\alpha^2x^2))^2}\right]+c\\
&= -\frac{\sigma^2}{4n_i^{\star 2}}\mathbb{E}_{x\sim\mathcal{N}(0,1)}\left[1+(m-2)n_i^\star\frac{-(m-2)n_i^\star-4(n_i^\star+2\alpha^2x^2)+4\alpha^2x^2}{(4\alpha^2x^2+mn_i^\star)^2}\right]+c\\
&= -\frac{\sigma^2}{4n_i^{\star 2}}\mathbb{E}_{x\sim\mathcal{N}(0,1)}\left[1+(m-2)n_i^\star\frac{-(m+2)n_i^\star-4\alpha^2x^2}{(4\alpha^2x^2+mn_i^\star)^2}\right]+c\\
&= -\frac{\sigma^2}{4n_i^{\star 2}}+\frac{\sigma^2}{4n_i^{\star 2}}(m-2)n_i^\star\mathbb{E}_{x\sim\mathcal{N}(0,1)}\left[\frac{(4\alpha^2x^2+mn_i^\star)+2n_i^\star}{(4\alpha^2x^2+mn_i^\star)^2}\right]+c\\
&= -\frac{\sigma^2}{4n_i^{\star 2}}+\frac{\sigma^2}{4n_i^{\star 2}}(m-2)n_i^\star\mathbb{E}_{x\sim\mathcal{N}(0,1)}\left[\frac{1}{4\alpha^2x^2+mn_i^\star}+\frac{2n_i^\star}{(4\alpha^2x^2+mn_i^\star)^2}\right]+c\\
&= -\frac{\sigma^2}{4n_i^{\star 2}}+\frac{\sigma^2}{4n_i^{\star 2}}(m-2)n_i^\star\mathbb{E}_{x\sim\mathcal{N}(0,1)}\left[\frac{1}{4\alpha^2}\frac{1}{x^2+\frac{mn_i^\star}{4\alpha^2}}+\frac{2n_i^\star}{16\alpha^4}\frac{1}{\left(x^2+\frac{mn_i^\star}{4\alpha^2}\right)^2}\right]+c\\
&=c-\frac{\sigma^2}{4n_i^{\star 2}}+\frac{\sigma^2}{4n_i^{\star 2}}(m-2)n_i^\star\left(\frac{1}{4\alpha^2}+\frac{2n_i^\star}{16\alpha^4}\frac{1-\frac{mn_i^\star}{4\alpha^2}}{\frac{mn_i^\star}{2\alpha^2}}\right)\exp\left(\frac{mn_i^\star}{8\alpha^2}\right)\operatorname{Erfc}\left(\sqrt{\frac{mn_i^\star}{8\alpha^2}}\right)\sqrt{\frac{\pi}{\frac{mn_i^\star}{2\alpha^2}}}\\
&\quad+\frac{\sigma^2}{4n_i^{\star 2}}(m-2)n_i^\star\frac{2n_i^\star}{16\alpha^4}\frac{1}{\frac{mn_i^\star}{2\alpha^2}}\\
&\quad\left(\text{In (F.42) and (F.43) and let } L=\tfrac{mn_i^\star}{4\alpha^2}\right)\\
&=c-\frac{\sigma^2}{4n_i^{\star 2}}\left(1-\frac{(m-2)n_i^\star}{4\alpha^2m}\right)\\
&\quad+\frac{\sigma^2}{4n_i^{\star 2}}(m-2)n_i^\star\left(\frac{1}{4\alpha^2}+\frac{1}{4m\alpha^2}-\frac{n_i^\star}{16\alpha^4}\right)\exp\left(\frac{mn_i^\star}{8\alpha^2}\right)\operatorname{Erfc}\left(\sqrt{\frac{mn_i^\star}{8\alpha^2}}\right)\sqrt{\frac{\pi}{\frac{mn_i^\star}{2\alpha^2}}}\\
&=c-\frac{\sigma^2}{4n_i^{\star 2}}\left(1-\frac{(m-2)n_i^\star}{4\alpha^2m}\right)\\
&\quad+\frac{\sigma^2}{4n_i^{\star 2}}(m-2)n_i^\star\frac{\alpha\sqrt{2\pi}}{\sqrt{mn_i^\star}}\frac{n_i^\star}{16\alpha^4}\left(\frac{4\alpha^2}{mn_i^\star}(m+1)-1\right)\exp\left(\frac{mn_i^\star}{8\alpha^2}\right)\operatorname{Erfc}\left(\sqrt{\frac{mn_i^\star}{8\alpha^2}}\right)
\end{aligned}
$$

=RHS of (F.41)

■

We will now prove the statements in (F.42) and (F.43). Both statements follow from the Lemma below by substituting $t = 1/2$.

Lemma F.7.7. *For all $t \geq 0$ and some $L > 0$,*

$$I(t) := \int_{-\infty}^{\infty} \frac{1}{L+x^2}\frac{1}{\sqrt{2\pi}}\exp\Big(-tx^2\Big)dx = \exp(Lt)\,\mathrm{Erfc}(\sqrt{Lt})\sqrt{\frac{\pi}{2L}}$$
$$J(t) := \int_{-\infty}^{\infty} \frac{1}{(L+x^2)^2}\frac{1}{\sqrt{2\pi}}\exp\Big(-tx^2\Big)dx = \sqrt{\frac{\pi}{2L}}\Big(\frac{1}{2L}-t\Big)\exp(Lt)\,\mathrm{Erfc}(\sqrt{Lt}) + \frac{\sqrt{t}}{\sqrt{2}L}$$

Proof. We derive $I(t)$ and $J(t)$ as the solutions to two ODEs and solve the ODEs to obtain the results. Firstly, by calculation:

$$-I'(t) + LI(t) = \int_{-\infty}^{\infty} \frac{x^2+L}{L+x^2}\frac{1}{\sqrt{2\pi}}\exp\Big(-tx^2\Big)dx = \frac{1}{\sqrt{2t}}.$$

and

$$I(0) = \int_{-\infty}^{\infty} \frac{1}{L+x^2}\frac{1}{\sqrt{2\pi}}dx = \sqrt{\frac{\pi}{2L}}.$$

This means $I(t)$ satisfies the following ODE:

$$\begin{cases} -I'(t) + LI(t) = \frac{1}{\sqrt{2t}} \\ I(0) = \sqrt{\frac{\pi}{2L}} \end{cases}. \tag{F.44}$$

We solve (F.44) by multiplying integrating factor $-\exp(-Lt)$:

$$\exp(-Lt)I'(t) - L\exp(-Lt)I(t) = -\frac{1}{\sqrt{2t}}\exp(-Lt)$$

Note that the LHS is the derivative of $\exp(-Lt)I(t)$, the ODE becomes:

$$\frac{d}{dt}(\exp(-Lt)I(t)) = -\frac{1}{\sqrt{2t}}\exp(-Lt)$$

Integrating both sides over t, we get:

$$\begin{aligned}\exp(-Lt)I(t) &= -\int \frac{1}{\sqrt{2t}}\exp(-Lt)dt = -\int \frac{2}{\sqrt{2L}}\exp(-Lt)d\sqrt{Lt} \\ &= \operatorname{Erfc}(\sqrt{Lt})\sqrt{\frac{\pi}{2L}} + C,\end{aligned}$$

where we use integration by substitution for the last two equalities and C is some constant that does not depend on t. This means $I(t)$ satisfies the following form:

$$I(t) = \exp(Lt)\left(\operatorname{Erfc}(\sqrt{Lt})\sqrt{\frac{\pi}{2L}} + C\right)$$

Using the initial condition $I(0) = \sqrt{\frac{\pi}{2L}}$ and the fact that $\operatorname{Erfc}(0) = 0$, we conclude that $C = 0$. Thus

$$I(t) = \exp(Lt)\operatorname{Erfc}(\sqrt{Lt})\sqrt{\frac{\pi}{2L}}.$$

We can similarly derive an ODE for $J(t)$. By calculation:

$$\begin{aligned}-J'(t) + LJ(t) &= \int_{-\infty}^{\infty} \frac{x^2+L}{(L+x^2)^2}\frac{1}{\sqrt{2\pi}}\exp\left(-tx^2\right)dx = I(t) \\ J(0) &= \int_{-\infty}^{\infty} \frac{1}{(L+x^2)^2}\frac{1}{\sqrt{2\pi}}dx = \frac{1}{2L^{3/2}}\sqrt{\frac{\pi}{2}}\end{aligned}$$

Thus $J(t)$ satisfies the following ODE:

$$\begin{cases} -J'(t) + LJ(t) = I(t) \\ J(0) = \frac{1}{2L^{3/2}}\sqrt{\frac{\pi}{2}} \end{cases}. \tag{F.45}$$

We similarly multiply integrating factor $-\exp(-Lt)$ and integrate both sides:

$$\begin{aligned}&\int_0^t d\exp(-Lx)J(x) = -\int_0^t I(x)\exp(-Lx)dx = -\int_0^t \operatorname{Erfc}(\sqrt{Lx})\sqrt{\frac{\pi}{2L}}dx \\ &= -\left(x\operatorname{Erfc}(\sqrt{Lx})\sqrt{\frac{\pi}{2L}}\Big|_0^t + \int_0^t x\frac{\exp(-Lx)}{\sqrt{2x}}dx\right) \\ &\quad \text{(Integration by parts)}\end{aligned}$$

$$
\begin{aligned}
&= -t\,\mathrm{Erfc}(\sqrt{Lt})\sqrt{\frac{\pi}{2L}} - \frac{\sqrt{2}}{L^{3/2}}\int_0^{\sqrt{Lt}} y^2\exp(-y^2)dy \\
&\quad \Big(\text{Change of variable: } y=\sqrt{Lx}\Big) \\
&= -t\,\mathrm{Erfc}(\sqrt{Lt})\sqrt{\frac{\pi}{2L}} + \frac{\sqrt{2}}{L^{3/2}}\left(\frac{1}{2}y\exp(-y^2)\Big|_0^{\sqrt{Lt}} - \int_0^{\sqrt{Lt}}\frac{1}{2}\exp(-y^2)dy\right) \\
&\quad (\text{Integration by parts}) \\
&= -t\,\mathrm{Erfc}(\sqrt{Lt})\sqrt{\frac{\pi}{2L}} + \frac{\sqrt{2}}{L^{3/2}}\frac{1}{2}\sqrt{Lt}\exp(-Lt) - \frac{\sqrt{2}}{L^{3/2}}\int_0^{\sqrt{Lt}}\frac{1}{2}\exp(-y^2)dy \\
&= -t\,\mathrm{Erfc}(\sqrt{Lt})\sqrt{\frac{\pi}{2L}} + \frac{\sqrt{t}}{\sqrt{2}L}\exp(-Lt) - \frac{\sqrt{\pi}}{2\sqrt{2}L^{3/2}}\mathrm{Erf}\Big(\sqrt{Lt}\Big) \\
&\quad (\text{By definition of } \mathrm{Erf}) \\
&= -t\,\mathrm{Erfc}(\sqrt{Lt})\sqrt{\frac{\pi}{2L}} + \frac{\sqrt{t}}{\sqrt{2}L}\exp(-Lt) - \frac{\sqrt{\pi}}{2\sqrt{2}L^{3/2}}\Big(1-\mathrm{Erfc}\Big(\sqrt{Lt}\Big)\Big) \\
&\quad (\text{By definition of } \mathrm{Erfc}) \\
&= \Big(\frac{1}{2L}-t\Big)\,\mathrm{Erfc}(\sqrt{Lt})\sqrt{\frac{\pi}{2L}} + \frac{\sqrt{t}}{\sqrt{2}L}\exp(-Lt) - J(0) \\
&\quad (\text{By (F.45)})
\end{aligned}
$$

This means:

$$
\begin{aligned}
J(t) &= \exp(Lt)\left(\int_0^t d\exp(-Lx)J(x) + J(0)\right) \\
&= \exp(Lt)\left(\Big(\frac{1}{2L}-t\Big)\,\mathrm{Erfc}(\sqrt{Lt})\sqrt{\frac{\pi}{2L}} + \frac{\sqrt{t}}{\sqrt{2}L}\exp(-Lt)\right) \\
&= \sqrt{\frac{\pi}{2L}}\Big(\frac{1}{2L}-t\Big)\exp(Lt)\,\mathrm{Erfc}(\sqrt{Lt}) + \frac{\sqrt{t}}{\sqrt{2}L}
\end{aligned}
$$

■